Chapter One

Number systems

&

codes

Under graduate (2017)

Girma Adam (M.Tech)

Cont'd..

Topics discussed in this section:

Introduction

Analog versus Digital

Introduction to number systems

Number representation in binary

Conversion between bases

Signed number representation

Radix complement arithmetic

Diminished Radix complement Arithmetic

Codes

Introduction

- Digital Logic design is essential to understanding the design and working principles of a wide range of applications:-
- Computer are used in:
  1. Scientific calculation
  2. Image processing
  3. Air traffic control
  4. Educational field
  5. Industrial and commercial applications
- Telephone switching exchanges
- Digital camera
- Electronic calculators, PDA's ,Digital TV

Cont'd..

- The operation of these systems and many other systems are working based on the principles of digital techniques and these system are referred to as Digital System
- Characteristics of a digital system its manipulation of discrete elements of information. Such as:

  1. Electrical signal
  2. decimal digit
  3. Alphabet
  4. arithmetic operations
  5. punctuation marks and any set of meaningful symbols.

Benefits of Digital over Analog

1. Small size
2. Accuracy
3. Transmission
4. Noise immunity
5. Information storage
6. Computation (speed)
7. Ease of design
8. Data protection
9. Programmable etc..

Cont,d..

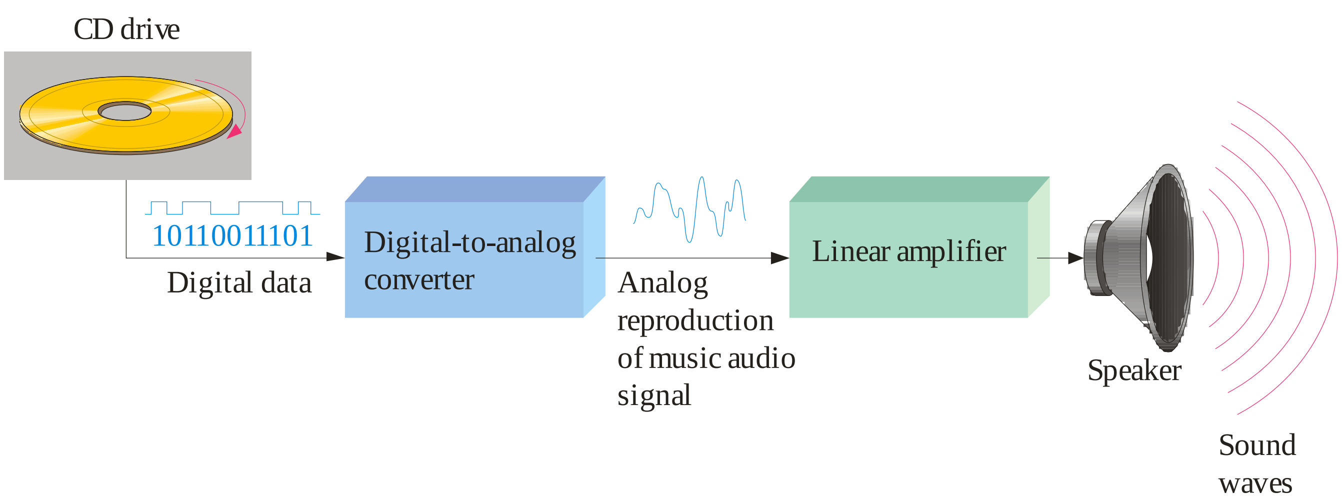

Intelligent networks allow handheld devices to receive news and emails, and to send text.
Video conferencing around the globe is in the palm of your hand.
Phones connect globally to share voice, text, and images.
The Human Network is everywhere.
Online gaming connects thousands of people seamlessly.

Cont,d..

- The real world, however, is analogue. Most physical quantities:-
  - position,
  - velocity,
  - acceleration,
  - force, pressure
  - temperature and flow rate
  - voice, video, etc

Analog and Digital Systems

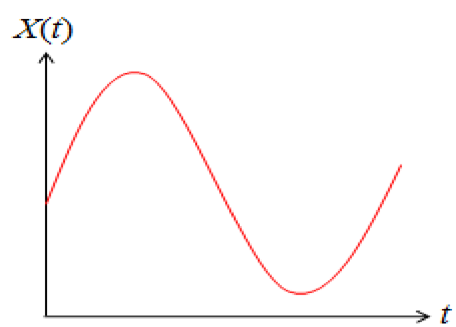
- Many systems use a mix of analog and digital electronics to take advantage of each technology. A typical CD player accepts digital data from the CD drive and converts it to an analog signal for amplification.



Cont'd..

1. Analogue:-is to express the numerical value of the quantity as or signals may vary a continuous range of values between the two expected extreme values (range).

example:- the temperature of an oven settable anywhere from 0 to 100 °C, may be measured to be 65 °C or 64.96 °C or 64.958 °C or even 64.9579 °C and so on.

