Department of Biomedical Engineering
Digital Electronic / Third stage
Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

Lecture One

Number system, operations, and codes

A **numbering system** is a mathematical framework for representing and organizing numbers using a specific set of symbols and rules. It defines how numbers are expressed and manipulated within a given base or radix, determining the number of unique symbols representing values. Numbering systems are essential for various fields, including mathematics, engineering, and computer science, as they provide a structured way to handle numerical data.

Common Numbering Systems:

1. **Decimal (Base 10):** The most widely used system, with digits 0–9.
   - **Example**: 123_9 (decimal).
2. **Binary (Base 2):** Used in computing and digital systems, with digits 0 and 1.
   - **Example**: 1010_2 (binary).
3. **Octal (Base 8):** Used in computing, with digits 0 –7.
   - **Example**: 74_8 (octal).
4. **Hexadecimal (Base 16):** Used in computing and programming, with digits 0–9 and letters A–F.
   - Example: $1F3_{16}$ (hexadecimal).

Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

1- BINARY NUMBERS

The binary number system uses only two symbols (0, l). It is said to have a radix of 2 and is commonly called the **base 2** number system. Each binary digit is called a bit.

Counting in binary is illustrated in Fig. 1. The binary number is shown on the right with its decimal equivalent. Notice that the least significant bit (LSB) is the 1s place.

Decimal count	Binary count				
	16s	8s	4s	2s	1s
0					**0**
1				**0**	**1**
2				**1**	**0**
3				**1**	**1**
4			**1**	**0**	**0**
5			**1**	**0**	**1**
6			**1**	**1**	**0**
7			**1**	**1**	**1**
8		**1**	**0**	**0**	**0**
9		**1**	**0**	**0**	**1**
10		**1**	**0**	**1**	**0**
11		**1**	**0**	**1**	**1**
12		**1**	**1**	**0**	**0**
13		**1**	**1**	**0**	**1**
14		**1**	**1**	**1**	**0**
15		**1**	**1**	**1**	**1**
16	**1**	**0**	**0**	**0**	**0**
	2^4	2^3	2^2	2^1	2^0
	Power of 2				

Fig. 1 Counting in binary and decimal

Consider the number shown in Fig. 2. This figure shows how to convert the binary 10011 to its decimal equivalent. Note that, for each 1 bit in the binary number, the decimal equivalent for that place value is written below. The decimal numbers are then added (16 + 2 + 1 = 19) to yield the decimal equivalent. Binary 10011 then equals a decimal 19.

Power of 2	2^4	2^3	2^2	2^1	2^0	
Place value	**16s**	**8s**	**4s**	**2s**	**1s**	
Binary	1	0	0	1	1	
Decimal	16	0	0	2	1	= 19

Fig. 2 Binary to decimal conversion

How about converting fractional numbers? Figure 3 illustrates the binary number 1110.101 being converted to its decimal equivalent. The place values are given across the top. The place value of each 1 bit in the binary number is added to form the decimal number. In this problem 8 + 4 + 2 + 0.5 + 0.125 = 14.625 in decimal

Power of 2	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}
Place value	**8s**	**4s**	**2s**	**1s**		**0.5s**	**0.25s**	**0.125s**
Binary	1	1	1	0	.	1	0	1
Decimal	8	4	2	0	.	0.5	0	0.125 = 14.625

Department of Biomedical Engineering
Digital Electronic / Third stage
Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

Converting decimal to binary

The decimal 87 is first divided by 2, leaving 43 with a remainder of 1. The remainder is important and is recorded at the right. It becomes the **LSB** in the binary number. The quotient (43) is then transferred as shown by the arrow and becomes the dividend. The quotients are repeatedly divided by 2 until the quotient becomes 0 with a remainder of 1, as in the last line of Fig. 3. Near the bottom, the figure shows that decimal 87 equals binary 1010111.

