

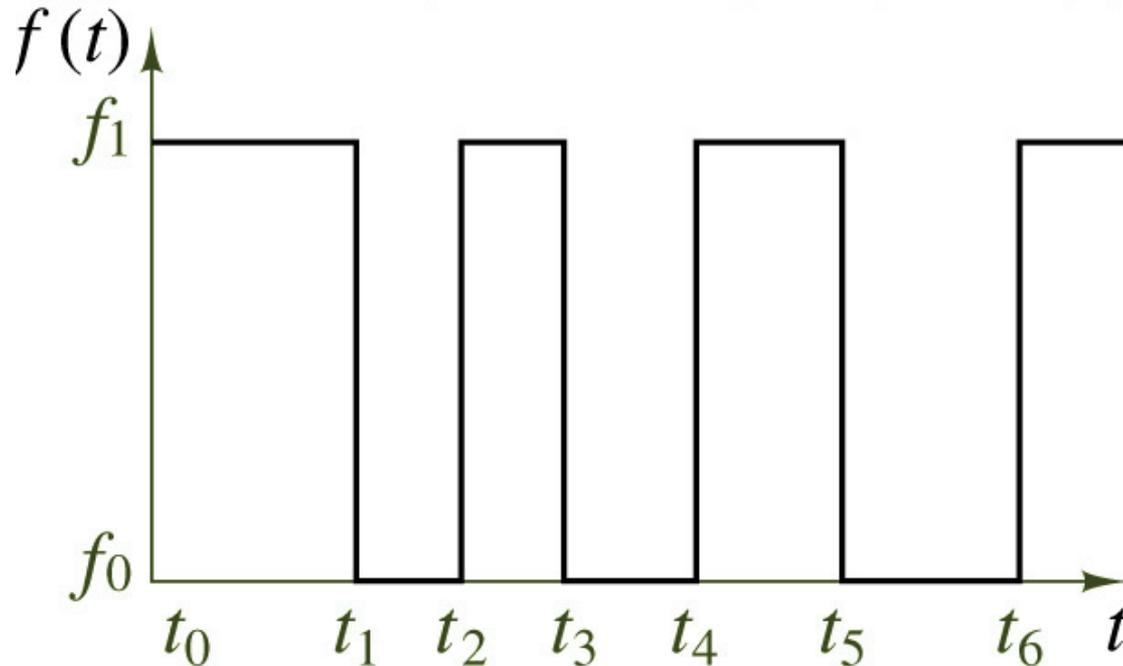
Digital Logic Circuits

- Digital describes any system based on discontinuous data or events. Typically digital is computer data or electronic sampling of an analog signal. Computers are digital machines because at their most basic level they may distinguish between just two values, 0 and 1, or off and on. All data that a computer processes must be encoded digitally, as a series of zeroes and ones.
- The opposite of digital is analog. A typical analog device is a clock in which the hands move continuously around the face. Such a clock is capable of indicating every possible time of day. In contrast, a digital clock is capable of representing only a finite number of times (every tenth of a second, for example).

Binary Signals

Logic gate circuits are designed to input and output only two types of signals: “high” (1) and “low” (0), as represented by a variable voltage: full power supply voltage for a “high” state and zero voltage for a “low” state