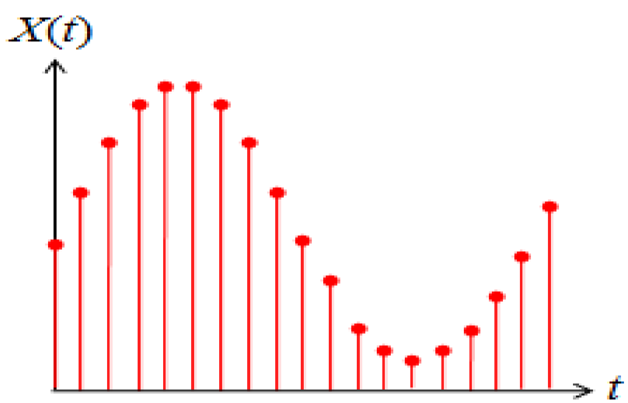


Analog signal

Cont'd..

2. Digital:-represents the numerical value of the quantity or signals can in steps of discrete values.

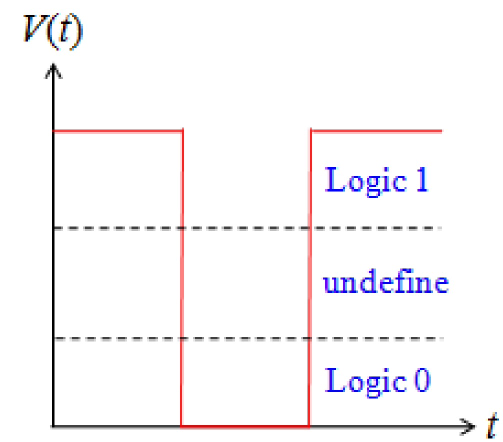
example:- the temperature of the oven may be represented in steps of 1 °C as 64 °C, 65 °C, 66 °C and so on.



Digital signal

Binary Digital Signal

- For digital systems, the variable takes on discrete values.
  - Two level, or binary values are the most prevalent values.
- Binary values are represented abstractly by:
  - Digits 0 and 1
  - Words (symbols) False (F) and True (T)
  - Words (symbols) Low (L) and High (H)
  - And words On and Off
- Binary values are represented by values or ranges of values of physical quantities.



Binary digital signal

The Digital Revolution

- Recently, many types of devices have been converted from analog to digital.

Examples:

Analog	Digital
Record albums	CDs
VHS tapes	DVDs
Analog television	Digital TV

- In all of these digital devices, info is stored and transmitted as long strings of 1s and 0s.

Number Systems

- The study of number systems is important from the viewpoint of understanding how data are represented before they can be processed by any digital system including

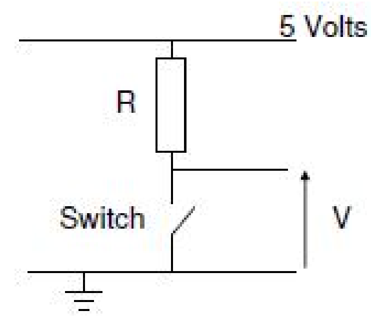
  - computer
  - Telephone switching exchanges
  - Digital camera, Electronic calculators, iPod's Digital TV etc

- Data :- physical representation of information

- A useful device is a switch

  – closed: V = 0 Volts

  – open: V = 5 Volts



Cont'd..

- Information can be :-

    - numbers, music, pictures, videos, text, etc

- Data can be either stored(e.g.: computer disk, DVD, SIM card, flash disc..etc) or transmitted(e.g.: fax, text message)
- Logic operations are the backbone of any digital computer, although solving a problem on computer could involve an arithmetic operation too.

Cont'd..

- The introduction of the mathematics of logic by George Boole laid the foundation for the modern digital computer.
- He reduced the mathematics of logic to a binary notation of '0' and '1'.
- The most fundamental is the number of independent digits or symbols used in the number system is known as the *radix* or *base* of the number system.

