INTEGRATED CIRCUITS

Lecture-10

By
Bhagyalaxmi Behera
Asst. Professor (Dept. of ECE)

INTEGRATED CIRCUITS

- An integrated circuit (IC) is fabricated on a die of a silicon semiconductor crystal, called a *chip, containing the electronic components for constructing digital gates.*

- The various gates are interconnected inside the chip to form the required circuit. The chip is mounted in a ceramic or plastic container, and connections are welded to external pins to form the integrated circuit.

INTEGRATED CIRCUITS

- The number of pins may range from 14 on a small IC package to several thousand on a larger package.
- Each IC has a numeric designation printed on the surface of the package for identification.
- Vendors provide data books, catalogs, and Internet websites that contain descriptions and information about the ICs that they manufacture.

Levels of Integration

- Digital ICs are often categorized according to the complexity of their circuits, as measured by the number of logic gates in a single package.

- The differentiation between those chips which have a few internal gates and those having hundreds of thousands of gates is made by customary reference to a package as being either a small-, medium-, large-, or very large-scale integration device.

- ***Small-scale integration (SSI)*** *devices contain several independent gates in a single* package. The inputs and outputs of the gates are connected directly to the pins in the package. The number of gates is usually fewer than 10 and is limited by the number of pins available in the IC.

Levels of Integration

- ***Medium-scale integration (MSI)*** *devices have a complexity of approximately 10 to* 1,000 gates in a single package. They usually perform specific elementary digital operations.

- Example of MSI digital functions are decoders, adders, multiplexers , registers and counters.

- ***Large-scale integration (LSI)*** *devices contain thousands of gates in a single package.* They include digital systems such as processors, memory chips, and programmable logic devices.

Levels of Integration

- ***Very large-scale integration (VLSI)*** *devices now contain millions of gates within a* single package.

- Examples are large memory arrays and complex microcomputer chips.

- Because of their small size and low cost, VLSI devices have revolutionized the computer system design technology, giving the designer the capability to create structures that were previously uneconomical to build.

Digital Logic Families

- Digital integrated circuits are classified not only by their complexity or logical operation, but also by the **specific circuit technology** to which they belong.

- The circuit technology is referred to as a ***digital logic family***. *Each logic family has its own basic electronic* circuit upon which more complex digital circuits and components are developed. The basic circuit in each technology is a NAND, NOR, or inverter gate.

Digital Logic Families

- The electronic components employed in the construction of the basic circuit are usually used to name the technology. Many different logic families of digital integrated circuits have been introduced commercially. The following are the most popular:

- TTL transistor–transistor logic;
- ECL emitter-coupled logic;
- MOS metal-oxide semiconductor;
- CMOS complementary metal-oxide semiconductor

Digital Logic Families

- TTL is a logic family that has been in use for 50 years and is considered to be standard.

- ECL has an advantage in systems requiring high-speed operation.

- MOS is suitable for circuits that need high component density, and CMOS is preferable in systems requiring low power consumption, such as digital cameras, personal media players, and other handheld portable devices. Low power consumption is essential for VLSI design; therefore, CMOS has become the dominant logic family, while TTL and ECL continue to decline in use.

DIGITAL LOGIC GATES

- Since Boolean functions are expressed in terms of AND, OR, and NOT operations, it is easier to implement a Boolean function with these type of gates. Still, the possibility of constructing gates for the other logic operations is of practical interest.

- Factors to be weighed in considering the construction of other types of logic gates are

(1) The feasibility and economy of producing the gate with physical components

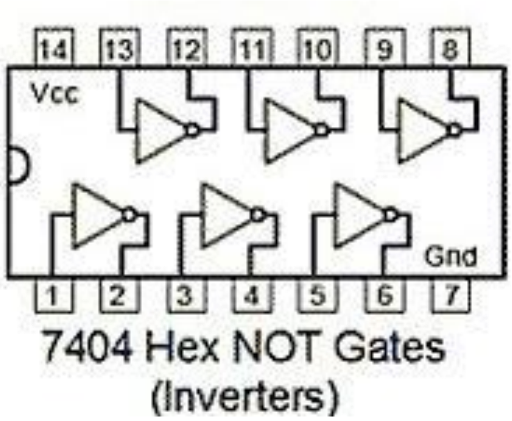
(2) The possibility of extending the gate to more than two inputs

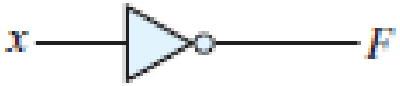
(3) The basic properties of the binary operator, such as commutativity and associativity

(4) The ability of the gate to implement Boolean functions alone or in conjunction with other gates.

