Module 1 – Number systems

Module 1 overview

Number systems and logic elements (module 2) are the foundations of digital design and microcomputer systems. So, a mastery of this topic is essential to computer engineering.

Number systems (concept map)

Objectives

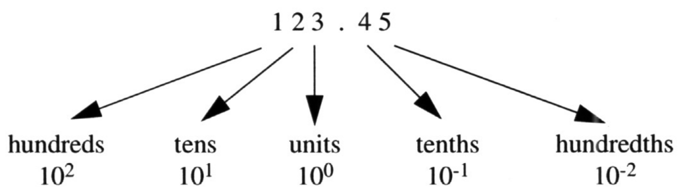
At the completion of this module you will be able to:

- describe the decimal, binary, octal and hexadecimal number systems
- demonstrate an ability to convert from one number system to any other number system
- show how fractional decimal numbers may be converted to binary and vice versa.

1.1 Decimal system

The number system we are most familiar with is the **decimal** system. We write the number 123.45 as two parts separated by a decimal point. The 123 represents the **integer** part and the 45 represents the **fractional** part of the number.

Each digit in the number has a value (or weight) associated with its position in the number. In this case we assign values thus:



The values may be expressed as **powers** of a common **base** or **radix**. In this case the base is for the decimal system and is 10. This number's value is then:

$$1 \times 10^2 + 2\times10^1 + 3\times10^0 + 4\times10^{-1} + 5\times10^{-2}$$

i.e. $100 + 20 + 3 + 0.4 + 0.05 = 123.45$

The decimal system is therefore based on the radix 10 and if we had an electrical device which could assume any one of ten stable states, we could build a computer using the decimal system directly. This is far too difficult to achieve in practice and is not attempted.

1.2 Binary system

In 1623 Sir Francis Bacon experimented with a two-state system and this appears to be the earliest recorded use of the **binary** system which has **two** as its base.

A binary number is only composed of ones (1) or zeros (0). A typical binary number looks like: 1101.

Just as a decimal number has values (or weights) associated with each column, so does a binary number. In this case however the magnitudes are associated with powers of the base two.

For the binary number 1 1 1 1 we have:

2^3 2^2 2^1 2^0
8 4 2 1

weighted values for each column

The number 1101 has a value 8 + 4 + 0 + 1 = 13. Binary numbers can become very large when representing large decimal numbers. For instance the decimal number 999 can be represented with 3 digits however in binary it requires 10 digits.

i.e. 1111100111

1.3 Other systems

A table showing various systems and their bases is shown:

Table 1.1: Systems and bases

Base	System
2	Binary (BIN)
3	Ternary
4	Tetral
5	Quinary
8	Octal (OCT) *
10	Decimal (DEC) *
12	Duodecimal
16	Hexadecimal (HEX) *

32	Duotricenery

* indicates the most common number systems used in the computing field and the ones with which we will be concerned.

If there is any likelihood of confusion about a number, its base is always quoted.

For example:

Decimal 6	=	6_{10}
Octal 6	=	6_8
Binary 6	=	6_2
Hexadecimal 6	=	6_{16}

Unfortunately, the smaller the base number, the larger becomes the number of digits required for its representation. Binary numbers in particular can be lengthy, so the octal or hexadecimal systems are often used to represent them.

The following is a table showing the various number systems

Decimal	**Binary**	**Octal**	**Hexadecimal**
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
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
16	10000	20	10
17	10001	21	11
18	10010	22	12

If we are to use these other systems then it is obvious we need to convert easily from one system to another.

1.4 Conversion rules

1.4.1 Conversion from any system to decimal

Express the number as a series of coefficients times the appropriate power of the base, evaluate and add.

Example

Convert 123_5 to decimal.

Expressing the number in the required format we have:

$$1 \times 5^2 + 2 \times 5^1 + 3 \times 5^0$$

i.e. $25 + 10 + 3 = 38$

$\therefore$ $123_5 = 38_{10}$

1.4.2 Conversion from decimal to any other system

Successively divide the decimal number by the required base, expressing each division as a quotient plus a remainder. Continue this process until the quotient is zero.

The **remainder** taken in **reverse order** becomes the required number.

Example

Convert 123_{10} to binary.

2	123		
2	61	+	1
2	30	+	1
2	15	+	0
2	7	+	1
2	3	+	1
2	1	+	1
2	0	+	1

remainder

Reverse order $= \boxed{1\ 1\ 1\ 1\ 0\ 1\ 1_2} =$ required number

$\therefore \quad \underline{123_{10} \qquad 1111011_2}$

A **check** may be carried out by reconverting to Decimal.

$$
\begin{aligned}
11110112 &= 1 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 + 1 \times 2^4 + 1 \times 2^5 + 1 \times 2^6 \\
&= 1 + 2 + 0 + 8 + 16 + 32 + 64 \\
&= 123
\end{aligned}
$$

Note in this conversion example we have expressed 123 decimal as a **pure binary number**. Often in digital systems another conversion system is used in which **each digit** of the decimal number is encoded into the 8421 weighted system **separately**. This is called **binary coded decimal (BCD).**

An example will illustrate the system.

Example

Convert $1\ 2\ 3_{10}$ to BCD

0001 0010 0011

$\therefore \quad \underline{123_{10} = 100100011 \text{ in BCD.}}$

1.5 Common conversions

1.5.1 Octal system

It is simple to convert from binary to octal and vice-versa.

Example

Convert 110101011100 to octal.