1. Decimal Number System

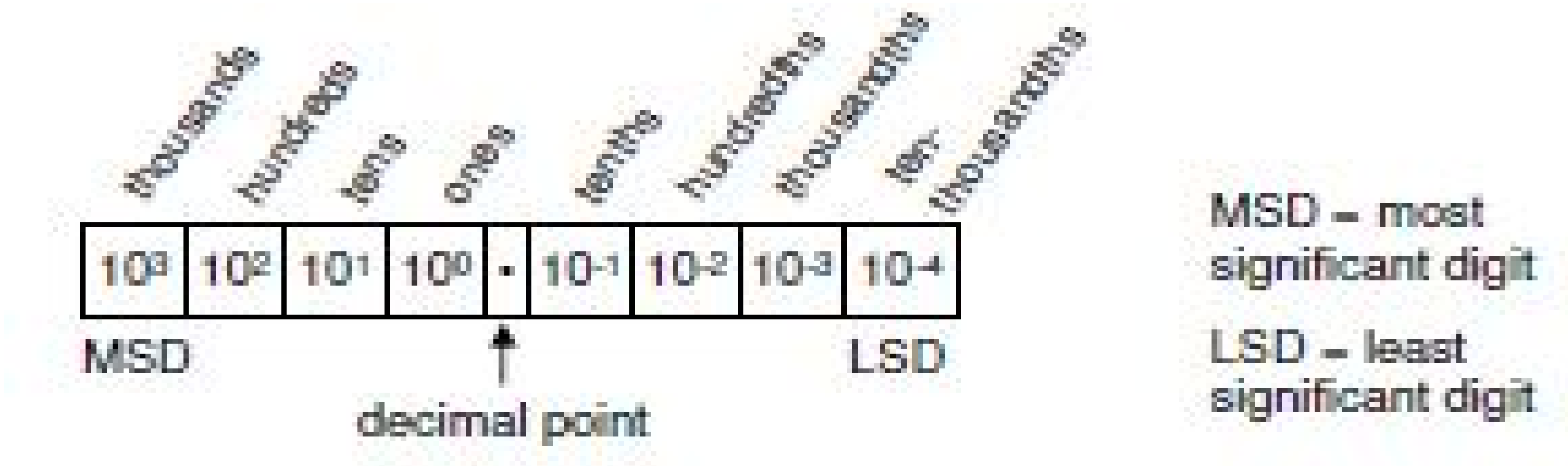
Deci = ten

Cont'd..

- The decimal number system is a radix-10 number system and therefore has 10 different digits or symbols. These are:-

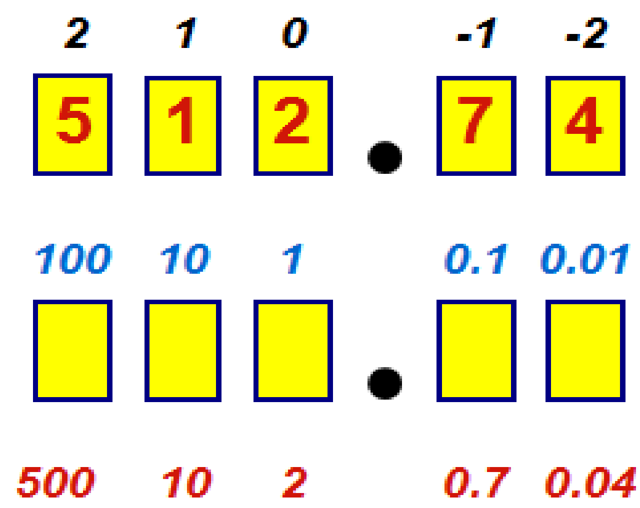
  0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

- The decimal number system has a base of 10, with each digit position weighted by a power of 10:



Cont'd..

- Base (also called radix) = 10
  - 10 digits { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 }
- Digit Position
  - Integer & fraction
- Digit Weight
  - Weight = $(Base)^{Position}$
- Magnitude
  - Sum of "*Digit* x *Weight*"
- Formal Notation



$d_2*B^2+d_1*B^1+d_0*B^0+d_{-1}*B^{-1}+d_{-2}*B^{-2}$

$(512.74)_{10}$

Cont'd..

- As an illustration, in the case of the decimal number 3586.265,
- The integer part (i.e. 3586) can be expressed as

$$3586 = 6\times10^0 + 8\times10^1 + 5\times10^2 + 3\times10^3$$

$$= 6+80+500+3000 = 3586 \text{ and}$$

- The fractional part can be expressed as

$$265 = 2\times10^{-1} + 6\times10^{-2} + 5\times10^{-3}$$

$$= 0.2+0.06+0.005$$

$$= 0.265$$

Binary numbers

- Bi = two
- The binary number system is a radix-2 number system with '0' and '1' as the two independent digits.
- All larger binary numbers are represented in terms of '0' and '1'.
- Example: (10011)2 is:

2^4	2^3	2^2	2^1	2^0
1	0	0	1	1

$$1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 19$$

Cont'd..

- Base = 2
  - 2 digits { 0, 1 }, called *b*inary dig*its* or "*bits*"
- Weights
  - Weight = *(Base)* ^{Position}
- Magnitude
  - Sum of "*Bit* x *Weight*"
- Formal Notation
- Groups of bits

4 bits = *Nibble*

8 bits = *Byte*

4	2	1		1/2	1/4
1	0	1	•	0	1
2	1	0		-1	-2

$1 * 2^2 + 0 * 2^1 + 1 * 2^0 + 0 * 2^{-1} + 1 * 2^{-}$

$= (5.25)_{10}$

$(101.01)_2$

1 0 1 1

1 1 0 0 0 1 0 1

Octal Number System

- Octal = eight
- The octal number system has a radix of 8 and therefore has eight distinct digits.
- All higher-order numbers are expressed as a combination of these on the same pattern as the one followed in the case of the binary and decimal number systems.
- therefore has 8 different digits or symbols. These are:-

  { 0, 1, 2, 3, 4, 5, 6, 7 }

Cont'd..

- Base = 8
  - 8 digits { 0, 1, 2, 3, 4, 5, 6, 7 }
- Weights
  - Weight = (*Base*) ^{Position}
- Magnitude
  - Sum of "*Digit* x *Weight*"
- Formal Notation

64	8	1		1/8	1/64
5	1	2	.	7	4
2	1	0		-1	-2

$$5 * 8^{2} + 1 * 8^{1} + 2 * 8^{0} + 7 * 8^{-1} + 4 * 8^{-2}$$

$$=(330.9375)_{10}$$

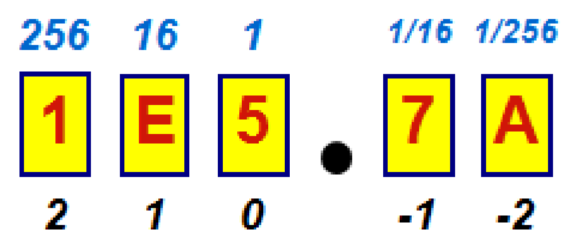
$$(512.74)_{8}$$

Hexadecimal numbers

- Hexadeci = sixteen
- The hexadecimal number system is a radix-16 number system and its 16 basic digits are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F.
- The decimal equivalent of A, B, C, D, E and F are 10, 11, 12, 13, 14 and 15 respectively, for obvious reasons.
- The hexadecimal number system provides a condensed way of representing large binary numbers stored and processed inside the computer.

