ME772A Mechatronics

Introduction to Digital Electronics

31-10-2018

Evolution of Mechatronics

Industrial Revolution

- Allowed design of products and processes for energy Conversion and Transmission.
- Engineering designs were largely mechanical– e.g., operations of motion transmission, sensing, actuation, and computation were performed using mechanical components such as cams, gears, levers, and linkages.
- Purely mechanical systems suffer from
- – Power amplification inability.
- – Energy losses due to tolerances, inertia, and friction.

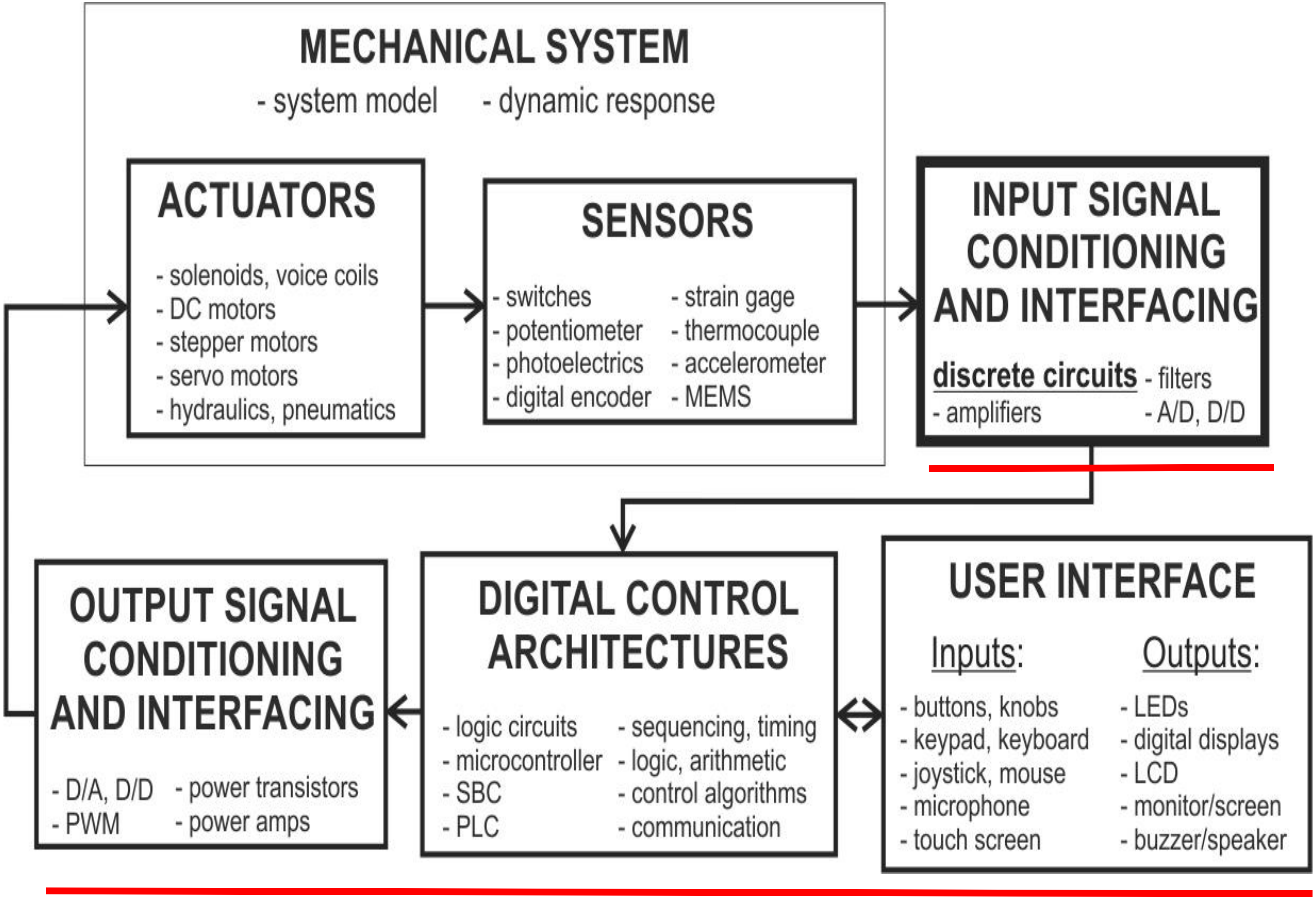
Semiconductor Revolution

- Led to the creation of integrated circuit (IC) technology.
- Effective, miniaturized, power electronics could amplify and deliver needed amount of power to actuators.
- Signal conditioning electronics could filter and encode sensory data in analog/digital format.
- Hard-wired, on-board, discrete analog/digital ICs provided computational and decision-making circuits for control of mechanical devices.

