Computer System Architecture

Dr. Raahat Devender Singh
CSED, TIET

This subject is about understanding the hardware operations of a computer system, as well as, the design and architecture of a computer's various components.

Number Systems

In this lecture, we will…

i. study various kinds of number systems (*decimal, binary, octal, hexadecimal*)

ii. learn how numbers are converted from one system to another (*Base Conversion*)

iii. try to understand binary addition and multiplication

iv. learn how negative numbers are represented in binary

Number Systems

Number Systems

Number systems describe how numbers are represented.

System	Base	Symbols	Usage
Decimal	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	Humans
Binary	2	0, 1	Computers
Octal	8	0, 1, 2, 3, 4, 5, 6, 7	Humans
Hexadecimal	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F	Humans

A Quick Example

$$25_{10} = 11001_2 = 31_8 = 19_{16}$$

Decimal Binary Octal Hexadecimal

Base

Base Conversion

Decimal to Binary

Divide the number and every subsequent quotient by two and keep track of the remainder.

First remainder is bit 0 (i.e., the least significant bit)

Second remainder is bit 1, and so on…

$123_{10} = ?_2$

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

$123_{10} = 1111011_2$

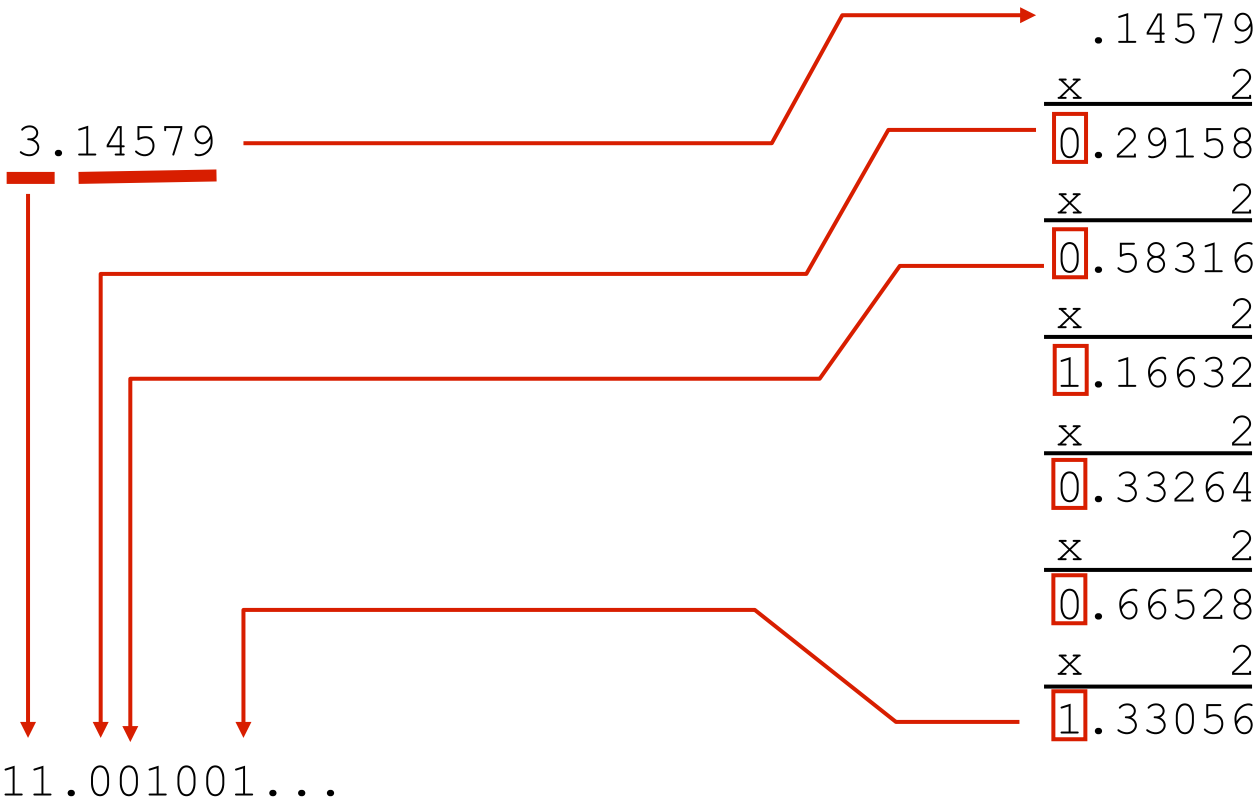
Do it Yourself

$$91_{10} = ?_2$$

Solution

$$91_{10} = 1011011_{2}$$

Decimal to Binary: Fractions



Binary to Decimal

Multiply each bit by 2^n, where n is the "weight" of the bit.

The weight is the position of the bit; the weight of the rightmost bit is 0, the weight of the next bit is 1, and so on.

Add the results.

$101011_2 \Rightarrow$

Bit 5

Bit 0

Weight

$1 \times 2^0 = 1$

$1 \times 2^1 = 2$

$0 \times 2^2 = 0$

$1 \times 2^3 = 8$

$0 \times 2^4 = 0$

$1 \times 2^5 = 32$

43_{10}

Do it Yourself

$$1010011_2 = ?_{10}$$

Solution

$$1010011_2 = 83_{10}$$

Binary to Decimal: Fractions

1 0 . 1 0 1 1

$$\underbrace{1 \qquad 0 \qquad -1 \qquad -2 \qquad -3 \qquad -4}_{\text{Weights}}$$

=>

$$
\begin{aligned}
1 \times 2^{-4} &= 0.0625 \\
1 \times 2^{-3} &= 0.125 \\
0 \times 2^{-2} &= 0.0 \\
1 \times 2^{-1} &= 0.5 \\
0 \times 2^{0} &= 0.0 \\
1 \times 2^{1} &= 2.0 \\
\hline
&\ 2.6875
\end{aligned}
$$

Decimal to Octal

Divide the number and every subsequent quotient by **eight** and keep track of the remainder.

$1234_{10} = ?_8$

		remainders
8	1234	
8	154	2
8	19	2
	2	3
		2

$1234_{10} = 2322_8$

Octal to Decimal

Multiply each digit by 8^n, where n is the "weight" of the digit.

The weight is the position of the digit; the weight of the rightmost digit is 0, the weight of the next digit is 1, and so on.

Add the results.

$723_8 \quad => \quad$

$$
\begin{aligned}
3 \times 8^0 &= 3 \\
2 \times 8^1 &= 16 \\
7 \times 8^2 &= \underline{448} \\
& 467_{10}
\end{aligned}
$$

Decimal to Hexadecimal

Divide the number and every subsequent quotient by **sixteen** and keep track of the remainder.

$1234_{10} = ?_{16}$

16	1234	
16	77	2
	4	13 (D)
		4

$1234_{10} = 4D2_{16}$

Hexadecimal to Decimal

Multiply each digit by 16^n,
where *n* is the "weight"
of the digit.

$$ABC_{16} \Rightarrow \quad C \times 16^0 = 12 \times 11 = 12$$

Add the results.

$$B \times 16^1 = 11 \times 16 = 176$$

$$A \times 16^2 = 10 \times 256 = 2560$$

$$2748_{10}$$

Octal to Binary

Convert each octal digit to its 3-bit binary representation

$705_8 = ?_2$

$$\begin{array}{ccc} 7 & 0 & 5 \\ \downarrow & \downarrow & \downarrow \\ 111 & 000 & 101 \end{array}$$

$705_8 = 111000101_2$

Hexadecimal to Binary

Convert each hexadecimal digit to its 4-bit binary representation

$10AF_{16} = ?_2$

$$
\begin{array}{cccc}
1 & 0 & A & F \\
\downarrow & \downarrow & \downarrow & \downarrow \\
0001 & 0000 & 1010 & 1111
\end{array}
$$

$10AF_{16} = 0001000010101111_2$

Binary to Octal

Group bits into sets of threes, starting from the RHS.

Convert each set to octal digits.

$1011010111_2 = ?_8$

$$\begin{array}{cccc} 1 & 011 & 010 & 111 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 1 & 3 & 2 & 7 \end{array}$$

$1011010111_2 = 1327_8$

Binary to Hexadecimal

Group bits into sets of fours, starting from the RHS.

Convert each set to hexadecimal digits.

$1010111011_2 = ?_{16}$

10	1011	1011
↓	↓	↓
2	B	B

$1010111011_2 = 2BB_{16}$

Octal to Hexadecimal

Use binary as an intermediary.

$1076_8 = ?_{16}$

$$\begin{array}{cccc} 1 & 0 & 7 & 6 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 001 & 000 & 111 & 110 \end{array}$$

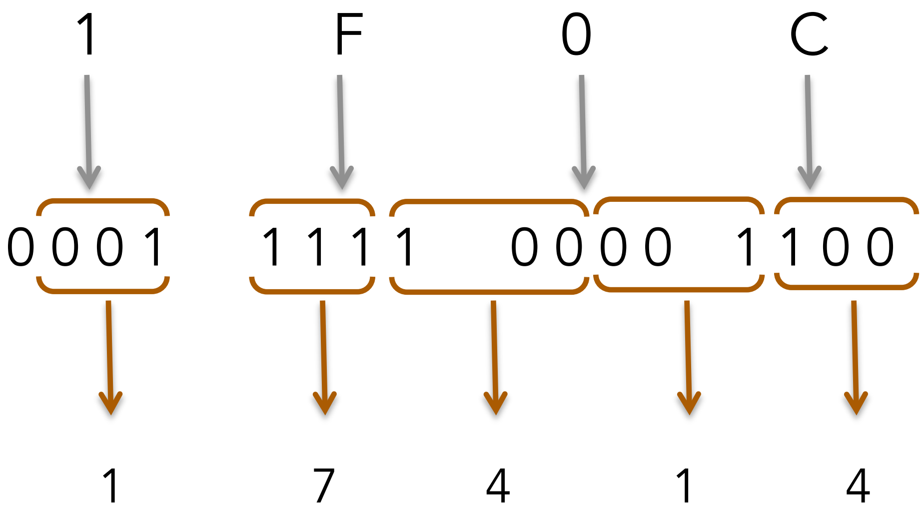
$$\underbrace{0\,0\,1\,0}_{2}\;\underbrace{0\,0\,1\,1}_{3}\;\underbrace{1\,1\,1\,0}_{E}$$

$1076_8 = 23E_{16}$

Hexadecimal to Octal

Use binary as an intermediary.

