TOPIC 2 – REPRESENTATION OF DATA

LEARNING OUTCOMES

By the end of this topics, you will be able to:

1. describe data organisations and numbering system.
2. convert numbers between number bases.
3. use binary numbers, hexadecimal numbers and arithmetic.
4. use fraction and mixed number conversions.

INTRODUCTION

This topic explains about data representation methods used to represent information stored in a computer. There are different types of data stored in the computer such as numbers, letters, sounds and images.

2.1 DATA ORGANISATIONS – BITS, NIBBLES, BYTES, WORDS

The computer uses numeric codes to represent the data. The information is stored in a series of zeroes and ones which is known as binary code. Basically, data is transmitted within a computer by electrical signals that represented by the values on = 1 or off =0. The length of the number is defined by these binary values.

Bits, nibbles, bytes and words are the common bit-lengths of binary. The bits consisted of *1* or *0* in a binary number. it is also known as digit or flag. Then a nibble or nybble is formed by a group of 4-bits and 8-bits makes a byte which also known as octet or character. Then a word is a double byte, 16-bits. The words can be extended to double word or long word which formed by a group of 32-bits and very long word, 64-bits.

It can be summarized as:

Table 2.8

Length	Name	Example
1	Bit/Digit/Flag	1 or 0
4	Nibble/Nybble	1100
8	Byte/Octet/Character	11110000
16	Word/Double Byte	1010101010101010
32	Double Word/Long Word	32 binary characters
64	Very Long Word	64 binary characters

SELF CHECK 2.1

1. What are bits, nibbles, bytes and words?
2. List the example of bits and bytes.

2.2 NUMBERING SYSTEM – DECIMAL, BINARY, HEXADECIMAL

2.2.1 DECIMAL NUMBERS

Decimal numbers are represented with a dot in between them which is called a decimal point. The numbers to left of the decimal point are the integers or whole numbers and 10 times bigger. The numbers to the right of the decimal point are called decimal numbers. The ones place to the right is 10 times smaller.

2.2.2 BINARY NUMBERS

Binary numbers are represented by zero (0) and one (1). A single binary digit is referred to as a bit. It is expressed the base-2 numeral system. For example, $(1010)_{2.}$ The binary number also consists of several bits such as 5-bit binary number (10100), 3-bit binary number $(100)_2$ or 6-bit binary number $(100000)_{2.}$

2.2.3 HEXADECIMAL NUMBERS

In the hexadecimal numbers, the numbers will be represented as the same as the decimal digits up to 9 however there are letters A, B, C, D, E, and F in place of decimal numbers 10. For example, 4B5.

SELF CHECK 2.2

1. Define the following numbering systems:
   - Decimal numbers
   - Binary numbers
   - Hexadecimal numbers

2.3 NUMERIC CONVERSION BETWEEN NUMBER BASES

2.3.1 CONVERSION DECIMAL NUMBERS TO BINARY NUMBERS WITH THE REMAINDER METHOD

Now, let's do the conversion decimal number to binary number. Given the decimal number 23_{10} and convert to Binary number.

Let's start by converting the integer portion then the fraction section.

1. First, take the integer 23, then create 3 columns for division, integer and remainder.
2. In the first row, start with 23 divides by 2 (23/2) then write the integer (11) and remainder (1) accordingly.

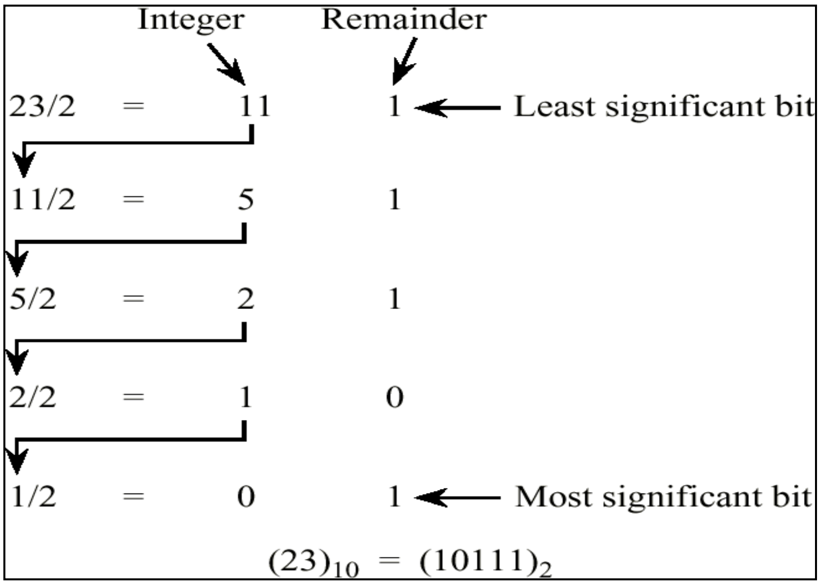

Figure 2.9: Conversion Decimal Number to Binary Number

3. Next, carry the first integer (11) to the second row and write the remainder in the remainder column.
4. Do the same steps until the integer equal to zero (0) then stop the division.
5. Then, refer to the remainder column for the binary numbers.
6. Write the answer from the most significant bit (the last remainder) to the least significant bit (the first remainder).
7. Then, that will be the binary number $(10111)_2$.