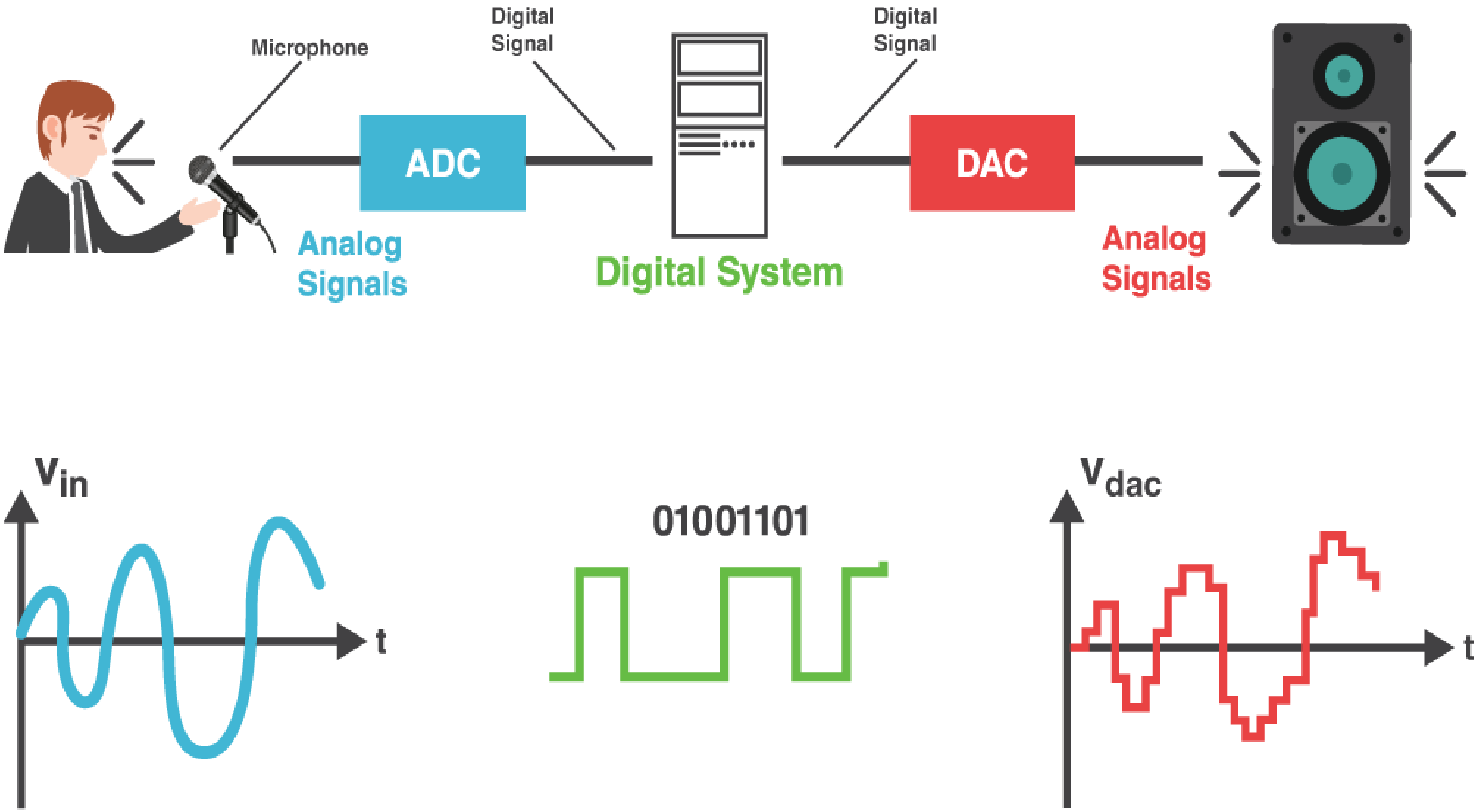
Evolution of Mechatronics continued…

Information Revolution

- Development of VLSI technology led to the introduction of microprocessor, microcomputer, and microcontroller.
- Now computing hardware is ubiquitous, cheap, and small.
- As computing hardware can be effortlessly interfaced with real world electromechanical systems, it is now routinely embedded in engineered products/processes for decision-making.
- Microcontrollers are replacing precision mechanical components, e.g., precision machined camshaft that in many applications functions as a timing device.
- Programmability of microcontrollers is providing a versatile and flexible alternative to the hard-wired analog/digital computational hardware.
- Integrated computer-electrical-mechanical devices are now capable of converting, transmitting, and processing both the *physical energy* and the *virtual energy* (information).
- As a Result: Highly efficient products and processes are now being developed by judicious selection and integration of sensors, actuators, signal conditioning, power electronics, decision and control algorithms, and computer hardware and software.


MECHANICAL SYSTEM
- system model - dynamic response
ACTUATORS
- solenoids, voice coils
- DC motors
- stepper motors
- servo motors
- hydraulics, pneumatics
SENSORS
- switches
- potentiometer
- photoelectrics
- digital encoder
- strain gage
- thermocouple
- accelerometer
- MEMS
INPUT SIGNAL CONDITIONING AND INTERFACING
discrete circuits - filters
- amplifiers - A/D, D/D
OUTPUT SIGNAL CONDITIONING AND INTERFACING
- D/A, D/D
- PWM
- power transistors
- power amps
DIGITAL CONTROL ARCHITECTURES
- logic circuits
- microcontroller
- SBC
- PLC
- sequencing, timing
- logic, arithmetic
- control algorithms
- communication
USER INTERFACE
Inputs:
- buttons, knobs
- keypad, keyboard
- joystick, mouse
- microphone
- touch screen
Outputs:
- LEDs
- digital displays
- LCD
- monitor/screen
- buzzer/speaker

Maintenance of mechanics with electronics through intelligent computer control.



OBJECTIVE

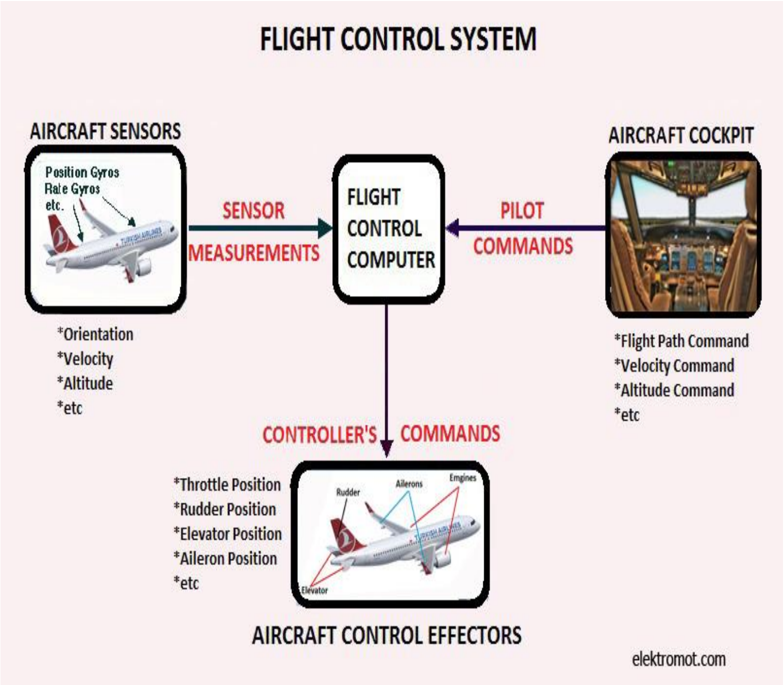
- Understand the relationship among Number system, Boolean logic and digital computer circuits.
- To Learn how to design simple logic circuits.
- Understand how digital circuits work together to form complex computer systems.
- Be able to define a digital signal/ binary signal
- Understand how the binary and hexadecimal number system are used in coding digital data.
- Importance of BCD and Gray code number system
- Know the characteristics of different logic gates
- Know the difference between Combinational and Sequential logic and their applications
- Be able to draw a timing diagram for a digital circuit.