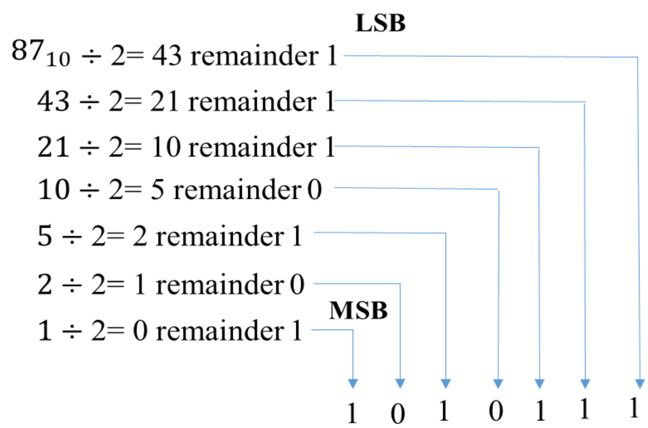


Fig. 3 Decimal-to-binary conversion

Convert the decimal number 0.375 to a binary number. Figure 5 illustrates one method of performing this task. Note that the decimal number (0.375) is being multiplied by 2. When the product is 1.00, the conversion process is complete. Figure 4 shows a decimal 0.375 being converted into a binary equivalent of 0.011.

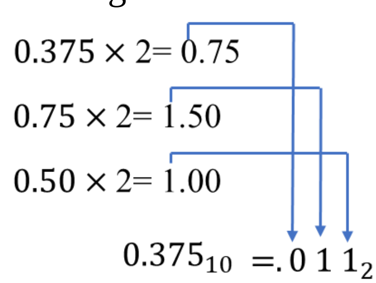


Fig. 4 Fractional decimal-to-binary conversions

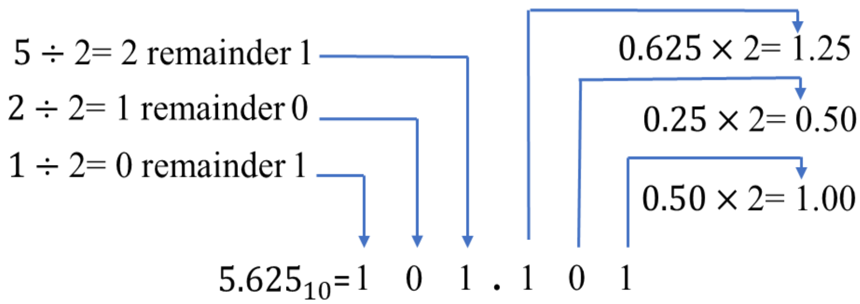


Fig. 5 Decimal-to-binary conversion

Q/ Convert the following binary numbers to their decimal equivalents:

(a) 001100 **(b)** 000011 **(c)** 011100 **(d)** 111100

(e) 101010 **(f)** 111111 **(g)** 100001 **(h)** 111000

Solution:

(a) 001100 = 12 **(b)** 000011= 3 **(c)** 011100 = 28 **(d)** 111100 = 60
(e) 101010, = 42 **(f)** 111111= 63 **(g)** 100001 = 33 **(h)** 111000 = 56

Q/ Convert the binary number 11100.011 to decimal

Solution:

11100.011 = 28.375

Q/ Convert the following decimal numbers to their binary equivalents:

(a) 64 **(b)** 100 **(c)** 111 **(d)** 145 **(e)** 255 **(f)** 500 **(g)** 34.7510

Solution

(a) 64 = 1000000 **(b)** 100 = 1100100 **(c)** 111= 1101111 **(d)** 145 = 10010001

(e) 255 = 11111111 **(f)** 500 = 111110100 **(g)** 34.75= 1000l0.11

Octal-to-Decimal Conversion:

Each digit of the octal number represents a value from 0 to 7. To convert an **octal number** to a **decimal number,** each digit of the octal number is multiplied by 8^n, where n is the position of the digit (counting from right to left, starting from 0). Then, sum all these values.

Example (1): convert (345_8) to its decimal equivalent.