Cont'd..

- Base = 16
  - 16 digits { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F }
- Weights
  - Weight = $(Base)^{Position}$
- Magnitude
  - Sum of "*Digit* x *Weight*"
- Formal Notation



$$1 \times 16^{2} + 14 \times 16^{1} + 5 \times 16^{0} + 7 \times 16^{-1} + 10 \times 16^{-2}$$

$$=(485.4765625)_{10}$$

$$(1E5.7A)_{16}$$

Cont'd..

Decimal	Binary	Octal	Hexa-decimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7

Decimal	Binary	Octal	Hexa-decimal
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Cont'd..

Decimal	Binary	Octal	Hexa-decimal
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14
21	10101	25	15
22	10110	26	16
23	10111	27	17

Cont'd..

System	Base	Symbols	Used by humans?	Used in computers?
Decimal	10	0, 1, … 9	Yes	No
Binary	2	0, 1	No	Yes
Octal	8	0, 1, … 7	No	No
Hexa-decimal	16	0, 1, … 9, A, B, … F	No	No

Summary Binary Number System

- Bit:- is an abbreviation of the term 'binary digit' and is the smallest unit of information. It is either '0' or '1'.
- A *byte:-* is a string of eight bits.
- The word length(word size):- may equal one byte, two bytes, four bytes or be even larger.
- A computer word is again a string of bits whose size, called the 'word length' or 'word size', is fixed for a specified computer, although it may vary from computer to computer.

Number Representation in Binary

- Different formats used for binary representation of both positive and negative decimal numbers include the sign-bit magnitude method, the 1's complement method and the 2's complement method.

1. Sign-Bit Magnitude

- In the sign-bit magnitude representation of positive and negative decimal numbers, the MSB represents the 'sign', with
  - ✓ '0' denoting a plus sign and a
  - ✓ '1' denoting a minus sign.
- In eight-bit representation, while MSB represents the sign, the remaining seven bits represent the magnitude.

Cont'd..

- Example:- the eight-bit representation of +9 would be 0,0001001, and that for −9 would be 1,0001001.
- Generally an n−bit binary representation can be used to represent decimal numbers in the range of $-(2^{n-1}-1)$ to $+(2^{n-1}-1)$.
- That is, eight-bit representation can be used to represent decimal numbers in the range from −127 to +127 using the sign-bit magnitude format

Cont'd..

2. 1's Complement

- The 1's complement of a binary number is obtained by complementing all its bits,
  - ✓ All '0's become '1's
  - ✓ All '1's become '0's

Example:- the 1's complement of $(10010110)_2$ is $(01101001)_2$.

- In the 1's complement format, the positive numbers remain unchanged(same as sign magnitude system).
- The negative numbers are obtained by taking the 1's complement of the positive counterparts.

Cont'd..

Example:- +9 will be represented as 0,0001001 in eight-bit notation, and −9 will be represented as 1,1110110, which is the 1's complement of 0,0001001.

- Generally Again, n-bit notation can be used to represent numbers in the range from $-(2^{n-1}-1)$ to $+(2^{n-1}-1)$ using the 1's complement format.

2. 2's Complement

- The 2's complement of a binary number is obtained by adding '1' to its 1's complement.

Cont'd..

- **Addition**
  - Decimal Addition

	1	1	← Carry
		5	5
+		5	5
	1	1	0

$= \mathit{Ten} \geq \mathit{Base}$

→ **Subtract a Base**

Cont'd..

- Binary Addition
- Binary addition follow the same patter , but

0 + 0	= 0 carry 0
0 + 1 = 1 + 0	= 1 carry 0
1 + 1	= 0 carry 1
1 + 1 + 1	= 1 carry 1

$$
\begin{array}{cccccccccc}
 & 1 & 1 & 1 & 1 & 1 & 1 & & & \\
 & & 1 & 1 & 1 & 1 & 0 & 1 & & = 61 \\
+ & & & 1 & 0 & 1 & 1 & 1 & & = 23 \\
\hline
 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & & = 84
\end{array}
$$

$\geq (2)_{10}$

Cont'd..

- Example:- The 2's complement of (1 0 1 1 0 0 0 0)

Solution

- Take 1's complement then add 1

```
  1 0 1 1 0 0 0 0
  0 1 0 0 1 1 1 1
+               1
-----------------
  0 1 0 1 0 0 0 0
```

Cont'd..

- In the 2's complement representation of binary numbers, the MSB represents the sign, with :-
  - ✓ '0' used for a plus sign and a
  - ✓ '1' used for a minus sign.
  - ✓ The remaining bits are used for representing magnitude
- Positive magnitudes are represented in the same way as in the case of sign-bit or 1's complement representation.
- Negative magnitudes are represented by the 2's complement of their positive counterparts.

Cont'd..

Example:- +9 would be represented as 0,0001001, and −9 would be written as 1,1110111

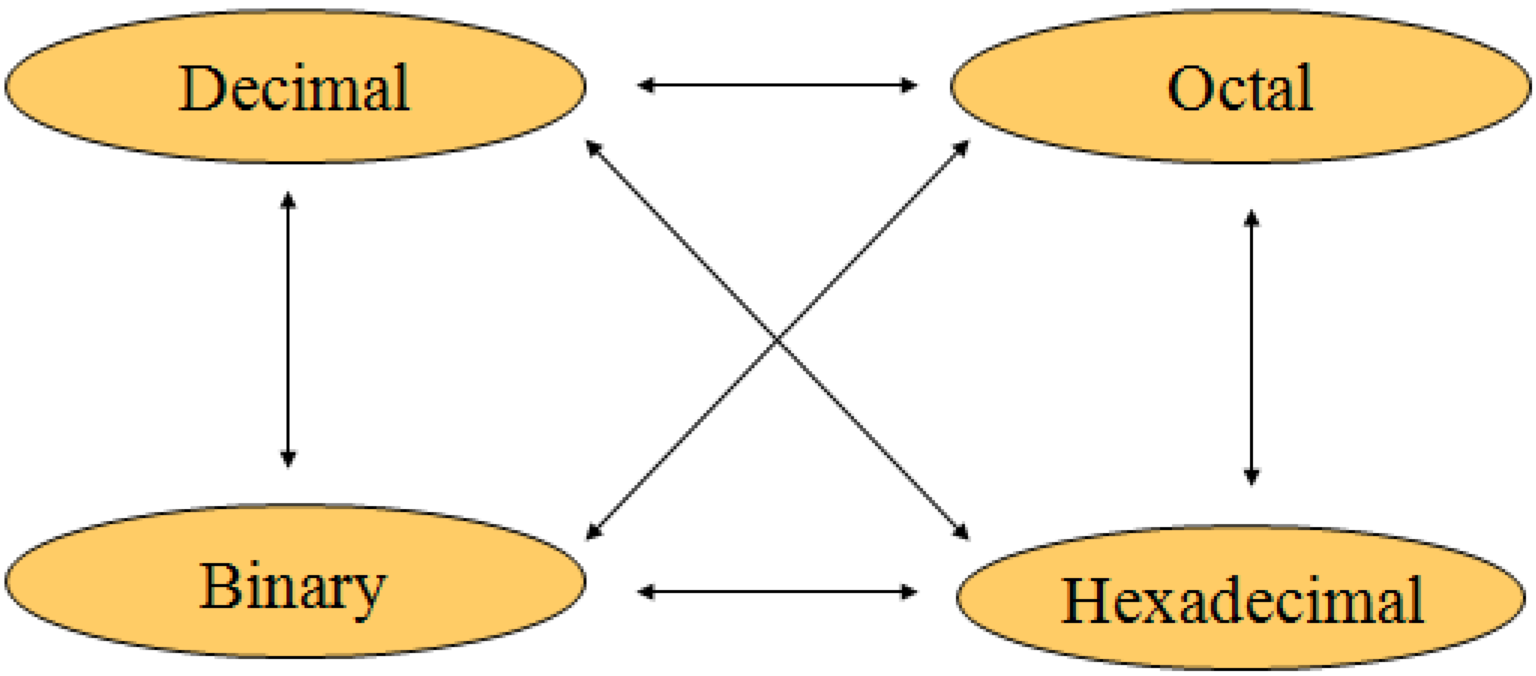
- Generally The n-bit notation of the 2's complement format can be used to represent all decimal numbers in the range from $+(2^{n-1}-1)$ to $-(2^{n-1})$.

Signed Binary Numbers

Table 1.3
Signed Binary Numbers

Decimal	Signed-2's Complement	Signed-1's Complement	Signed Magnitude
+7	0111	0111	0111
+6	0110	0110	0110
+5	0101	0101	0101
+4	0100	0100	0100
+3	0011	0011	0011
+2	0010	0010	0010
+1	0001	0001	0001
+0	0000	0000	0000
−0	—	1111	1000
−1	1111	1110	1001
−2	1110	1101	1010
−3	1101	1100	1011
−4	1100	1011	1100
−5	1011	1010	1101
−6	1010	1001	1110
−7	1001	1000	1111
−8	1000	—	—

Conversion Among Bases



Binary to Decimal

- Technique
  - Multiply each bit by 2^n, where n is the "weight" of the bit
  - The weight is the position of the bit, starting from 0 on the right
  - Add the results

Example

Bit "0"

$$101011_2 \Rightarrow \begin{array}{rcrcr} 1 & \times & 2^0 & = & 1 \\ 1 & \times & 2^1 & = & 2 \\ 0 & \times & 2^2 & = & 0 \\ 1 & \times & 2^3 & = & 8 \\ 0 & \times & 2^4 & = & 0 \\ 1 & \times & 2^5 & = & 32 \\ \hline & & & & 43_{10} \end{array}$$

Cont’d..

- Find the decimal equivalent of the following binary numbers expressed in the 2’s complement format:

  (a) 00001110;

  (b) 10001110.

Solution

(a) The MSB bit is ‘0’, which indicates a plus sign.

The magnitude bits are 0001110.

The decimal equivalent =

$$\begin{array}{lcr} 0 \times 2^0 = & & 0 \\ 1 \times 2^1 = & & 2 \\ 1 \times 2^2 = & & 4 \\ 1 \times 2^3 = & & 8 \\ 0 \times 2^4 = & & 0 \\ 0 \times 2^5 = & & 0 \\ 0 \times 2^6 = & & 0 \\ \hline & & +14_{10} \end{array}$$

Octal to Decimal

- Technique
  - Multiply each bit by 8^n, where n is the "weight" of the bit
  - The weight is the position of the bit, starting from 0 on the right
  - Add the results

Example

$$724_8 \Rightarrow$$

$$4 \times 8^0 = 4$$

$$2 \times 8^1 = 16$$

$$7 \times 8^2 = \underline{448}$$

$$468_{10}$$

Hexadecimal to Decimal

- Technique
  - Multiply each bit by 16^n, where n is the "weight" of the bit
  - The weight is the position of the bit, starting from 0 on the right
  - Add the results

Example

$$
\begin{aligned}
ABC_{16} \Rightarrow\ & C \times 16^0 = 12 \times 1 = 12 \\
& B \times 16^1 = 11 \times 16 = 176 \\
& A \times 16^2 = 10 \times 256 = 2560 \\
& \overline{2748_{10}}
\end{aligned}
$$

Decimal to Binary

Example:-find the binary equivalent of $(13.625)_{10}$.

Solution

- Technique

1. Integer
   - Divide the number by the 'Base' (=2)
   - Take the remainder (either 0 or 1) as a coefficient
   - Take the quotient and repeat the division
   - Etc.

Cont'd..

Example:-The integer $(13)_{10}$

Solution

	Quotient	Remainder	Coefficient
13/ 2 =	6	1	$a_0 = 1$
6 / 2 =	3	0	$a_1 = 0$
3 / 2 =	1	1	$a_2 = 1$
1 / 2 =	0	1	$a_3 = 1$

Answer: $(13)_{10} = (a_3\, a_2\, a_1\, a_0)_2 = (1101)_2$

MSB LSB

Cont'd..

- Technique

1. Fraction
    - Multiply the number by the 'Base' (=2)
    - Take the integer (either 0 or 1) as a coefficient
    - Take the resultant fraction and repeat the multiplication
    - Etc.

Cont'd..

Example:-The integer $(0.625)_{10}$

Solution

		Integer		Fraction	Coefficient
0.625	* 2 =	1	.	25	$a_{-1} = 1$
0.25	* 2 =	0	.	5	$a_{-2} = 0$
0.5	* 2 =	1	.	0	$a_{-3} = 1$

Answer: $(0.625)_{10} = (0.a_{-1}\, a_{-2}\, a_{-3})_2 = (0.101)_2$

MSB LSB

Octal to Binary

- Technique
  - Convert each octal digit to a 3-bit equivalent binary representation.

Example:-

$705_8 = ?_2$

7	0	5
111	000	101

$705_8 = 111000101_2$

Hexadecimal to Binary

- Technique
  - Convert each hexadecimal digit to a 4-bit equivalent binary representation.

Example:- find the binary equivalent of $(10AF)_{16}$

1	0	A	F
0001	0000	1010	1111

$$10AF_{16} = 0001000010101111_2$$

Decimal to Octal

Example: $(175.3125)_{10}$

- Technique

1. Integer

   - Divide by 8
   - Keep track of the remainder

	Quotient	Remainder	Coefficient
175 / 8 =	21	7	$a_0 = 7$
21 / 8 =	2	5	$a_1 = 5$
2 / 8 =	0	2	$a_2 = 2$

Answer: $(175)_{10} = (a_2\, a_1\, a_0)_8 = (257)_8$

Cont'd..

- Technique

1. Fraction
   - Multiply by 8
   - Take the integer as a coefficient
   - Take the resultant fraction and repeat the multiplication

	Integer		Fraction	Coefficient
0.3125 * 8 =	2	.	5	$a_{-1} = 2$
0.5 * 8 =	4	.	0	$a_{-2} = 4$

Answer: $(0.3125)_{10} = (0.a_{-1}\,a_{-2}\,a_{-3})_8 = (0.24)_8$

Decimal to Hexadecimal

- Example:-determine the hexadecimal equivalent of $(82.25)_{10}$
- Technique

1. Integer
   - Divide by 16
   - Keep track of the remainder

		Quotient	Remainder	Coefficient
82	/ 16 =	5	2	$a_0 = 2$
5	/ 16 =	0	5	$a_1 = 5$

Answer: $(82)_{10} = (a_1\, a_0)_{16} = (52)_{16}$

Cont'd;..