Introduction to Digital Logic Design

- The information system and electronic aspects of mechatronics system employ digital technology.
- Digital logic system will control overall system operation.
- The various digital logic systems are logic circuits, Microcontrollers, PLC’s, sequencing and timing control circuits
- In digital technology, the inputs and outputs are strictly digital in nature.
- Compare to analog signals, it is easy to handle, process, more accurate, reliable and error free.
- The term combinational logic is used for the combination of one or more basic logic gates to form a required logic function
- Understanding the basic functioning of all digital components and systems used in the control of Mechatronics system.
- Digital Logic designs involve complex electronic components that use both electrical and computational characteristics. These characteristics may involve power, current, logical function, protocol and user input.
- Digital Logic Design is used to develop hardware, such as circuit boards and microchip processors. This hardware processes user input, system protocol and other data in computers, navigational systems, cell phones or other high-tech systems.

Why Digital?

- Reproducibility of information
- Flexibility and functionality
- Easier to store, transmit and manipulate information
- Economy: cheaper device and easier to design
- Accuracy and Precision are easier to maintain throughout the system.
- Less affected by noise.
- All the operations can be programmed easily.
- Moore's law – Transistor geometry – Chips double its density (number of transistor) in every 18 months
- Devices become smaller, faster and cheaper
- "Digital circuitry" replaces many analog systems
- Audio recording: from tape to music CD to MP3 (MPEG Layer 3) player
- Image processing: from silver-halide film to digital camera
- Telephone switching networks
- Control of mechanical system: e.g., "flight-by-wire"



Number System

Number System In Mechatronics

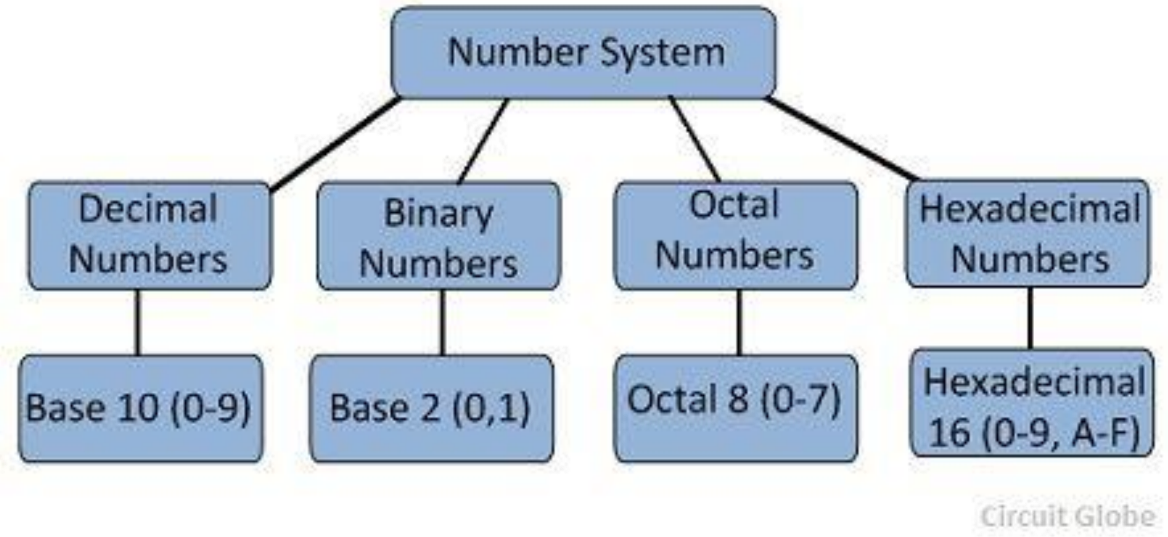
Decimal	Binary	BCD	Octal	Hexadecimal
0	0000	00000000	0	0
1	0001	0000 0001	1	1
2	0010	0000 0010	2	2
3	0011	0000 0011	3	3
4	0100	0000 0100	4	4
5	0101	0000 0101	5	5
6	0110	0000 0110	6	6
7	0111	0000 0111	7	7
8	1000	0000 1000	10	8
9	1001	0000 1001	11	9
10	1010	0001 0000	12	A
11	1011	0001 0001	13	B
12	1100	0001 0010	14	C
13	1101	0001 0011	15	D
14	1110	0001 0100	16	E
15	1111	0001 0101	17	F

Number System mainly used for representing information

Positional Number System

The value of each digit is determined by,

1. the digit itself
2. The position of the digit in the number
3. The base of the number system



Types of Number system

Decimal number system

- One of a Positional number system
- We use this number system in our day-to day life.
- The maximum value of a single digit is 9.
- Has 10 symbols or digits(0,1,2………..9), hence base is 10.
- Each position of a digit represents a specific power of the base (10)

Example

$2586_{10} = (2 \times 10^3) + (5 \times 10^2) + (8 \times 10^1) + (6 \times 10^0)$

$= 2000 + 500 + 80 + 6$

Binary number system

- Positional number System.
- The maximum value of a single digit is 1.
- Has 2 symbols or digits(0,1) hence base is 2.
- Each position of a digit represents a specific power of the base (2)
- This number system widely used in Computers. Bit means a binary digit. In computer terminology bit represents either 0 or 1.