$345_8 = (3\times8^2) + (4\times8^1) + (5\times8^0) = 192+32+5=229_{10}$

Example (2): convert (127_8) to its decimal equivalent

$127^8 = (1\times8^2) +(2\times8^1) +(7\times8^0) = 64+16+7=87_{10}$

Decimal-to-Octal Conversion:

A decimal integer can be converted to octal using the same repeated division method used in the decimal-to-binary conversion but with a division factor of 8 instead of 2. An example is shown below:

Example 1: Convert 229_{10} to octal

$229 \div 8 = 28$ remainder 5

$28 \div 8 = 3$ remainder 4

$3 \div 8 = 0$ remainder 3

$229_{10} = 3\ 4\ 5_8 =$

Example 2: Convert 87_{10} to octal

$87 \div 8 = 10$ remainder 7

$10 \div 8 = 1$ remainder 2

$1 \div 8 = 0$ remainder 1

$87_{10} = 1\ 2\ 7_8 =$

Octal-to-Binary Conversion:

To convert octal to binary, replace each octal digit with its 3-bit binary equivalent

Example 1: Convert 345_8 to binary

Solution

1- Break the octal number into individual digits: 3, 4, 5.

2- Convert each digit to binary: 3 = 011
4 = 100
5 = 101

345_8 = 011100101

Binary-to-Octal Conversion:

To convert binary to octal by grouping the binary digits into groups of three bits starting from the rightmost digit or LSB. If the leftmost group has fewer than three digits, pad it with zeros. Replace each 3-bit binary group with its equivalent octal value.

Example 2: Convert 101011_2 to octal

Solution: $101\ 011_2$
5 3

Example 3: convert 11010110_2 to its octal equivalent

Solution: 011 010 110
3 2 6

Hexadecimal Number

The hexadecimal number system has a radix of 16. It is referred to as the base 16 number system. It uses the symbols 0-9, A, B, C. D. E, and F. Where the character A=10, B=11, C=12, D=13, E=14, F=15 in decimal number.

Decimal	Binary	Octal	Hexadecimal
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
4	1110	16	E
15	1111	17	F

Hexadecimal-to-decimal conversion

Example 1: Convert the hexadecimal number 2B6 to **a** decimal number.

Power of 16	16^2	16^1	16^0
Place value	256	16	1

Hexadecimal number 2 B 6

Decimal number 2 ×256 11×16 6×1

= 512 + 176 + 6 = **694_{10}**

Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

Example 2: Convert the hexadecimal number A3F.C to **a** decimal number.

Power of 16	16^2	16^1	16^0	16^{-1}
Place value	256	16	1	0.0625

Hexadecimal number	A	3	F	.	C
Decimal number	10 ×256	3×16	15×1		12×0.0625

= 2560 + 48 + 15 + 0.75

$= 2623.75_{10}$

Decimal-to-hexadecimal conversion

Example 1: Convert the decimal number 45 to its hexadecimal equivalent.

45 ÷ 16 = 2 remainder 13

2 ÷ 16= 0 remainder 2

$\mathbf{45_{10}} = \mathbf{2\ D_{16}}$

Example 2: Convert the decimal number 250.25 to its hexadecimal equivalent

250 ÷ 16 = 15 remainder 10

15 ÷ 16 = 0 remainder 15

0.25 × 16= 4.00

250.25_{10}= F A . 4

Hexadecimal-to-binary conversion

Each hexadecimal number is converted to its 4-bit binary equivalent then combined to form the binary number.

Example 1: Convert the hexadecimal number **3B9** to its binary equivalent

Solution :

3	B	9_{16}		
↓	↓	↓		
0011	1011	1001	=	001110111001

Example 2: Convert the hexadecimal number **47. FE** to its binary equivalent

Solution :

4	7	.	F	E		
↓	↓		↓	↓		
0100	0111		1111	1110	=	01000111.11111110

Binary-to- hexadecimal conversion

Example 1: Convert the binary number 10010.011011 to its hexadecimal equivalent

Solution:

0001	**0010**	**.**	**0110**	**1100**
↓	↓		↓	↓
1	2	.	6	C

The binary number 10010.011011 then equals $12.6C_{16}$

Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

Arithmetic Operations and Codes

1- Binary Arithmetic

Binary arithmetic is essential in all digital computers and other digital systems. To understand digital systems, you must know the basics of binary addition, subtraction, multiplication, and division.

a- Binary Addition

The four basic rules of addition binary digits are as follows:

0 + 0 = 0 Sum of 0 with a carry of 0
0 + 1 = 1 Sum of 1 with a carry of 0
1 + 0 = 1 Sum of 1 with a carry of 0
1 + 1 = 10 Sum of 0 with a carry of 1