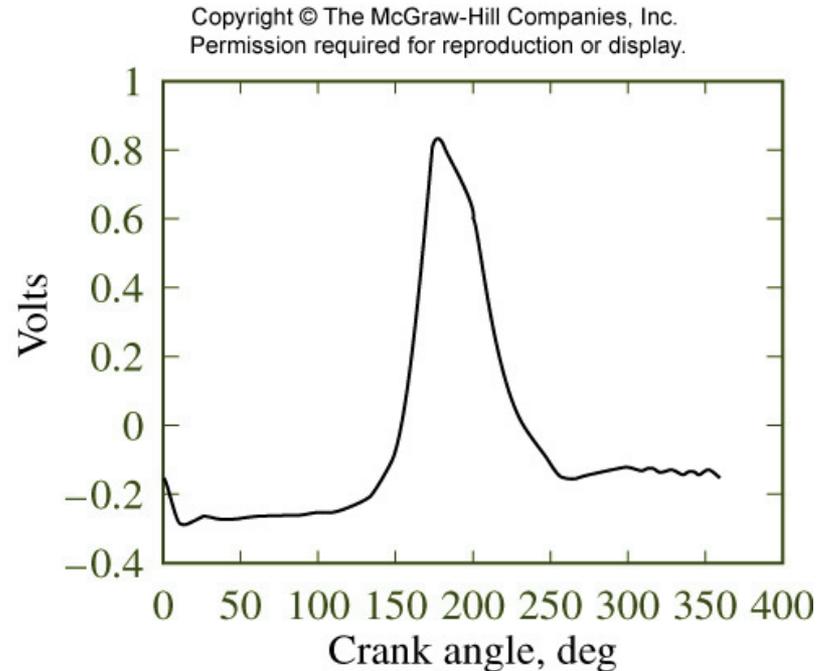
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display



Digital Logic Circuits

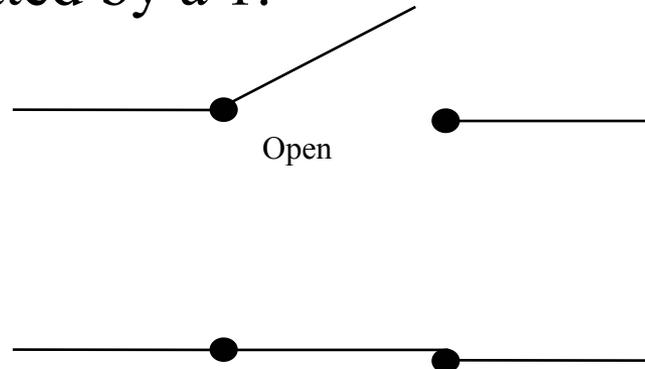
Logic is defined as the science of reasoning. It is the development of a reasonable or logical conclusion based on known information.

Voltage Analog of Internal Combustion Engine in-Cylinder Pressure



Logic Gates

- A gate is an electronic circuit that performs a logical operation. Logic gates are the basic building blocks for digital electronic circuits. A switching circuit or circuit is a composition of gates. Operations on the binaries 0, 1 may be viewed as truth functional operations, and binary arithmetic as an application of propositional logic. Binary arithmetic is implemented by building switching circuits.
- A switch may represent a two-state logic condition. An open switch will be indicated by a 0, while a closed switch will be indicated by a 1.



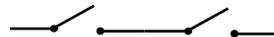
Logic Operators

There are three basic logic operations which are used in the design of all logic circuits. These logic operations are:

- The OR operator which is indicated by a plus sign (+) as: $A + B$
- The AND operator which is indicated by a dot or multiplication sign (\times): $A . B$ or $A \times B$.
- The NOT operator which is indicated by an overbar.

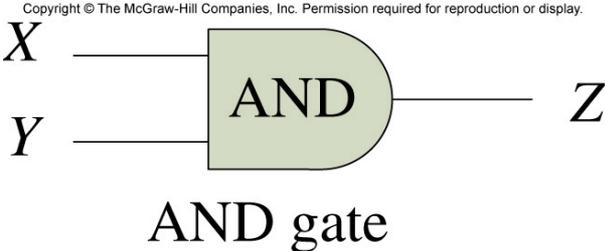
The AND Operator

- The AND operator implements the AND function. It depends upon two or more events happening at the same time. The digital device used for this operation is called the AND gate. With. All inputs must have logic 1 signals in order for the output to be logic 1. With either input at logic 0, the output will be held to logic 0. We may visualize the AND gate as an electrical circuit involving two switches in series as shown.
- An example of an AND gate is an interlock control system for a machine tool such that if the safety guard is in place and gives a logic 1 and the power is on, giving a logic 1, then there can be an output of 1. Accordingly, the machine will operate.



The AND gate is a logic circuit that requires all inputs to be “true” at the same time in order for the output to be “true”.

Logical multiplication and the AND gate



<i>X</i>	<i>Y</i>	<i>Z</i>
0	0	0
0	1	0
1	0	0
1	1	1

Truth table

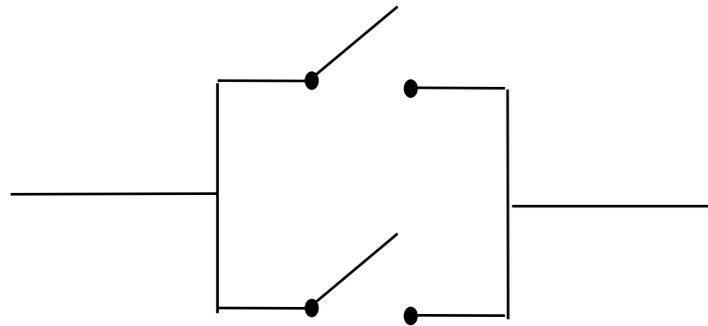
Rules for logical multiplication (AND)

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

$$0 \cdot 0 = 0$$
$$0 \cdot 1 = 0$$
$$1 \cdot 0 = 0$$
$$1 \cdot 1 = 1$$

The OR Gate

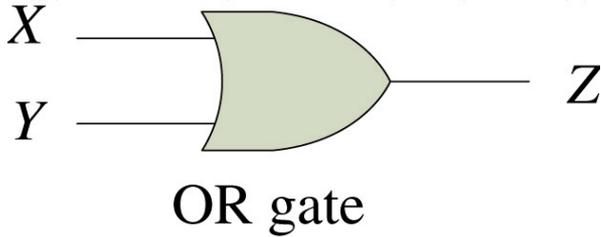
- The OR operator is sort of the reverse of the AND operator. In symbols, the OR function is designated with a plus sign (+). The OR function, like its verbal counterpart, allows the output to be true (logic 1) if any one or more of its inputs are true. The digital device used for this operation is called the OR gate.
- Like the AND gate, the OR gate has 2^n possible combinations required to describe its operation (n is the number of inputs). We may visualize the OR gate as an electrical circuit involving two switches in series as shown



An easy way to remember the OR gate is that any “high” input will yield a “high” output.

Logical addition and the OR gate

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.



X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

Truth table

Rules for logical addition (OR)

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 1$$

The NOT Operator or Inverter

- The function of the NOT operator (gate) may be understood by studying the behavior of the switch. The switch is a two-state device, however, it can be in only one state at a time. For example, the logic number 1 represents the switch in the closed position, while the logic number represents the open position of the switch.
- The NOT gate which is called also an **inverter** is the simplest logic gate. It is a little different from AND and OR gates in that it always has exactly one input as well as one output. Whatever logical state is applied to the input, the opposite state will appear at the output.

The NAND Gate

- The NAND gate is the complementary form of the AND gate. It is very commonly used in practice. It is verified that the logic function implemented by NAND gate corresponds to AND gate followed by an inverter.
- Another way of considering the NAND gate is as an AND gate with a NOT gate applied to invert both the inputs before they reach the AND gate.
- When input A is 1 and input B is 1 there is an output of 0, all other inputs giving an output of 1.

The NOR Gate

- The NOR gate is the complementary form of the OR gate. It is verified that the logic function implemented by NOR gate corresponds to OR gate followed by an inverter.
- Another way of considering the NOR gate is as an OR gate with a NOT gate applied to invert both the inputs before they reach the OR gate. When input A or input B is 1 there is an output of 0.

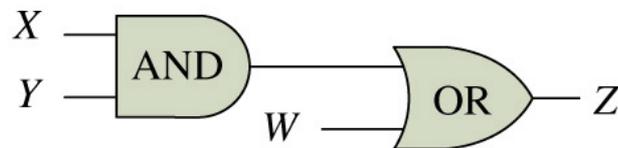
Example of logic function implementation with logic gates

Copyright © The McGraw-Hill Companies, Inc.
Permission required for reproduction or display.

Logic gate realization of the statement “the output Z shall be logic 1 only when the condition $(X = 1 \text{ AND } Y = 1) \text{ OR } (W = 1)$ occurs, and shall be logic 0 otherwise.”

X	Y	W	Z
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

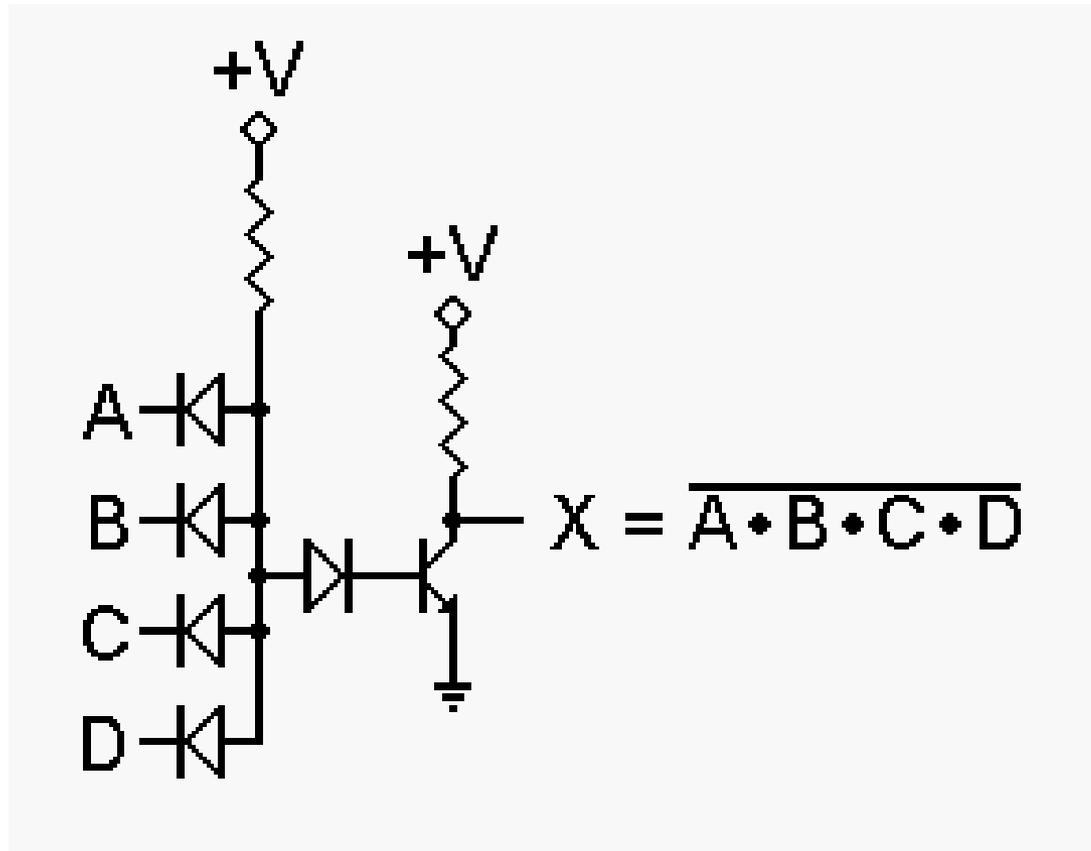