Advantages of Binary system ?

- Easy to represent just two things instead of ten things like decimal
- With just two levels there is more margin against the noise
- With 10 levels noise play havoc with the system

Example

$10101_2 = (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) \times (1 \times 2^0)$

$= 16 + 0 + 4 + 0 + 1$

$= 21_{10}$

Octal number system

Since there are only 8 digits, 3 bits(2^3=8) are sufficient to represent any octal number into binary.

Representing larger numbers in binary number system was a difficult task

Hence the other two number systems namely octal number system and hexadecimal number system were introduced.

Example

$$2057_8 = (2 \times 8^3) + (0 \times 8^2) + (5 \times 8^1) + (7 \times 8^0)$$
$$= 1024 + 0 + 40 + 7$$
$$= 1071_{10}$$

Binary Number: 101010011.110100

Group of three digits: 101 010 011. 110 100

Octal Equivalent: 5 2 3 6 4

$= (523.64)_8$

Hexa Decimal number system

Each position of the digit represents a specific power of the base(16)

Since there are only 16 digits, 4 bits (2^4=16) are sufficient to represent any hexadecimal number into binary

Example

$$1AF_{16} = (1 \times 16^2) + (A \times 16^1) + (F \times 16^0)$$
$$= 1 \times 256 + 10 \times 16 + 15 \times 1$$
$$= 256 + 160 + 15$$
$$= 431_{10}$$

Binary Number: 101010011.00110100

Group of four digits: 0001 0101 0011. 0011 0100

1 5 3 3 4

Hexadecimal Equivalent: $(153.34)_{16}$

BCD code number system

The binary representation of numbers from 0 to 9

The binary-coded decimal (BCD) is an encoding for decimal numbers.
It is sometimes used since it is easy to code and decode, however it is not an efficient way to store numbers.

To express a decimal number in the 8421 code:
-Replace each decimal digit by the 4-bit code

To determine a decimal number from an 8421 code:
- Break the code into groups of 4 bits
- Write the decimal digit represented by each 4-bit group.

To encode the same decimal number, BCD needs more number of bits than binary number

Eg. $(78)_{10} \longrightarrow (0111\ 1000)_{BCD}$

$\longrightarrow$ (1001110)2

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

Gray Code number system

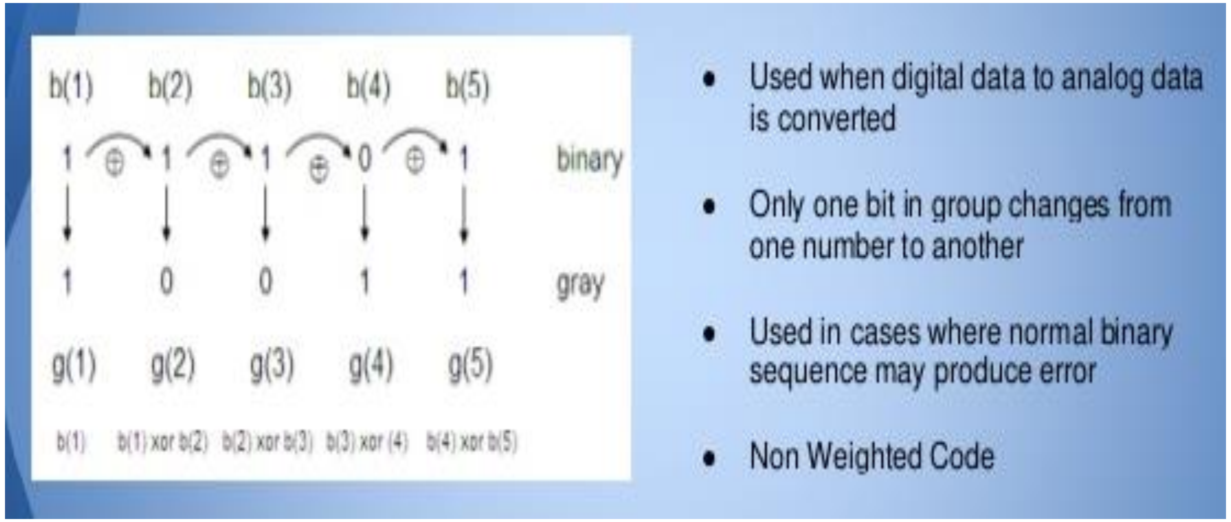
Similar to binary number system

Converted to binary prior to arithmetic operation

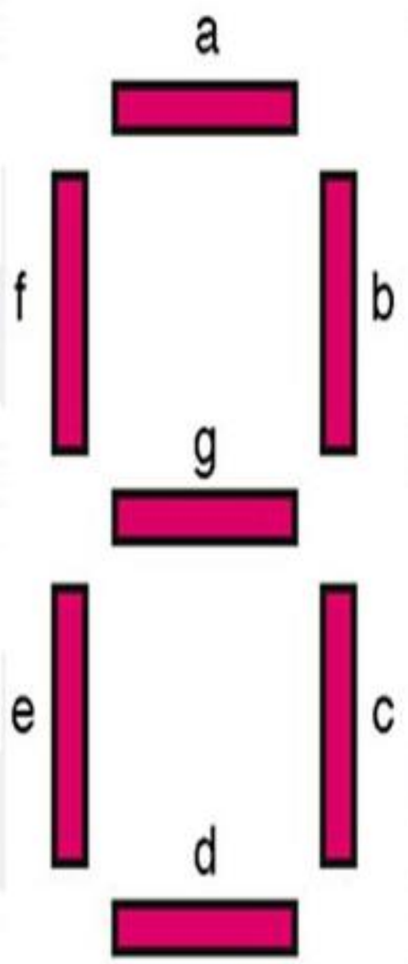
Used for detecting the position of a motor shaft

Only one digit changes state as a number of increments or decrements

Used in designing the pattern of the disk of the Optical encoder.