Example 1

Add the following binary numbers:

(a) 11 + 11 *(b) 100 + 10* *(c) 111 + 11* *(d) 110 + 100*

Solution

	Binary	Decimal
(a)	11 + 11 = 110	3 + 3 = 6
(b)	100 + 10 = 110	4 + 2 = 6
(c)	111 + 11 = 1010	7 + 3 = 10
(d)	110 + 100 = 1010	6 + 4 = 10

Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

b- **Binary Subtraction**

The four basic rules of Subtraction of binary digits are as follows:

0 - 0 = 0
1 - 1 = 0
1 - 0 = 1
10 – 1 = 1   0 – 1 with a borrow of 1

Example (1)

Perform the following subtraction

(a) 11 – 01   (b) 11 – 10

Solution

(a)
$$\begin{array}{r} 11 \\ -\ 01 \\ \hline 10 \end{array} \qquad \begin{array}{r} 3 \\ -\ 1 \\ \hline 2 \end{array}$$

(b)
$$\begin{array}{r} 11 \\ -\ 10 \\ \hline 01 \end{array} \qquad \begin{array}{r} 3 \\ -\ 2 \\ \hline 1 \end{array}$$

Example (2)

Subtract 011 from 101

Solution

$$\begin{array}{r} 101 \\ -\ 011 \\ \hline 010 \end{array} \qquad \begin{array}{r} 5 \\ -\ 3 \\ \hline 2 \end{array}$$

c- **Binary Multiplication**

The four basic rules of multiplying bits are as follows:

0 × 0 = 0
0 × 1 = 0
1 × 0 = 0
1 × 1 = 1

Example (1)

Perform the following binary multiplications

(a) 11 × 11 (b) 101 × 111

Solution

(a)

```
   11      3
 × 11    × 3
 ----    ---
   11      9
 + 11
 ----
 1001
```

(b)

```
                 111      7
               × 101    × 5
               -----    ---
Partial  ⎧       111     35
products ⎨      000
         ⎩ + 111
           ------
           100011
```

d- Binary Division

Example (1)

Perform the following binary Division

(a) 110 ÷ 11 (b) 110 ÷ 10 (c) 1001011 ÷ 11

Solution

```
          10      2                 11      3
(a)  11)110    3)6      (b)  10)110    2)6
         11      6                10      6
        ---     --               ---     --
        000      0                10      0
                                  10
                                 ---
                                  00
```

(c)

```
       1 1 0 0 1
    -------------
1 1 ) 1 0 0 1 0 1 1
    - 1 1
      ---
        1 1
      - 1 1
        ---
        0 0 0
      -     0
          ---
            0 1
          -   0
            ---
              1 1
            - 1 1
              ---
                0
```

Complements of Binary Numbers

The 1's complement and the 2's complement of a binary number are important because they permit negative number representation. The method of 2's complement arithmetic is commonly used in computers to handle negative numbers.

Finding the 1's Complement

The 1's complement of binary number is found by changing all 1s to 0s and all 0s to 1's as illustrated below:

10110010
↓↓↓↓↓↓↓↓
01001101

Finding the second complement

The 2's complement of a binary number is found by adding 1 to the LSB of the 1's complement.

$$2\text{'s complement} = (1\text{'s complement}) + 1$$

Find the 2's complement of 10110010

Solution

10110010	Binary number
↓	
01001101	1's complement
+ 1	**add 1**
01001110	**2's complement**

Note:

The left most bit in a signed binary number is the sign bit, which tells you whether the number is positive or negative.

A 0 sign bit indicates a positive number, and a 1 sign bit indicate a negative number.

Hexadecimal Addition

When adding two hexadecimal numbers, use the following rules:

1- In any given column of an addition problem, think of the two hexadecimal digits in terms of their decimal value. For instance, $5_{16} = 5_{10}$ and $C_{16} = 12_{10}$.

2- If the sum of these two digits is 15_{10} or less, bring down the corresponding hexadecimal digit.

3- If the sum of these two digits is greater than 15_{10}, bring down the amount of the sum that exceeds 16_{10} and carry a 1 to the next column.