The decimal number also can be converted by look to the weight of each digit from right to the left. It increases by a factor of 10 while in the binary number system, the weight of each digit increases by a factor of 2 as shown in Table 2. So, the first digit has a weight of 1 that represented by the (2^0). Then follow by the second digit 2 (2^1), the third digit 4 (4^2), the forth digit 9 (2^3) and so on.

Table 2.10

MSB	**Binary Digit**							**LSB**
2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
256	128	64	32	16	8	4	2	1

2.3.2 CONVERSION BINARY NUMBERS TO DECIMAL NUMBERS

Table 2.11

Decimal Digit Value	256	128	64	32	16	8	4	2	1
Binary Digit Value	1	0	1	1	0	1	1	0	0

Based on the Table 3, the conversion of binary number to decimal number can be performed by adding together all the decimal number values from right to left at the positions that are represented by a "1".

So, it gives: (256) + (64) + (32) + (8) + (4) = 364_{10}.

2.3.3 CONVERSION BINARY NUMBERS TO HEXADECIMAL NUMBERS

The conversion of binary number to hexadecimal can be done directly using the conversion table as shown in Figure 2.2 below:

Binary (base 2)	Octal (base 8)	Decimal (base 10)	Hexadecimal (base 16)
0	0	0	0
1	1	1	1
10	2	2	2
11	3	3	3
100	4	4	4
101	5	5	5
110	6	6	6
111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F

Figure 2.12: Conversion Table

SELF CHECK 2.3

1. Convert the following decimal numbers to binary numbers:
   - 50_{10}
   - 20_{10}
   - 256_{10}
2. Convert the following binary numbers to decimal numbers:
   - 11011_2
   - 11110000_2
   - 11001100_2
3. Convert the following decimal numbers to hexadecimal numbers:
   - 50_{10}
   - 20_{10}
   - 256_{10}

2.4 BINARY NUMBERS AND ARITHMETIC

The binary numbers and arithmetic means the operation of binary addition, binary subtraction, binary multiplication and binary division.

2.4.1 BINARY ADDITION

The binary addition can be performed by the following rules:

- 0 + 0 = 0
- 0 + 1 = 1
- 1 + 0 = 1
- 1 + 1 = 0 (carry 1 to the next significant bit)

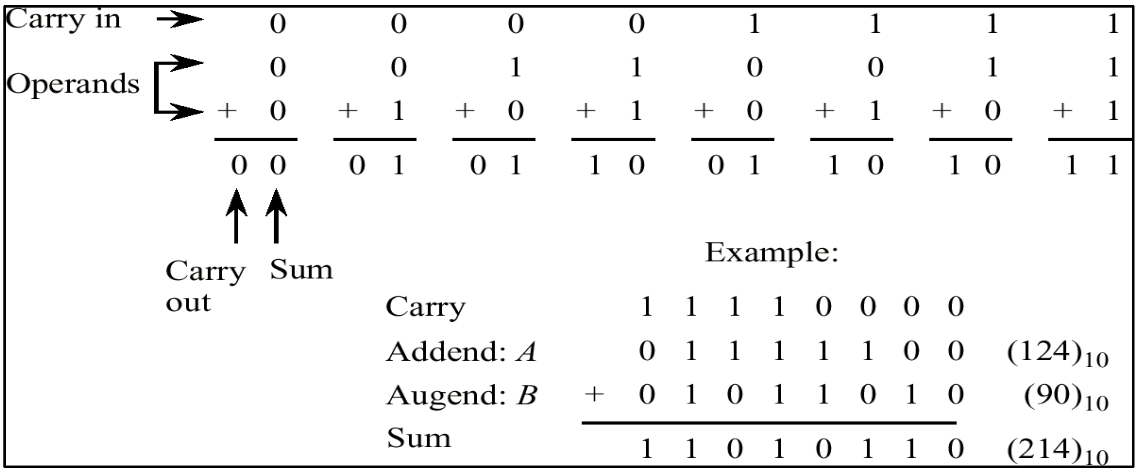

Figure 2.13: Binary Addition

Given the binary number A $(01111100)_2$ and B $(01011010)_{2,}$ then arrange it like a normal addition where A is Addend and B is Augend. Next sum up the digit and carry in the digit like on the upper side of operand and write the carry out digit next to the sum (left). Perform for all digit accordingly.

2.4.2 BINARY SUBTRACTION

The binary subtraction can be performed by the following rules:

- 0 – 0 = 0
- 0 – 1 = 1, borrow 1 from the next more significant bit
- 1 – 0 = 1
- 1 – 1 = 0

Binary subtraction is performed when 1 is subtracted from 0, it is necessary to borrow 1 from the next higher order bit and that bit is reduced by 1 (or 1 is added to the next bit of subtrahend) and the remainder is 1.

For example, subtract the binary number 101_2 from 1001_2:
The solution:

```
1   Borrow
1001
 101
----
 100
```

From the right, refer to the rules and perform the subtraction. then the result is 100_2

2.4.3 BINARY MULTIPLICATION

There are four rules for binary multiplication:

- 0×0=0
- 1×0=0
- 0×1=0
- 1×1=1 (there is no carry or borrow for this)

For example, multiply the 1001_2 by 101_2.
The solution:

```
   1001
x   101
-------
   1001
  0000
 1001
-------
 101101
```

Based on the rules, perform the multiplication operation and the result is 101101_2.

2.4.4 BINARY DIVISION

The binary division can be performed with the following rules:

- 1÷1 = 1
- 1÷0 = Meaningless
- 0÷1 = 0
- 0÷0 = Meaningless

Follow the normal steps; divide, multiply, subtract and bring down the digit. The long division is the best method to get the result.

For example, 01111100 ÷ 0010 where the dividend is 01111100 and the divisor is 0010.

1. Firstly, the most significant bit has to remove in both dividend and divisor which does not change the value of the number.
2. Then the dividend becomes 1111100 and the divisor becomes 10.
3. Next, look at the first two numbers in the dividend and compare with the divisor. Add the number 1 in the quotient place. Then subtract the value and get 1 as remainder.
4. Then bring down the next number from the dividend portion and repeat the step 3.
5. Repeat the process until the remainder becomes zero (0) by comparing the dividend and the divisor value.
6. Lastly, the remainder value equal to 0, the zero is on the left in the dividend portion, then bring that zero to the quotient portion.
7. It will give the result $(111110)_2$

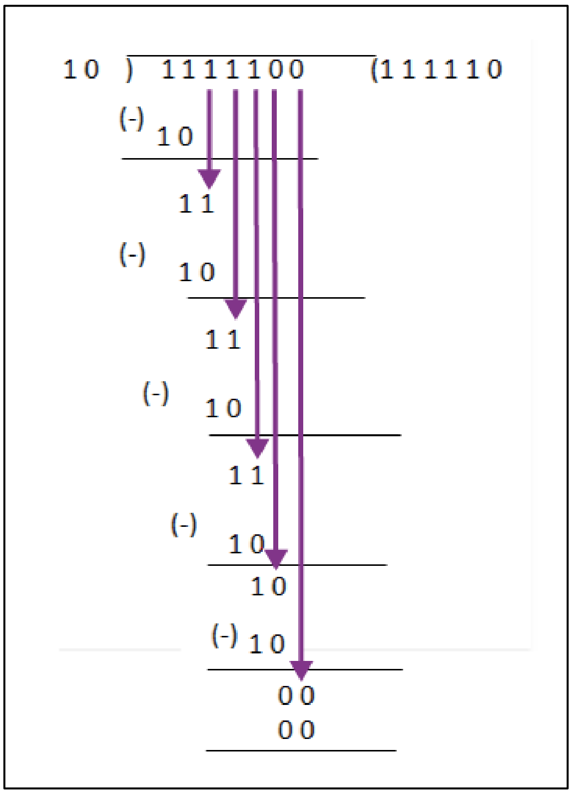

Figure 2.14: Binary Division

SELF CHECK 2.4

1. Add the following numbers:
    - 1011 and 1001
    - 10010 and 1110
2. Subtract the following numbers:
    - 1011 and 1001
    - 10010 and 1110
3. Multiply the following numbers:
    - 1011 and 1001
    - 10010 and 1110
4. Divide the following numbers:
    - 1011 and 1001
    - 10010 and 1110

2.5 HEXADECIMAL NUMBERS AND ARITHMETIC

2.5.1 HEXADECIMAL ADDITION

The hexadecimal addition can be performed by referring to the hexadecimal addition table.

+	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
2	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11
3	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12
4	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13
5	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14
6	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15
7	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16
8	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17
9	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18
A	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19
B	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A
C	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B
D	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C
E	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D
F	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E

X (top row), Y (left column), Sum (table body)

Figure 2.15: Hexadecimal Addition Table

Given the hexadecimal number, $4A6_{16}$ + $1B31_{16.}$

1. Firstly, by refer to the table in Figure 13, add the A_{16} and 5_{16} by locate A in the column X and 5 in the column Y.
2. The point in sum area where the two columns intersect is the sum of two numbers.
3. So, the result is F_{16}.
4. Complete the addition by:

```
   1          carry
   4  A  6  = 1190₁₀
+  1  B  3  =  435₁₀
   6  5  9  = 1625₁₀
```

2.5.2 HEXADECIMAL SUBTRACTION

For subtraction of hexadecimal numbers, it has the same rules as the subtraction of numbers in any other number system.

For example, $4A6_{16}$ - $1B31_{16}$

1. Perform the normal subtraction steps.
2. But, in hexadecimal system a group of 16_{10} will be borrowed.
3. Complete the steps as following:

```
      16        borrow
  ³4  A  6  = 1190₁₀
-  1  B  3  =  435₁₀
   2  F  3  =  755₁₀
```

4. Then the result is $2F3_{16}$.

SELF CHECK 2.5

1. Add the following hexadecimal numbers:
   - $FF45_{16}$ + $86CB_{16}$
   - $2E9F_{16}$ + $947A_{16}$

2. Subtract the following hexadecimal numbers:
   - $FF45_{16}$ + $86CB_{16}$
   - $2E9F_{16}$ + $947A_{16}$

2.6 FRACTION AND MIXED NUMBER CONVERSION

For the fraction number, let say it is (.375).

1. Take the integer .375, then create 3 columns for fraction number, multiplication number and the result.
2. In the first row, write the first fraction number (.375) then multiply by 2 Write the result (0.75) in the other column.
3. Then continue with the last result (0.75) to multiply by 2 then get the result (1.5).
4. Continue with the next step with (1.5) multiply by 2 and get the result (1.0).
5. The multiplication operation will end once the result equal to 1
6. Write the answer in binary number in fraction format from the most significant bit (the first row) to the least significant bit (the last row).
7. So, the answer is $(.011)_2$.

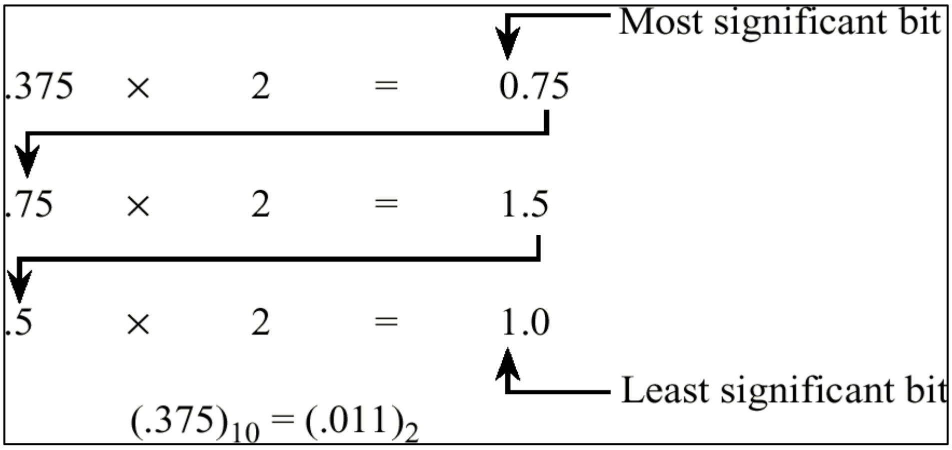

Figure 2.16: Conversion Decimal Number to Binary Number with Fraction Number

In certain case, there is a nonterminating base 2 fraction which has never end such as,

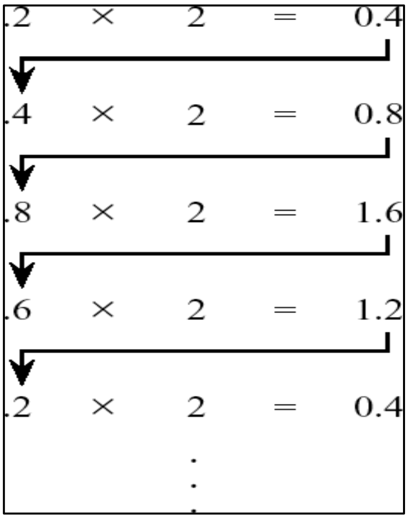

Figure 2.17: Nonterminating Fraction Number

SELF CHECK 2.6

1. Convert the following decimal number to binary number:
    - 0.125_{10}
    - 0.12_{10}

SUMMARY

In this topic you have learnt that:

Data organisation, numbering system, numeric conversion between number bases, binary numbers, hexadecimal numbers, arithmetic, fraction and mixed number conversions.

KEY TERMS

Numbering Systems A systematic method for representing numbers using a particular set of symbols

REFERENCES

Stallings, W., (2019). *Computer Organization and Architecture Designing for Performance.* 11th ed. New York: Pearson.