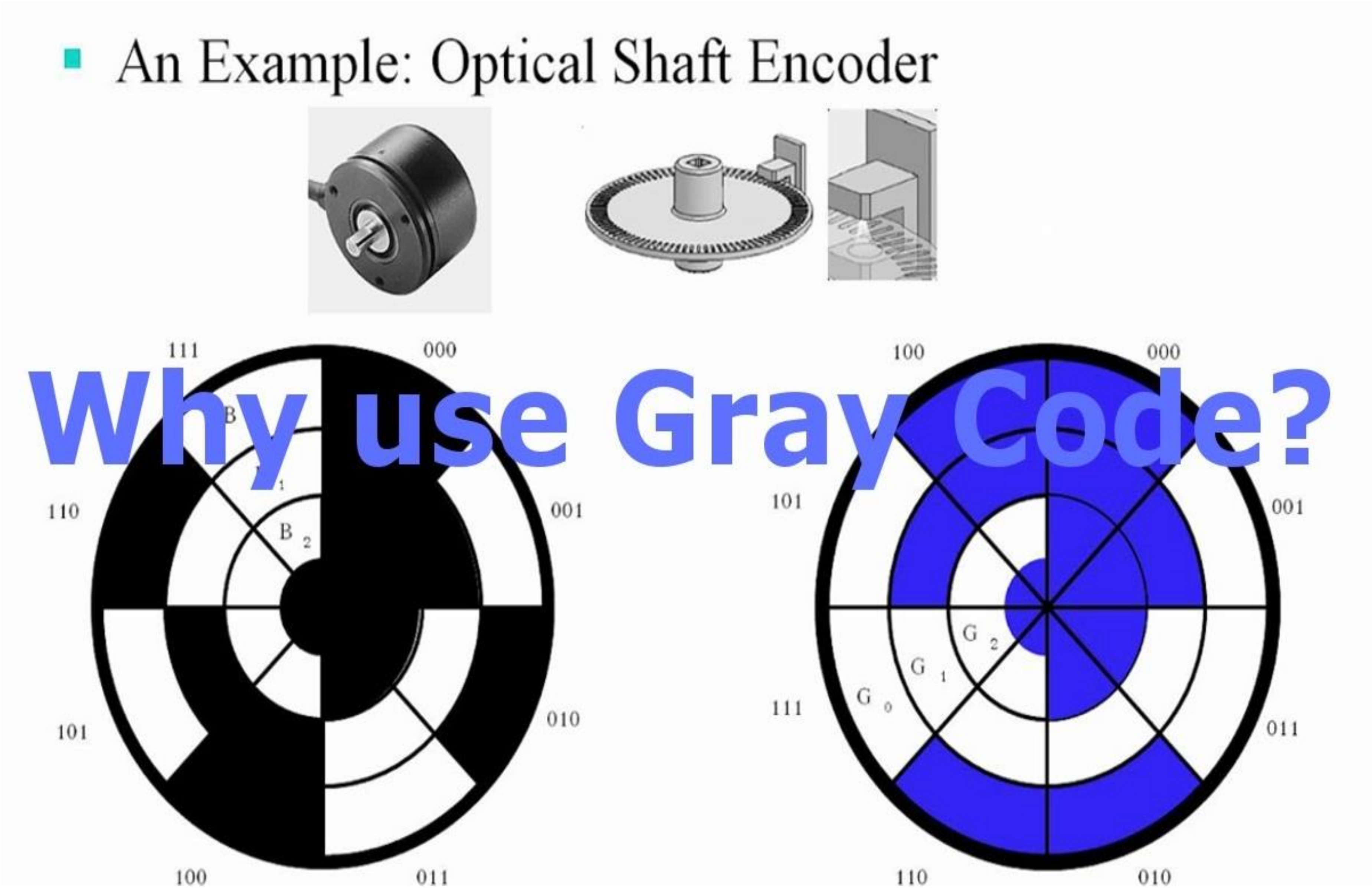
Application of BCD



Most digital equipment has some means for displaying information in a form that can be understood by the user. This information is often numerical data but also be alphanumeric.

One of the simplest and most popular methods for displaying numerical digits uses a 7-segment configuration to form digital characters 0 to 9 and some times the hex characters A to F

Gray code number system -Application



Boolean Algebra

- An algebraic system that describes the logic circuit, in which the variables are limited to two values, usually 0 and 1.
- George Boole developed an algebra for values for the systematic treatment of logic.
- Boolean algebra deals with variables that take on two discrete values, 0 and 1 , and with operations that assume logical meaning.
- Situations involving "yes-no, true –false,on-off" can be represented by Boolean Logical operations.

Applications

It is used to perform the logical operations in digital computer.

Logical operations are performed by logical operators.

The fundamental logical operators are,

1. AND (conjunction)
2. OR (disjunction)
3. NOT (negation)

Basic Rules of Boolean Algebra

1. $A + 0 = A$	7. $A \cdot A = A$
2. $A + 1 = 1$	8. $A \cdot \overline{A} = 0$
3. $A \cdot 0 = 0$	9. $\overline{\overline{A}} = A$
4. $A \cdot 1 = A$	10. $A + AB = A$
5. $A + A = A$	11. $A + \overline{A}B = A + B$
6. $A + \overline{A} = 1$	12. $(A + B)(A + C) = A + BC$

DeMorgan's Theorem

$$\overline{(AB)} = (\overline{A} + \overline{B}) \qquad \overline{(A + B)} = (\overline{A}\ \overline{B})$$

Commutative Axiom:

A.B=B.A

A+B=B+A

Distributive Axiom:

A.(B+C)=(A.B) +(A.C)

A+(B.C)=(A+B).(A+C)

Idempotency Axiom:

A.A=A

A+A=A

Absorption Axiom

A.(A +B)=A

A +(A.B)=A

Complementation Axiom

$A.\overline{A}=0$

$A+\overline{A}= 1$

$\overline{\overline{A}} = A$

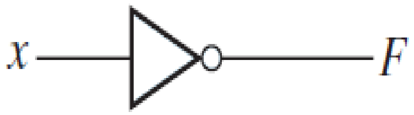
Principle of Duality:

In Boolean Algebra, the duality principle is obtained by interchanging AND and OR operators .
Replacing 0's by 1's and 1's by 0's

Logic gates

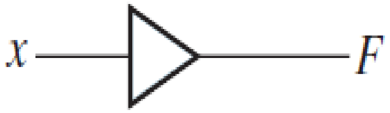
- Logic gates are the basic building blocks for digital electronic circuits.
- A logic gate is an idealized or physical device implementing a Boolean function.
- Primarily implemented using diodes or transistors.
- Logic is the branch of mathematics deals with true('1s') and False ('0s')
- A gate is a device that performs a basic operation on electrical signals.
- The basic gates are AND, OR, NOT gates.
- The universal gates are such as NAND,NOR gates.
- Basically all logic gates have two inputs and a single output, except inverter and buffer.
- Inverter, Buffer have single input and single output.
- Voltage Levels: In most logic gates, the low state is approximately is equal to zero volts, while the high state refers to 5V positive.
- Gates are combined into circuits to perform more complicated tasks.

Inverter



$F = x'$

Buffer

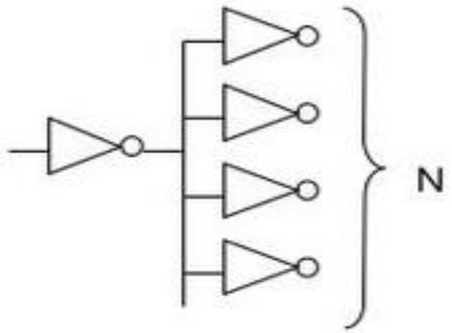


$F = x$

Fan-Out and Fan-In

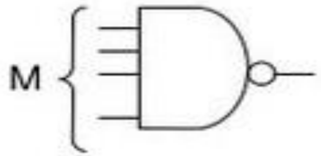
Fan-out – number of load gates connected to the output of the driving gate

- gates with large fan-out are slower

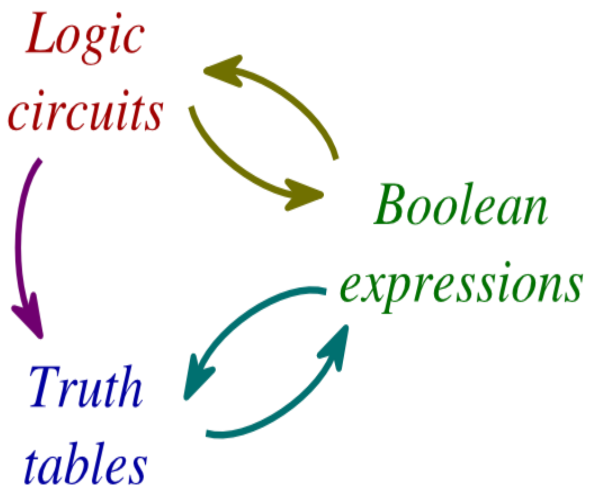


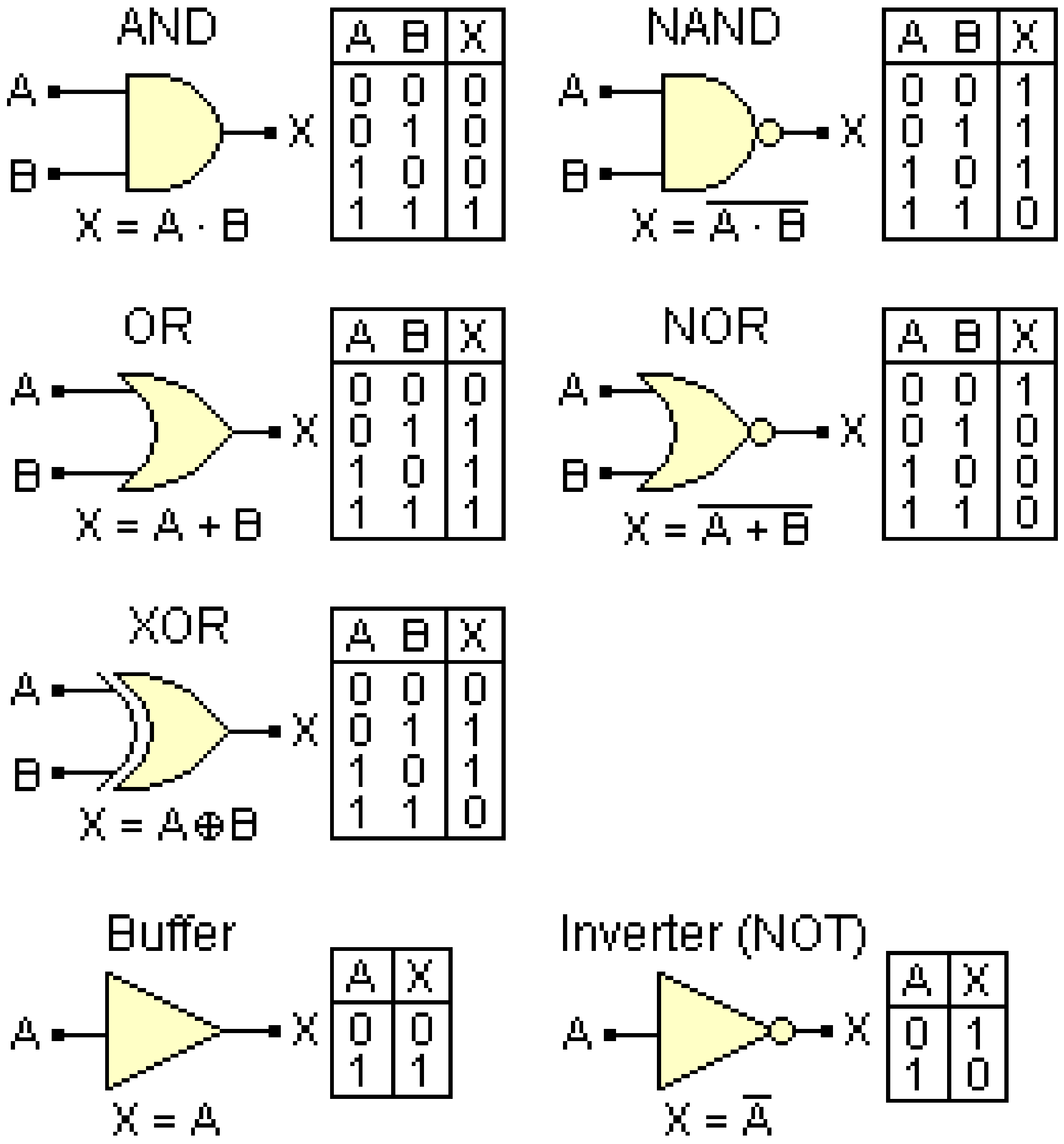
Fan-in – the number of inputs to the gate