Example

Add the following hexadecimal numbers:

(a) $23_{16} + 16_{16}$ (b) $58_{16} + 22_{16}$ (c) $2B_{16} + 84_{16}$ (d) $DF_{16} + AC_{16}$

Solution

(a) $23_{16} + 16_{16} = \mathbf{39_{16}}$

right column: $3_{16} + 6_{16} = 3_{10} + 6_{10} = 9_{10} = 9_{16}$
left column: $2_{16} + 1_{16} = 2_{10} + 1_{10} = 3_{10} = 3_{16}$

(b) $58_{16} + 22_{16} = \mathbf{7A_{16}}$

right column: $8_{16} + 2_{16} = 8_{10} + 2_{10} = 10_{10} = A_{16}$
left column: $5_{16} + 2_{16} = 5_{10} + 2_{10} = 7_{10} = 7_{16}$

(c) $2B_{16} + 84_{16} = \mathbf{AF_{16}}$

right column: $B_{16} + 4_{16} = 11_{10} + 4_{10} = 15_{10} = F_{16}$
left column: $2_{16} + 8_{16} = 2_{10} + 8_{10} = 10_{10} = A_{16}$

(d) $DF_{16} + AC_{16} = \mathbf{18B_{16}}$

right column: $F_{16} + C_{16} = 15_{10} + 12_{10} = 27_{10}$
$27_{10} - 16_{10} = 11_{10} = B_{16}$ with a 1 carry

left column: $D_{16} + A_{16} + 1_{16} = 13_{10} + 10_{10} + 1_{10} = 24_{10}$
$24_{10} - 16_{10} = 8_{10} = 8_{16}$ with a 1 carry

Hexadecimal Subtraction

There are several methods to subtract hexadecimal, one of the most common methods is converting the hexadecimal number to binary. Take the 2's complement of the binary number. Convert the result to hexadecimal.

Example

Subtract the following hexadecimal numbers:

(a) $84_{16} - 2A_{16}$ (b) $C3_{16} - 0B_{16}$

Solution

(a) 2A = 00101010 Binary number

↓

11010101 1's complement

\+ **1** **add 1**

11010110 = D6 **2's complement**

84

+ D6

~~1~~5A Drop carry

The difference is $5A_{16}$

(b) 0B = 00001011 Binary number

11110100 1's complement

\+ **1** **add 1**

11110101 = F5 **2's complement**

C3

+ F5 **add**

~~1~~B8 Drop carry

The difference is $B8_{16}$

Department of Biomedical Engineering
Digital Electronic / Third stage
Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

Binary Coded Decimal (BCD)

Binary coded decimal (BCD) is a way to express each of the decimal digits with a binary code. There are only ten code group in the BCD system, so it is very easy to convert between decimal and BCD. Because we like to read and write in decimal, the BCD code provides an excellent interface to a binary system.

The 8421 BCD code

The 8421 code is a type of BCD code. Binary coded decimal means that each decimal digit, 0 through 9, is represented by a binary code of four bits. The designation 8421 indicates the binary weight of the four bits (2^3, 2^2, 2^1, 2^0)

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

Invalid codes

In the 8421 code, only ten of (0000 through 1111) are used. The six code combinations that are not used are 1010, 1011, 1100, 1101, 1110, and 1111 which are invalid in the 8421 BCD code.

Example: Convert each of the following decimal numbers to BCD

(a) 35 (b) 98 (c) 170 (d) 2469

Solution

(a) 3 5 → **00110101**

(b) 9 8 → **10011000**

(c) 1 7 0 → **000101110000**

(d) 2 4 6 9 → **0010010001101001**

Example 2

Convert each of the following BCD codes to decimal

(a) 10000110 (b) 001101010001 (c) 1001010001110000

Solution

(a) 1000 0110 → 8 6

(b) 0011 0101 0001 → 3 5 1

(c) 1001 0100 0111 0000 → 9 4 7 0

BCD addition

BCD is a numerical code and can be used in arithmetic operations.
The steps of adding two BCD numbers:

1- Add the two BCD numbers using the role of binary.
2- If a 4-bit sum is equal to or less than 9, it is a valid BCD number.
3- If a 4-bit sum is greater than 9, or if a carry-out of the 4-bit group is generated, it is an invalid result. Add 6 (0110) to the 4-bit sum to skip the six invalid states and return the code to 8421.