Note: The octal system is to the base 8, so it requires 3 binary digits to represent each digit.

In this case, divide the number in groups of 3 binary digits:

Then convert each group of three: 110 | 101 | 011 | 100

$$1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$$

$$4 + 2 + 0$$

$$= 6 \qquad 5 \qquad 3 \qquad 4$$

Similarly for the other groups: $\therefore \quad \underline{110101011100_2 = 6534_8}$

The reverse procedure is used for octal to binary conversions. Try it for yourself.

1.6 Hexadecimal system

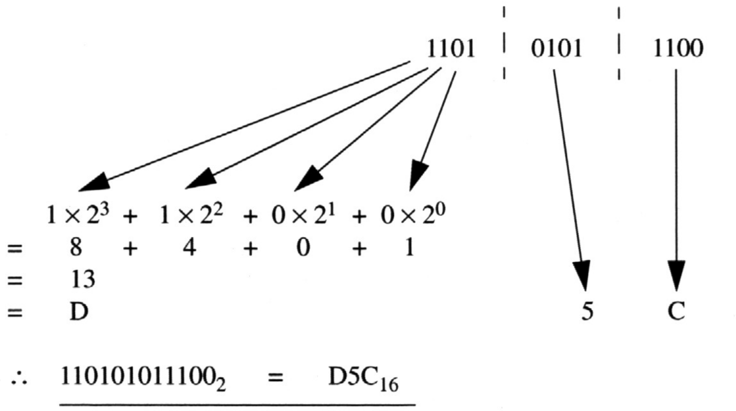
The hexadecimal number system uses the base 16. It differs from other systems in that it uses letters as well as numerals. For example:

Table 1.2: Hexademical system

Decimal	Hexadecimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F

Again, binary – hexadecimal conversions and their reverse are simple.

Divide the binary number in groups of 4 binary digits and proceed as before.



The reverse procedure is used for hexadecimal to binary conversions.

1.7 Fractional numbers

Just as fractional parts are used with the decimal number system, so they must have equivalents in other systems.

1.7.1 Conversion of fractional decimal numbers to any system

To convert the fractional part of a **decimal** number to another base:

1. Multiply successively by that base.
2. Discard and note the digit to the left hand side of the decimal point.
3. The discarded values taken in normal order become the required number.
4. Continue this process until the required accuracy is obtained.

An example best illustrates this process.

Example

Convert 0.523_{10} to octal.

Discarded values

$$
\begin{aligned}
8 \times 0.523 &= 4\;.184 \\
8 \times 0.184 &= 1\;.472 \\
8 \times 0.472 &= 3\;.776 \\
8 \times 0.776 &= 6\;.208 \\
8 \times 0.208 &= 1\;.664
\end{aligned}
$$

$$\text{Normal order} = \boxed{0.41361 \text{ ....(8)}}$$

$$\therefore \quad \underline{0.523_{10} = 0.41361_8}$$

You will notice that from step 4 above, a required accuracy is required. In other words fractional conversions have an error which is dependant on the number of significant figures used by a particular system.

Further, the **integer part** of a number and the **fractional part** of a number are separately converted to form the one number.

1.7.2 Conversion of fractional number to decimal

To convert the fractional part of a number to decimal, use the method described in section 1.4.1. This time, however, use the negative powers of the base:

Example

Convert 0.123_5 to decimal.

Expressing the number in the required format we have:

$$1 \times 5^{-1} \quad + \quad 2 \times 5^{-2} \quad + \quad 3 \times 5^{-3}$$

i.e. $\quad 0.2 \quad + \quad 0.08 \quad + \quad 0.024 \quad = \quad 0.304$

$\therefore \quad 0.123_5 = 0.304_{10}$

1.7.3 Conversion of fractional binary numbers to octal

To convert the fractional part of a binary number to octal, divide the fractional binary number into groups of 3 stating from the decimal point. Convert each group of 3 binary numbers to their octal equivalent.

Example

Convert 0.1011101_2 to octal.

$$0.\underline{101}\,\underline{110}\,\underline{100}_2$$

$$0.\;5\;\;6\;\;4_8$$

Activity 1.1

Attempt the following problems on number systems. The answers are given in brackets. Note, exam questions will be simplified so as not to require a calculator.

1. Convert the following decimal numbers to binary notation.

(i)	81	(1010001)
(ii)	138	(10001010)
(iii)	0.34375	(0.01011)

2. Convert the following decimal numbers to octal notation.

(i)	17	(21)
(ii)	1571	(3043)
(iii)	0.03125	(0.02)

3. Convert the following binary numbers to decimal notation.

(i)	1010110	(86)
(ii)	110110100	(436)
(iii)	0.101001	(0.640625)

4. Convert the following binary numbers to octal notation.

(i)	111	(7)
(ii)	10100110	(246)
(iii)	0.10011	(0.46)

5. Convert the following octal numbers to binary notation.

(i)	172	(1111010)
(ii)	0.622	(0.11001001)

6. Convert the following octal numbers to decimal notation.

(i)	27	(23)
(ii)	0.62	(0.78125)

7. Convert the following decimal numbers to hexadecimal notation.

(i)	25	(19)
(ii)	43	(2B)
(iii)	157	(9D)

8. Convert the following hexadecimal numbers to decimal notation.

(i)	BC	(188)
(ii)	45	(69)
(iii)	2C	(44)

9. Convert the following as indicated.

(i)	11001011_2	=	$?_{16}$	(CB)
(ii)	$E845_{16}$	=	$?_2$	(1110100001000101)