Digital Logic gates
NOT Gate

x	F = x’
0	1
1	0



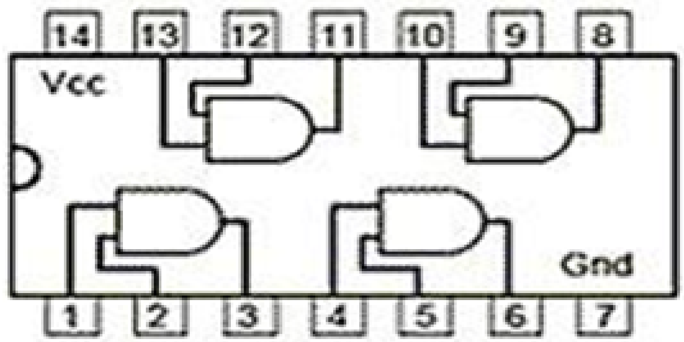


$F = x'$

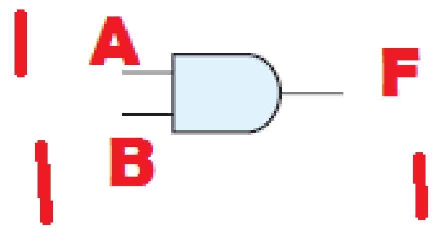
Digital Logic gates

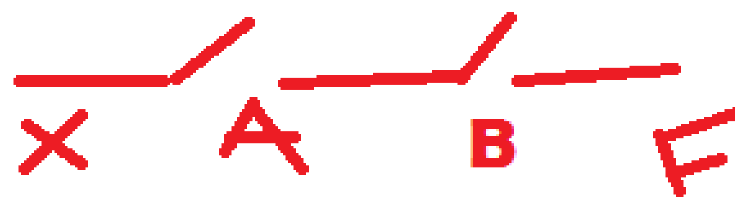
AND Gate

A	B	F = A.B
0	0	0
0	1	0
1	0	0
1	1	1

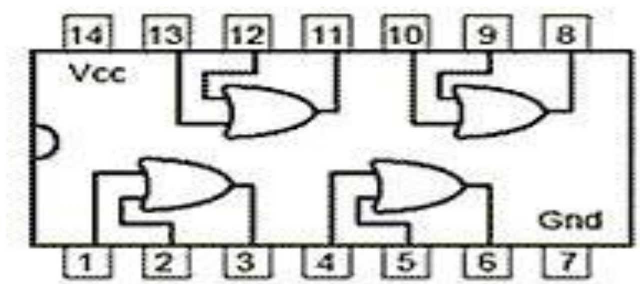


7408 Quad 2 input
AND Gates





A	B	F = A+B
0	0	0
0	1	1
1	0	1
1	1	1

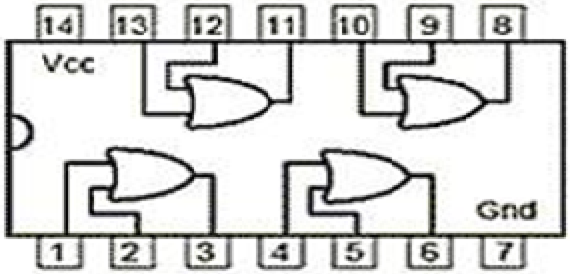


7432 Quad 2 input
OR Gates

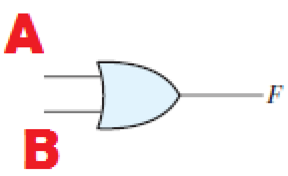
Digital Logic gates

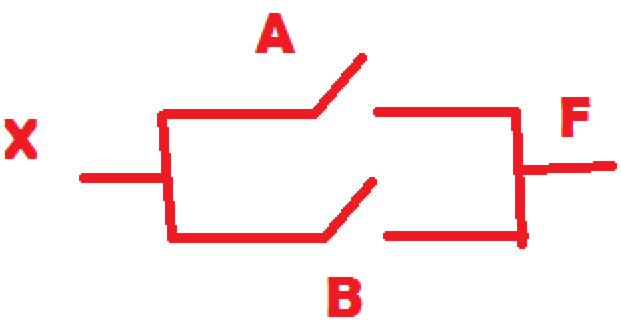
2 Input OR Gate

A	B	F = A+B
0	0	0
0	1	1