Truth table



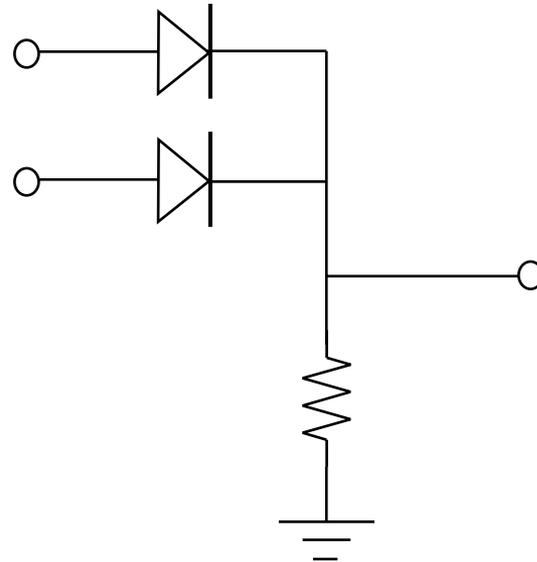
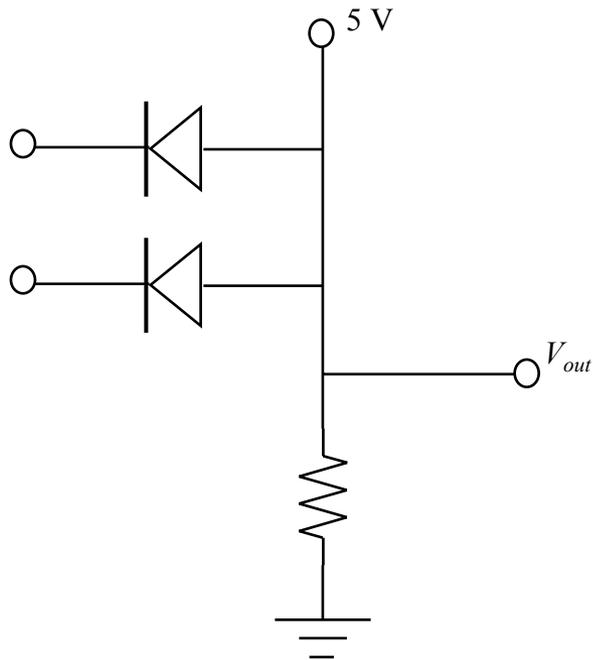
Solution using logic gates

Inside a Logic Gate

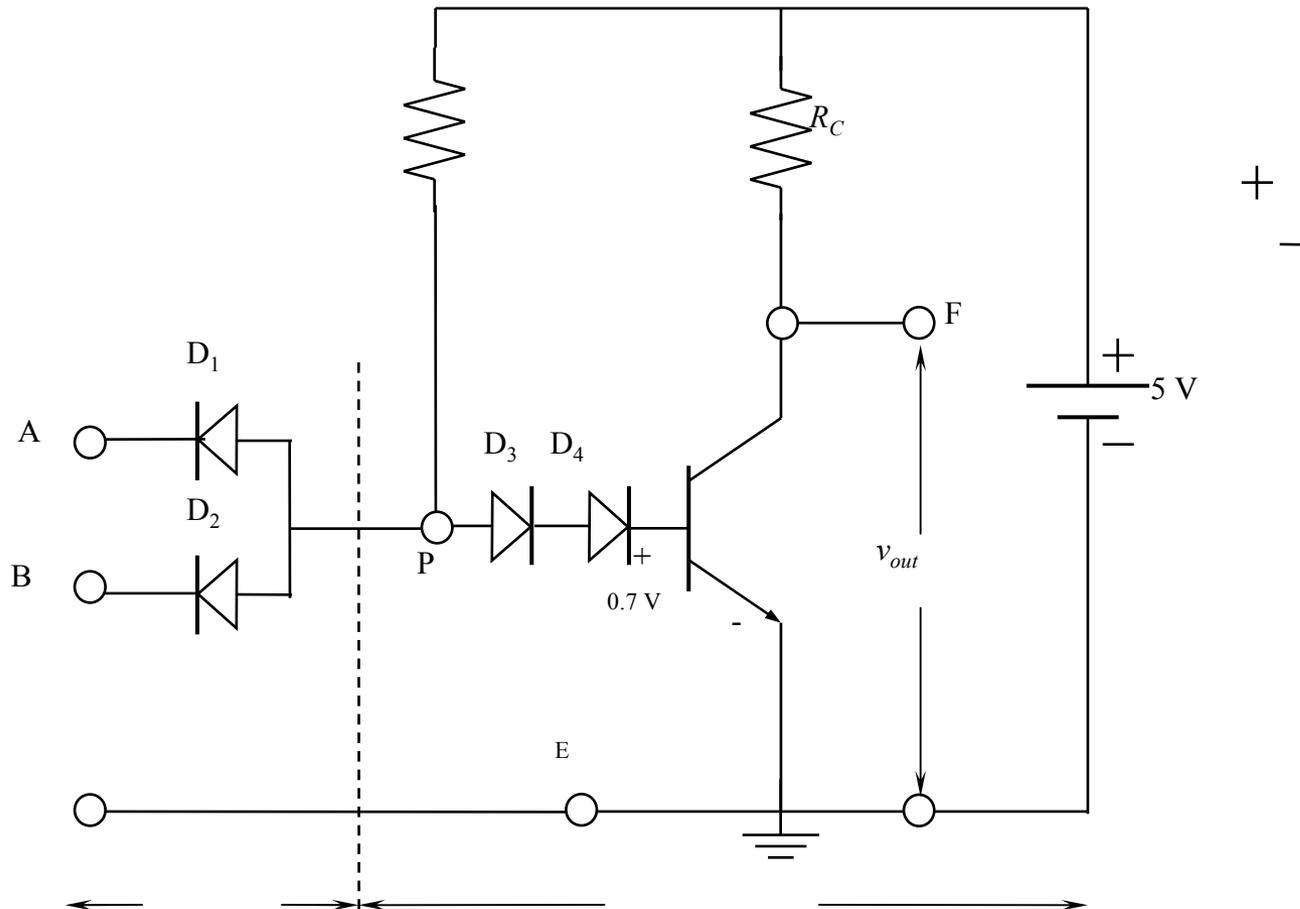
http://www.play-hookey.com/digital/electronics/dtl_gates.html