- Technique

1. Fraction

   - Multiply by 16
   - Take the integer as a coefficient
   - Take the resultant fraction and repeat the multiplication

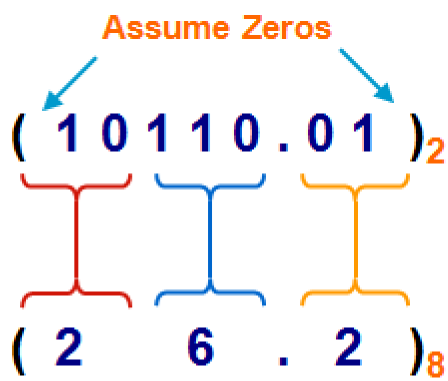
		Integer		Fraction	Coefficient
0.25	* 16 =	4	.	0	$a_{-1} = 4$
0.0	* 16 =	0	.	0	$a_{-2} = 0$

Answer: $(0.25)_{10} = (0.a_{-1}\, a_{-2})_{16} = (0.40)_{16}$

Binary to Octal

- Technique
  - $8 = 2^3$
  - Group bits in threes, starting on right
  - Convert to octal digits

Example:



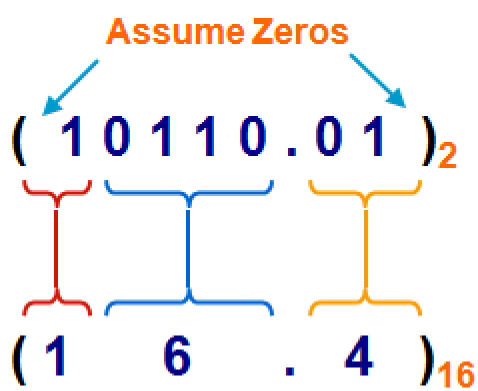
Works **both** **ways (*Binary* to *Octal* & *Octal* to *Binary*)**

Octal	Binary
0	0 0 0
1	0 0 1
2	0 1 0
3	0 1 1
4	1 0 0
5	1 0 1
6	1 1 0
7	1 1 1

Binary to Hexadecimal

- Technique
  - $16 = 2^4$
  - Group bits in fours, starting on right
  - Convert to hexadecimal digits

Example:



Works **both** **ways (*****Binary*** **to** ***Hex*** **&** ***Hex*** **to** ***Binary*****)**

Hex	Binary
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1
A	1 0 1 0
B	1 0 1 1
C	1 1 0 0
D	1 1 0 1
E	1 1 1 0
F	1 1 1 1

Octal to Hexadecimal

- Technique
  - Use binary as an intermediary

Example:

$$(2\ 6\ .\ 2)_8$$

Assume Zeros $(0\,1\,0\,1\,1\,0\,.\,0\,1\,0)_2$ Assume Zeros

$$(1\ 6\ .\ 4)_{16}$$

Works both ways (*Octal* to *Hex* & *Hex* to *Octal*)

Hexadecimal to Octal

- Technique
  - Use binary as an intermediary
  - Example:-

$$(1\quad 6)_{16}$$

$$(0\,0\,0\,1\,0\,1\,1\,0)_2$$

$$(2\quad 6)_8$$

Exercise – Convert...

Decimal	Binary	Octal	Hexa-decimal
33			
	1110101		
		703	
			1AF

Solution

Decimal	Binary	Octal	Hexa-decimal
33	100001	41	21
117	1110101	165	75
451	111000011	703	1C3
431	110101111	657	1AF

Multiplication

- Binary, two 1-bit values

A	B	$A \times B$
0	0	0
0	1	0
1	0	0
1	1	1

Complementary Number systems

- Complementary numbers form the basis of Complementary arithmetic, which is a powerful methods used in digital system for handling mathematical operation on signed numbers.
- Most digital computers use a complementary number system to minimize the amount of circuitry needed to perform integer arithmetic.
- For example A –B can be computing A+(-B)
- Radix complement and diminished radix complement are important number systems

Radix complement Arithmetic

- The Radix Complement [N]r of a number (N)r is defined as

  $[N]_r = r^n - (N)r$, where n-the number of digit in (N)r and

  r- is base

- Example 1.0 Determine the 2's complement of $(N)_2 = (01100101)_2$

  $[N]_2 = [01100101]2$

  $= 2^8 - (01100101)_2 = (100000000)_2 - (01100101)_2$

  $= (10011011)_2$

- Example 2.0

  Given the two binary numbers X = 1010100 and Y = 1000011, perform the subtraction (a) $X - Y$; and (b) $Y - X$, by using 2's complement.

Cont'd..

(a)	$X =$	1010100
	2's complement of $Y =$	$+0111101$
	Sum $=$	10010001
	Discard end carry $2^7 =$	-10000000
	Answer. $X - Y =$	0010001

(b)	$Y =$	1000011
	2's complement of $X =$	$+\ 0101100$
	Sum $=$	1101111

There is no end carry. Therefore, the answer is Y – X = – (2's complement of 1101111) = – 0010001.

Diminished Radix complement Arithmetic

- **Diminished Radix Complement - (r-1)'s Complement**
  - Given a number N in base r having n digits,
  - The (r–1)'s complement of N is defined as:

$$[N]_{r-1} = r^n - [N]_r - 1$$

Example 1.0 Determine the 1's complement of $(01100101)_2$

Solution

$$[N]_{2-1} = 2^8 - (01100101)_2 - (00000001)_2$$

$$= (10011011)_2 - (00000001)_2$$

$$= (10011010)_2$$

Cont'd..