1	0	1
1	1	1



7432 Quad 2 input OR Gates

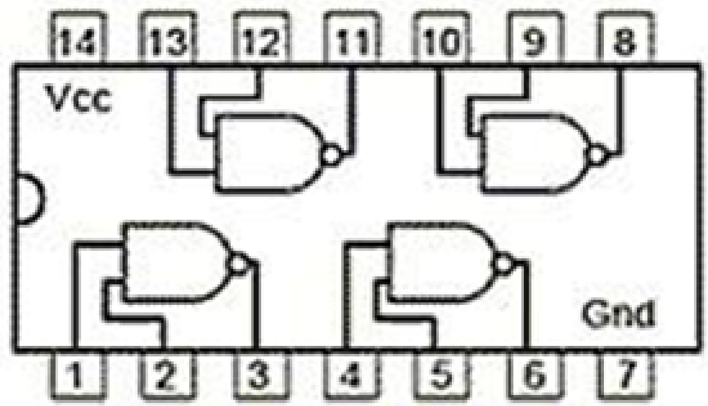




Digital Logic gates

2 Input NAND Gate

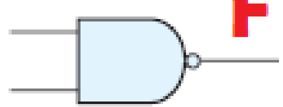
A	B	F = (A.B)’
0	0	1
0	1	1
1	0	1
1	1	0



7400 Quad 2 input NAND Gates

A

F



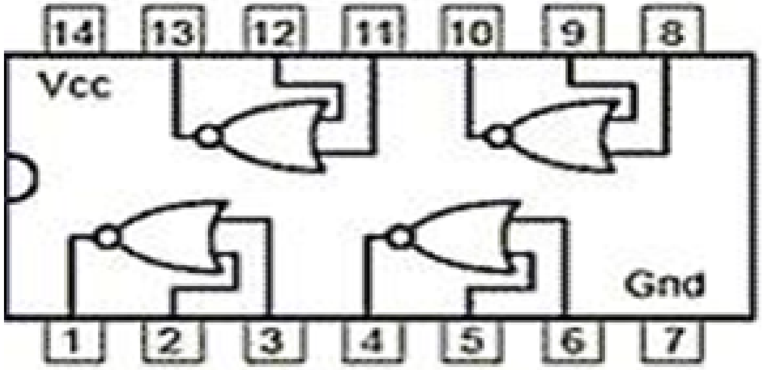
B

=0

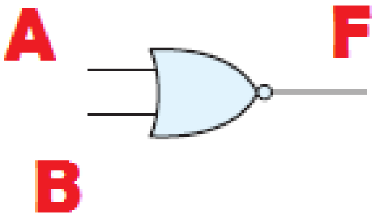
Digital Logic gates

2 Input NOR Gate

A	B	F = (A+B)'
0	0	1
0	1	0
1	0	0
1	1	0



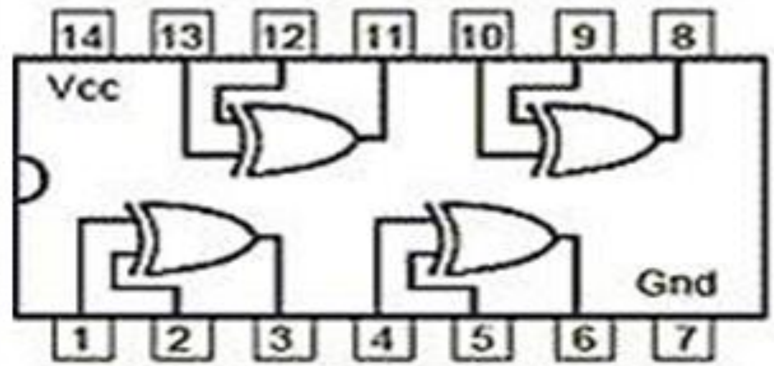
7402 Quad 2 input
NOR Gates



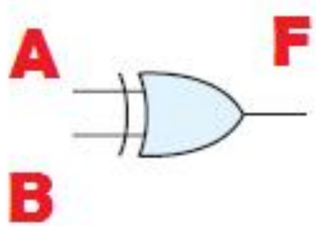
Digital Logic gates

2 Input XOR Gate