Diode Logic Gates: AND and OR Gates



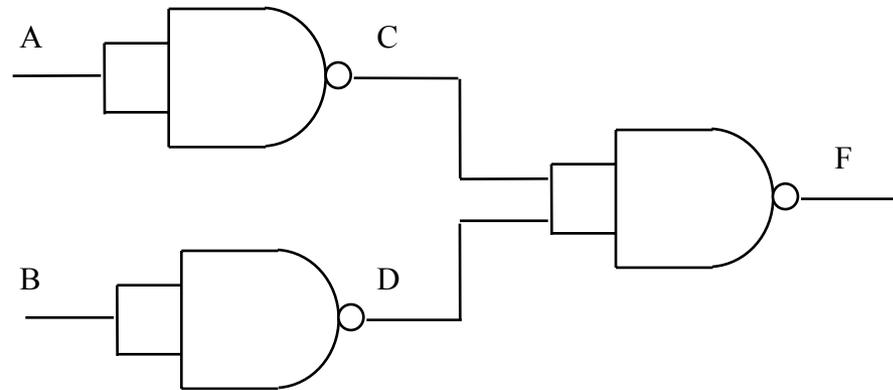
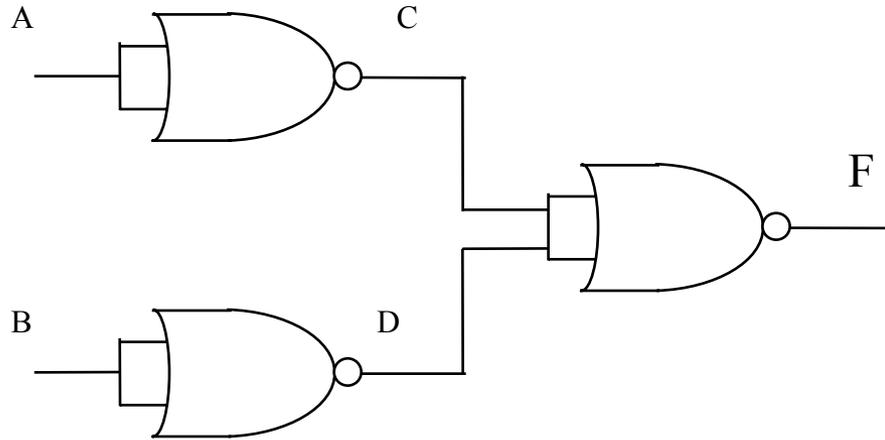
BJT Logic Gates: NAND Gate