- gates with large fan-in are bigger and slower



Logic Gates and its Truth Table





AND

A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

NAND

A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

OR

A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

NOR

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

XOR

A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

Buffer

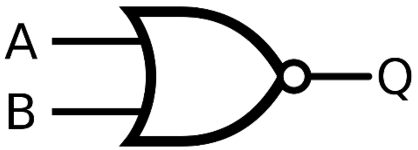
A	X
0	0
1	1

Inverter (NOT)

A	X
0	1
1	0

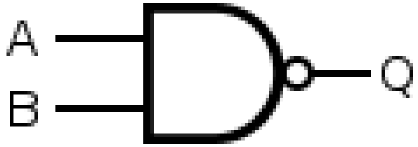
Derived from basic gates
(AND,OR,NOT)

NOR



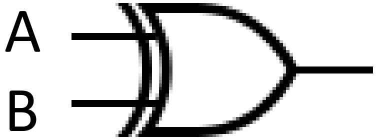
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

NAND



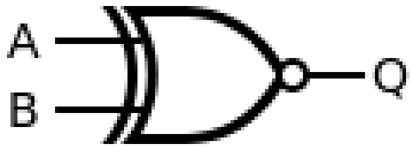
A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

XOR



A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

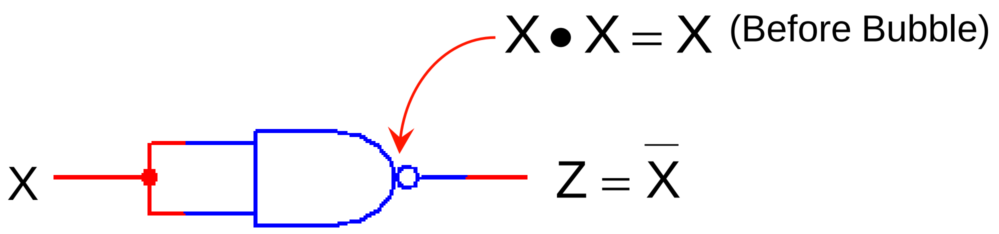
Q
XNOR



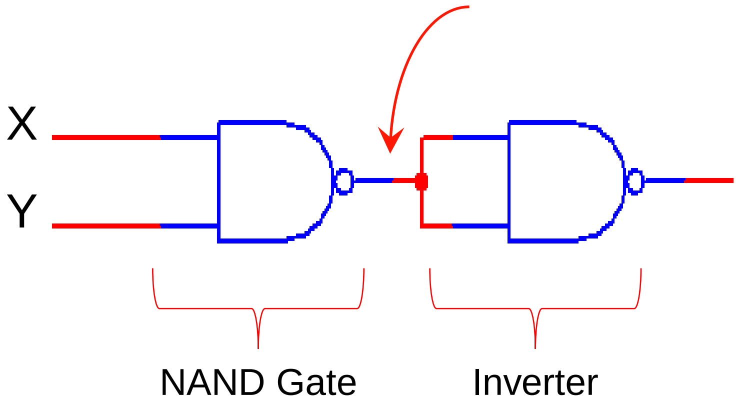
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