Example1

Add the following BCD numbers:

(a) 0011 + 0100 (b) 00100011 + 00010101

(c) 10000110 + 00010011 (d) 01001010000 + 010000010111

Solution

(a)
```
  0011        3
+ 0100      + 4
  ----      ---
  0111        7
```

(b)
```
  0010 0011      23
+ 0001 0101    + 15
  ---------    ----
  0011 1000      38
```

(c)
```
  1000 0110      86
+ 0001 0011    + 13
  ---------    ----
  1001 1001      99
```

(d)
```
  0100 0101 0000      450
+ 0100 0001 0111    + 417
  --------------    -----
  1000 0110 0111      867
```

Example:

Add the following BCD numbers:

(a) 1001 + 0100 **(b)** 1001 + 1001

(c) 00010110 + 00010101 **(d)** 01100111 + 01010011

Solution

The decimal number additions are shown for comparison.

(a)

	1001		9
	+ 0100		+4
	1101	Invalid BCD number (>9)	13
	+ 0110	Add 6	
0001	**0011**	Valid BCD number	
↓	↓		
1	3		

(b)

	1001		9
	+ 1001		+ 9
1	0010	Invalid because of carry	18
	+ 0110	Add 6	
0001	**1000**	Valid BCD number	
↓	↓		
1	8		

(c)

0001	0110		16
+ 0001	0101		+ 15
0010	1011	Right group is invalid (>9), left group is valid.	31
	+ 0110	Add 6 to invalid code. Add carry, 0001, to next group.	
0011	**0001**	Valid BCD number	
↓	↓		
3	1		

(d)

	0110	0111		67
	+ 0101	0011		+ 53
	1011	1010	Both groups are invalid (>9)	120
	+ 0110	+ 0110	Add 6 to both groups	
0001	**0010**	**0000**	Valid BCD number	
↓	↓	↓		
1	2	0		

Department of Biomedical Engineering
Digital Electronic / Third stage
Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

Gray code

Binary to Gray code conversion

There are two steps in conversion from binary to gray code

1- The most significant bit (left-most) in the Gray code is the same as the corresponding MSB in the binary.

2- Going from left to right, add each adjacent pair of binary code bits to get the next Gray code bit. Discard the carries.

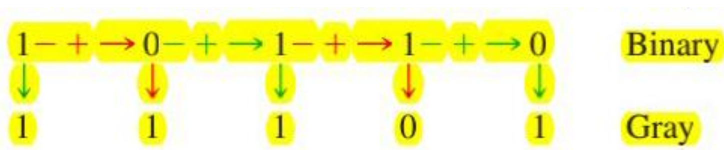
Example 1

(a) Convert the binary number 10110 to Gray code

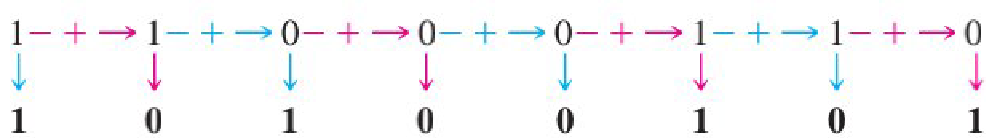
(b) Convert the binary number 11000110 to Gray code

Solution

(a)



(b)



Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

Gray to binary code conversion

There are two steps in conversion from gray code to binary

1- The most significant bit (left-most) in the binary code is the same as the corresponding bit in the Gray code.

2- Add each binary code bit generated to the Gray code bit in the next adjacent position. Discard carries.

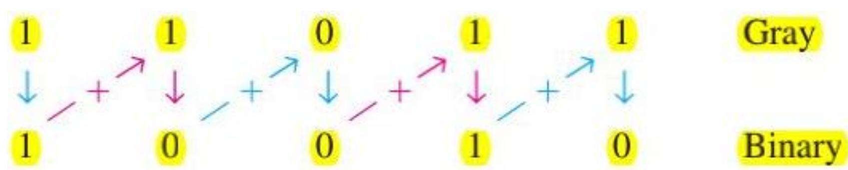
Example

(a) Convert the Gray code 11011 to binary

(b) Convert the Gray code 10101111 to binary

Solution

(a)



(b) Gray code to binary: