Computer Fundamentals

Lecture # 4:

Number systems and Logical Operations

Today’s Aim

- Learning about the different Numbering Systems
- Learning Conversion Techniques among the different Number Systems
- Studying the Important Logical Operations

Numbers:

- Number Sense
- Counting

History of Number Systems:

- Quipu of the Inca Umpire
- Fractions in Ancient Egypt
- The Mayan Number System
- The Egyptian Number System
- The Greek Number System
- The Babylonian Number System

Main Numbering Systems:

- Decimal
- Binary
- Hexadecimal

Decimal Number System:

- Base-10 system: 0,1,2,3,4,5,6,7,8,9
- Positional Number System i.e., every place has its own weight
- Examples:
  - $123.64 = 1*10^2 + 2*10^1 + 3*10^0 + 6*10^{-1} + 4*10^{-2}$
  - $0.456 = 4*10^{-1} + 5*10^{-2} + 6*10^{-3}$

Binary Number System:

- Base-2 system : 0,1
- Examples:
  - 1011001
- Used in all Digital Devices
- Why?

Hexadecimal Number System:

- Base-16 Number System :
  0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
- Used for compact representation of binary numbers. How?

Conversion:

Binary to Decimal:

$1011_2 = 1 * 2^0 + 1 * 2^1 + 0 * 2^2 + 1 * 2^3$

$= 1+2+0+8 = 11$

Decimal to Binary:

$5_{10} = 101_2$

2	5 - 1
2	2 - 0
	1

Conversion (continued):

Hexadecimal to decimal:

$DEAD_{16} = ?$

Decimal to hexadecimal:

$207_{10} = ?$

Data Organization:

- Bits (Binary Digits)
- Nibbles = 4 bits
- Bytes = 8 bits
- Word = 16 bits

Logical Operations

- The value of a Boolean expression may either be 'true' or 'false', represented by a '1' or a '0'
- There are three basic Boolean expressions:
  - AND
  - OR
  - NOT
- Other logic operations (derived from these three) include 'NAND', 'NOR', 'XOR' etc.

Logical Operations

- AND
  - ‘if any input is low (0), output is low’ or ‘output is high (1) only if all the inputs are high’
  - Two or more inputs, only one output
- OR
  - ‘if any input is high, output is high’ or ‘output is low only if all the inputs are low’
  - Two or more inputs, only one output
- NOT
  - ‘output is complement of the input’
  - Single input, single output

Logical Operations

- Truth table
  - Contains all the possible input values and their outputs
  - 2^n entries of a truth table show all the possible input combinations (n = number of inputs)
- Boolean expression
  - Consists of Boolean variables
  - Another way to represent the input-output relationship

Logical Operations

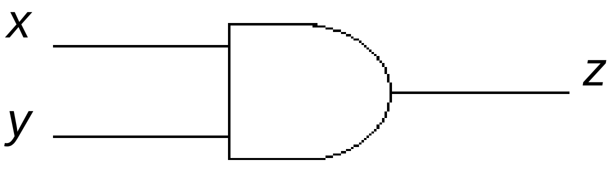
- Logical Diagram (circuit)
  - Representation in the form of a circuit
  - All the inputs and outputs can have only two values, ‘0’ or ‘1’

Logic AND:

Truth Table

x	y	z
0	0	0
0	1	0
1	0	0
1	1	1

Logical Diagram



Boolean Equation

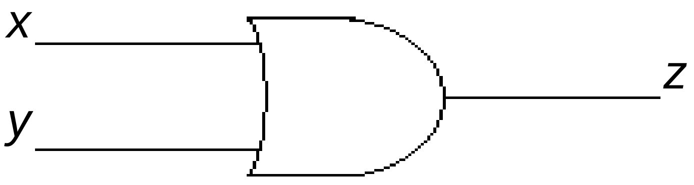
$$z = x \cdot y$$

Logic OR:

Truth Table

x	y	z
0	0	0
0	1	1
1	0	1
1	1	1

Logical Diagram



Boolean Equation

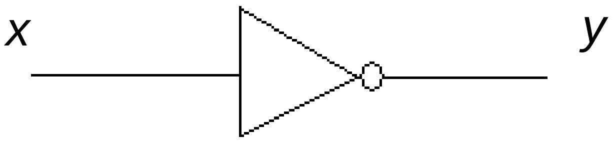
$$z = x + y$$

Logical NOT:

Truth Table

x	y
0	1
1	0

Logical Diagram



Boolean Equation

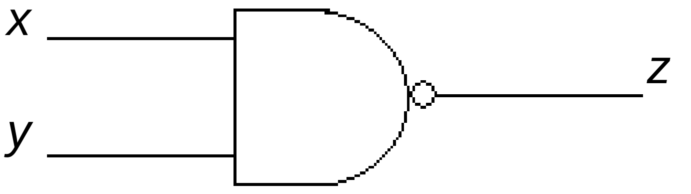
$$y = x'$$

Logic NAND:

Truth Table

x	y	z
0	0	1
0	1	1
1	0	1
1	1	0

Logical Diagram



Boolean Equation

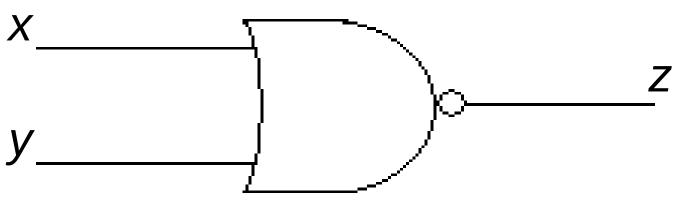
$$z = \overline{x \cdot y}$$

Logic NOR:

Truth Table

x	y	z
0	0	1
0	1	0
1	0	0
1	1	0

Logical Diagram



Boolean Equation

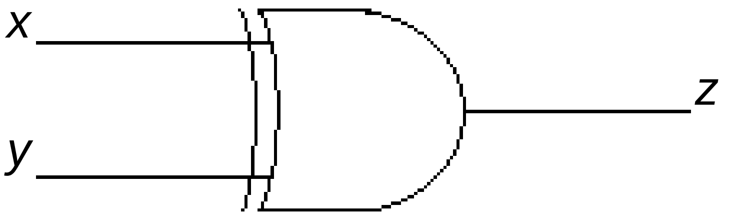
$$z = \overline{x + y}$$

Logic XOR:

Truth Table

x	y	z
0	0	0
0	1	1
1	0	1
1	1	0

Logical Diagram



Boolean Equation

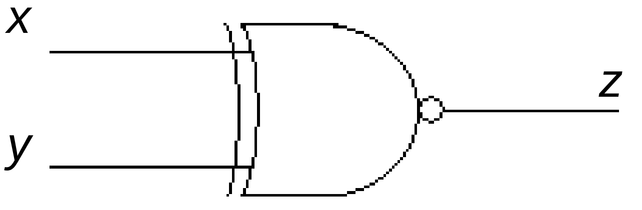
$$z = x \oplus y$$

Logic XNOR:

Truth Table

x	y	z
0	0	1
0	1	0
1	0	0
1	1	1

Logical Diagram



Boolean Equation

$$z = x \odot y$$

Today we Learnt:

- About Number Systems
- Conversion among different number systems
- Binary Logic