$1F0C_{16} = ?_8$



$1F0C_{16} = 17414_8$

Binary Addition

Addition of Two 1-Bit Numbers

A	**B**	**A+B**
0	0	0
0	1	1
1	0	1
1	1	10

Addition of Two n-Bit Numbers

Add the individual bits and propagate the carrier.

$$\begin{array}{ccccccc}
 & 1 & & & & 1 & \\
 & & 1 & 0 & 1 & 0 & 1 \\
 & + & 1 & 1 & 0 & 0 & 1 \\
\hline
 & 1 & 0 & 1 & 1 & 1 & 0
\end{array}$$

Binary Multiplication

Multiplication of Two 1-Bit Numbers

A	B	AxB
0	0	0
0	1	0
1	0	0
1	1	1

Multiplication of Two n-Bit Numbers

Binary equivalent of decimal multiplication

$$
\begin{array}{r}
1\,1\,1\,0 \\
\times\ 1\,0\,1\,1 \\
\hline
1\,1\,1\,0 \\
1\,1\,1\,0 \\
0\,0\,0\,0 \\
1\,1\,1\,0 \\
\hline
1\,0\,0\,1\,1\,0\,1\,0
\end{array}
$$

Representing Negative Numbers in Binary

Sign and Magnitude

The Most Significant Bit (MSB) is the sign bit.

0 = Positive

1 = Negative

The remaining bits are the number's magnitude.

For example: 3 is represented as 0011
and -3 is represented as 1011

Problems with 'Sign and Magnitude' Method

i. Two representations for zero (0):

0 = 0000 and also -0 = 1000

ii. Arithmetic operations become complicated:

$$\begin{array}{r} 4 \\ +\ 3 \\ \hline 7 \end{array} \qquad \begin{array}{r} 0100 \\ +\ 0011 \\ \hline 0111 \end{array} \qquad \Bigg| \qquad \begin{array}{r} 4 \\ +\ (-3) \\ \hline 1 \end{array} \qquad \begin{array}{r} 0100 \\ +\ 1011 \\ \hline 1111 \end{array}$$

Solution 1: Ones Complement

Negative number = Bitwise complement of the positive number

3 = 0011

- 3 = 1100

Ones Complement

Ones complement solves the arithmetic problem:

$$\begin{array}{r} 4 \\ +\ 3 \\ \hline 7 \end{array} \qquad \begin{array}{r} 0100 \\ +\ 0011 \\ \hline 0111 \end{array} \qquad \begin{array}{r} 4 \\ +\ (-3) \\ \hline 1 \end{array} \qquad \begin{array}{r} 1\ \ \ 1 \\ 0\ 1\ 0\ 0 \\ +\ 1\ 1\ 0\ 0 \\ \hline 1\ 0\ 0\ 0\ 0 \end{array} \qquad \begin{array}{r} 0\ 0\ 0\ 0 \\ +1 \\ \hline 0\ 0\ 0\ 1 \end{array}$$

Add the Carry

Problem with Ones Complement

Two representations for zero (0):

0 = 0000   and also   -0 = 1111

Solution 2: Twos Complement

Negative number = Bitwise complement of the positive number
+ one (1)

3 = 0011

- 3 = 1100+1 = 1101

Twos Complement

Twos complement solves the arithmetic problem:

$$
\begin{array}{r}
4 \\
+\ 3 \\
\hline
7
\end{array}
\qquad
\begin{array}{r}
0100 \\
+\ 0011 \\
\hline
0111
\end{array}
\qquad\Bigg|\qquad
\begin{array}{r}
\\
4 \\
+\ (-3) \\
\hline
1
\end{array}
\qquad
\begin{array}{r}
1\ \ 1\ \ \ \ \ \ \ \ \ \ \\
0\ 1\ 0\ 0 \\
+\ 1\ 1\ 0\ 1 \\
\hline
1\ 0\ 0\ 0\ 1
\end{array}
$$

Drop the Carry

In the next lecture, we will study…

i. Logic gates

ii. Boolean algebra

iii. K-maps and K-map simplification

iv. Combinational circuits (half adder, full adder)

Homework

What is the difference between *Computer Organization*, *Computer Design*, and *Computer Architecture*?

Reference Books

I. Morris Mano, “Computer System Architecture”, Prentice Hall.

II. J.P. Hayes, “Computer Architecture and Organization”, McGraw Hill.