A	B	$F = A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0



7486 Quad 2 input XOR Gates



$A'B + AB'$

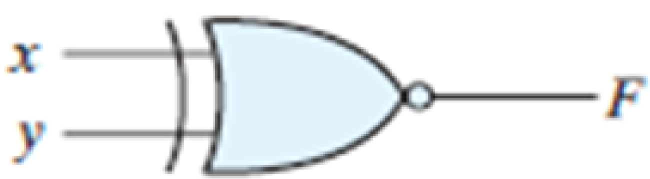
Digital Logic gates

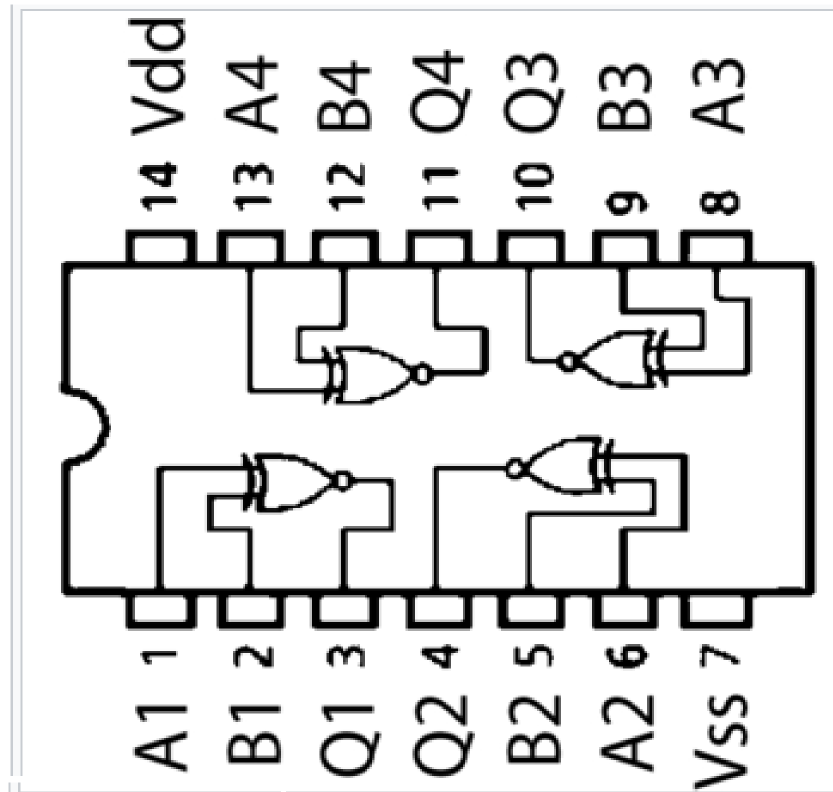
2 Input XNOR Gate

SN74LS266

x	y	F
0	0	1
0	1	0
1	0	0
1	1	1

$$F = xy + x'y' = (x \oplus y)'$$

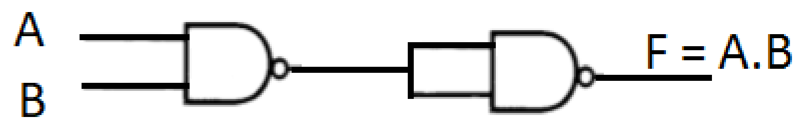




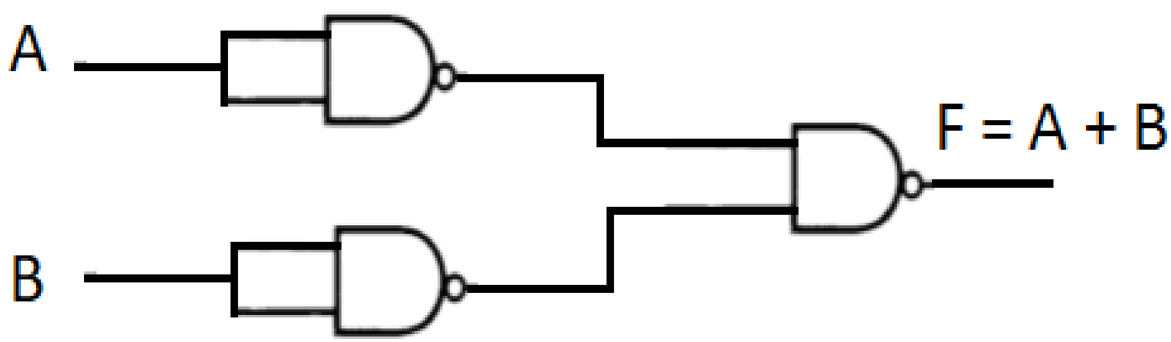
Quad 2 input XNOR gates

Using a single 7400 IC, connect a circuit that produces an inverter

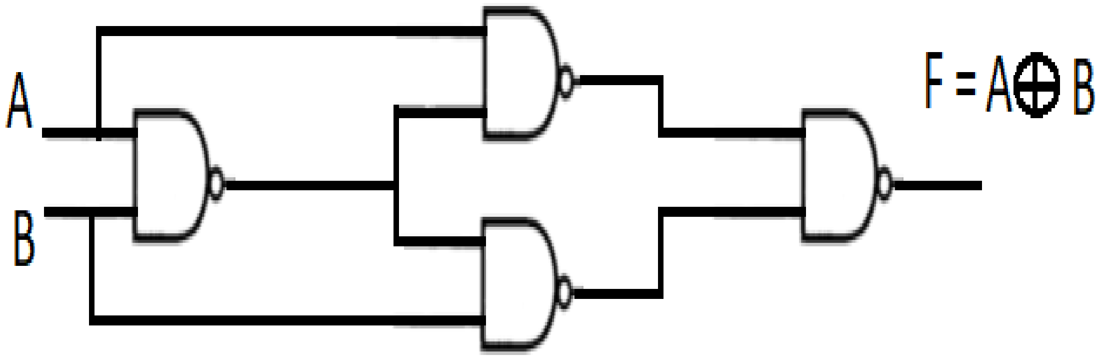
Using a single 7400 IC, connect a circuit that produces a two-input AND gate