Example 2.0

Repeat Example 1.0, but this time using 1's complement

(a) $X - Y = 1010100 - 1000011$

$$
\begin{array}{rr}
X = & 1010100 \\
\text{1's complement of } Y = & \pm\, 0111100 \\
\text{Sum} = & 10010000 \\
\text{End-around carry} = & \underline{+ \qquad 1} \\
\text{Answer. } X - Y = & 0010001
\end{array}
$$

(b) $Y - X = 1000011 - 1010100$

$$
\begin{array}{rr}
Y = & 1000011 \\
\text{1's complement of } X = & \underline{+\, 0101011} \\
\text{Sum} = & 1101110
\end{array}
$$

There is no end carry, Therefore, the answer is Y – X = – (1's complement of 1101110) = – 0010001.

Codes

1. BCD Code

- A number with k decimal digits will require 4k bits in BCD.
- Decimal 396 is represented in BCD with 12bits as 0011 1001 0110, with each group of 4 bits representing one decimal digit.
- A decimal number in BCD is the same as its equivalent binary number only when the number is between 0 and 9.
- The binary combinations 1010 through 1111 are not used and have no meaning in BCD.

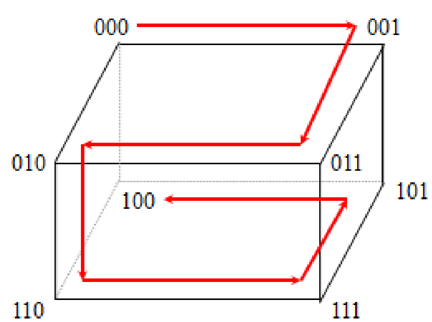
Binary-Coded Decimal (BCD)

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

Cont'd..

2. Gray Code

- The advantage is that only bit in the code group changes in going from one number to the next.
    - Error detection.
    - Representation of analog data.
    - Low power design.



Gray Code

Gray Code	Decimal Equivalent
0000	0
0001	1
0011	2
0010	3
0110	4
0111	5
0101	6
0100	7
1100	8
1101	9
1111	10
1110	11
1010	12
1011	13
1001	14
1000	15

Cont'd..

- Binary-to-Gray code conversion
  - The following rules explain how to convert from binary number to a Gray code :-

    1. The MSB (left most) in the Gray code is the same as the corresponding MSB in the binary number.
    2. Going from left to right, add each adjacent pair of binary code bits to get the next Gray code bit. Discard carries
  - For example , the conversion of the binary number 10110 to gray code is as follows

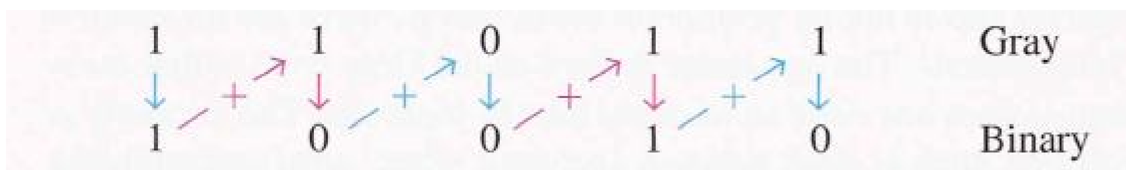
$$
\begin{array}{ccccccccccc}
1 & -+\rightarrow & 0 & -+\rightarrow & 1 & -+\rightarrow & 1 & -+\rightarrow & 0 & & \text{Binary} \\
\downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \\
1 & & 1 & & 1 & & 0 & & 1 & & \text{Gray}
\end{array}
$$

The Gray code is 11101.

Cont'd..

- Gray code - to-Binary conversion
  - The following rules explain how to convert from Gray code to a binary number :-

    1. The MSB (left most) in the binary number is the same as the corresponding MSB in the Gray code.
    2. Add each binary code bit generated to the Gray code bit in the next adjacent position. Discard carries
  - For example , the conversion of the Gray code 11011 to binary number is as follows



The binary number is 10010.

Cont'd..

3. American Standard Code for Information Interchange (ASCII)

   Character Code

American Standard Code for Information Interchange (ASCII)

	$b_7b_6b_5$							
$b_4b_3b_2b_1$	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	|
1101	CR	GS	−	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	—	o	DEL

Cont'd..

Control characters

NUL	Null	DLE	Data-link escape
SOH	Start of heading	DC1	Device control 1
STX	Start of text	DC2	Device control 2
ETX	End of text	DC3	Device control 3
EOT	End of transmission	DC4	Device control 4
ENQ	Enquiry	NAK	Negative acknowledge
ACK	Acknowledge	SYN	Synchronous idle
BEL	Bell	ETB	End-of-transmission block
BS	Backspace	CAN	Cancel
HT	Horizontal tab	EM	End of medium
LF	Line feed	SUB	Substitute
VT	Vertical tab	ESC	Escape
FF	Form feed	FS	File separator
CR	Carriage return	GS	Group separator
SO	Shift out	RS	Record separator
SI	Shift in	US	Unit separator
SP	Space	DEL	Delete