Implementation of Basic Logic gates using NAND gate

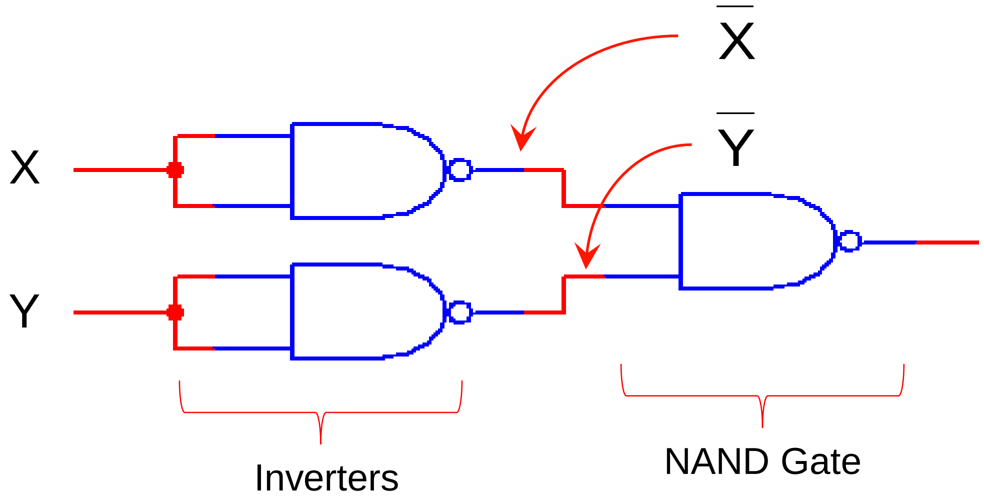
Inverter



AND gate

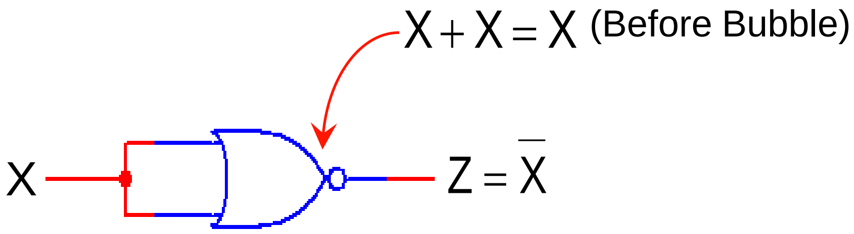


OR gate

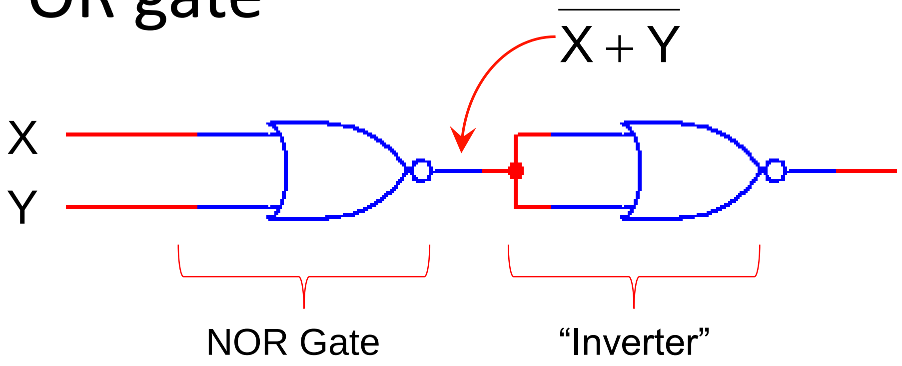


Implementation of Basic Logic gates using NOR gate

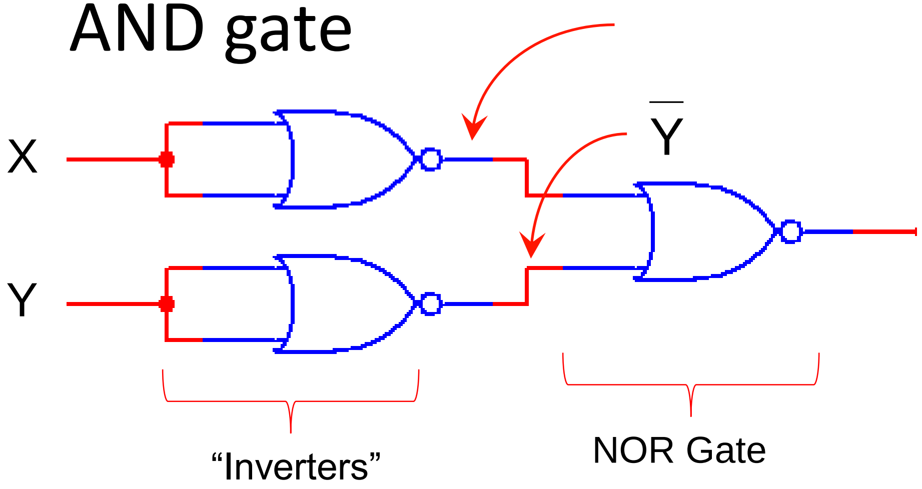
Inverter



OR gate

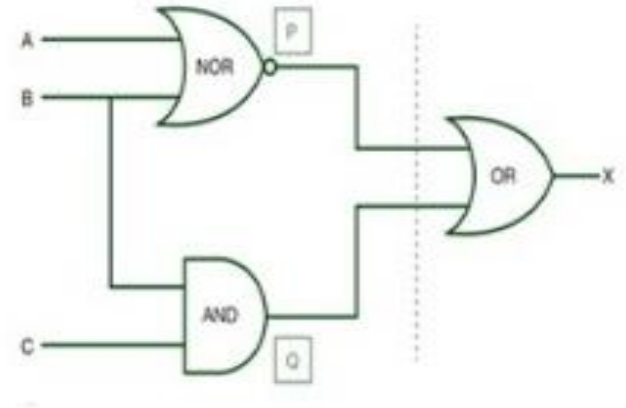


AND gate



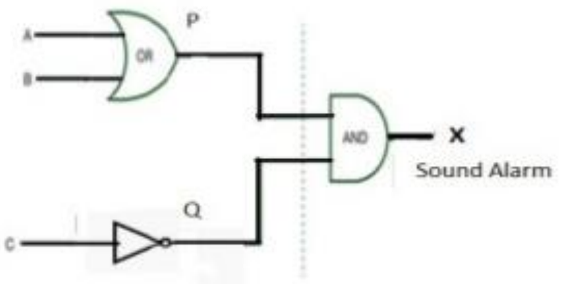
Implementation of Logic Circuit

With P(A,B) and Q© as inputs design a truth table from the following logic circuit network. X is the overall output



X= P OR Q
Where,
P= A NOR B= (A+B)'
Q=B AND C = B.C
So, X= P+Q
        = (A+B)'+B.C

- A system uses 3 switches: A,B and C. A combination of the three switches determines whether an alarm ,X, will make a sound.
- If switch A or B are in the ON position, and if switch C is in the OFF position then a signal to sound an alarm X, is produced.



Where,
A OR B =1(ON position), P=1;
C = 0 (OFF position), Q=1;
P AND Q, X=1---sound an alarm X, is produced.