Using a single 7400 IC, connect a circuit that produces a two-input OR gate



Using a single 7400 IC, connect a circuit that produces a two-input EX-OR gate

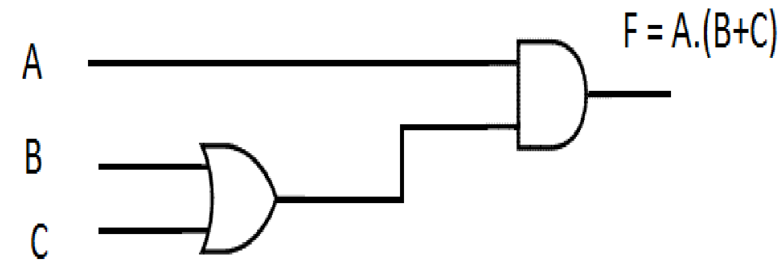


Implement the Boolean function:

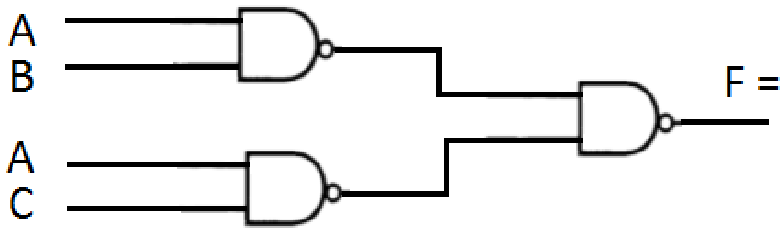
F = A . (B + C)

<u>Circuit Diagram</u>

A	B	C	F=A.(B+C)
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Circuit Diagram using NAND gates



$F = ((AB)'.(AC)')' = ((AB+AC)')' = AB+AC = A.(B+C)$

A	B	C	F=((AB)'.(AC)')'
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1