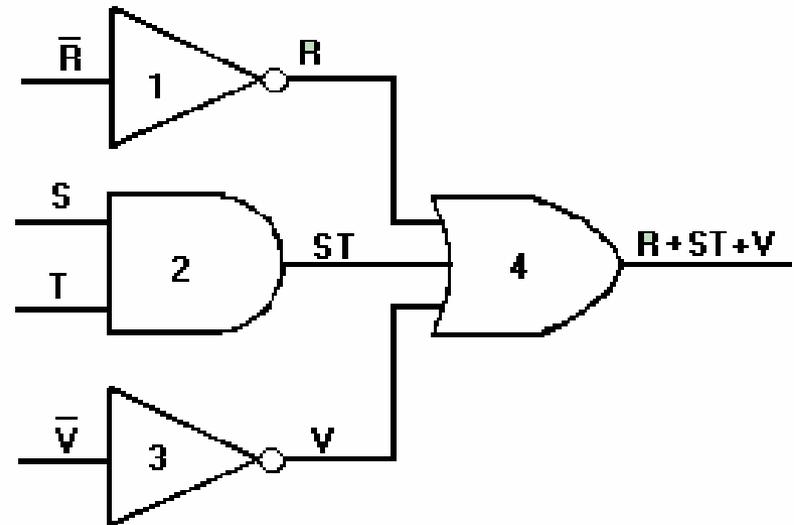
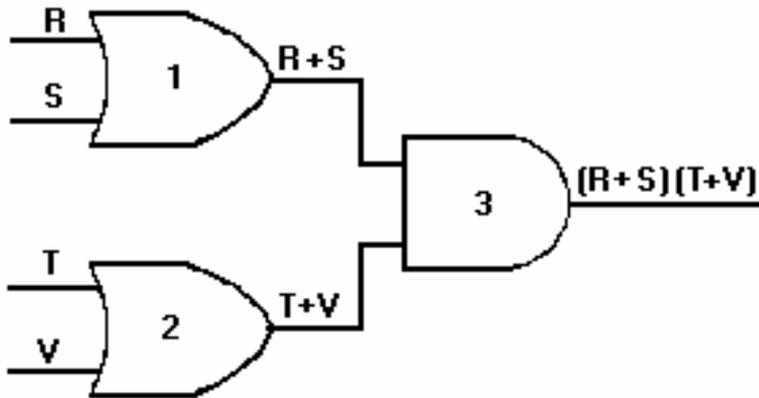
Common Logic Families

- RTL (RESISTOR-TRANSISTOR LOGIC): inputs are applied to resistors, and the output is produced by a transistor.
- DTL (DIODE TRANSISTOR LOGIC): Input signals are applied to diodes in this logic family. The diodes either conduct or cut off and produce the desired output from the transistor.
- TTL (TRANSISTOR-TRANSISTOR LOGIC): transistors with multiple emitters are used for the logic inputs.

Combinational Gates



Examples of Combinations



Boolean Algebra

- Boolean logic, or Boolean algebra as it is called today, was developed by an English mathematician, George Boole, in the 19th century. He based his concepts on the assumption that most quantities have two possible conditions: True or False.
- Boolean algebra is used mainly by design engineers. By using this system, engineers are able to arrange logic gates to accomplish certain objectives. Boolean algebra also enables the engineers to achieve the desired output by using the fewest number of logic gates. Since space, weight, and cost are important factors in the design of equipment, you would usually want to use as few parts as possible.

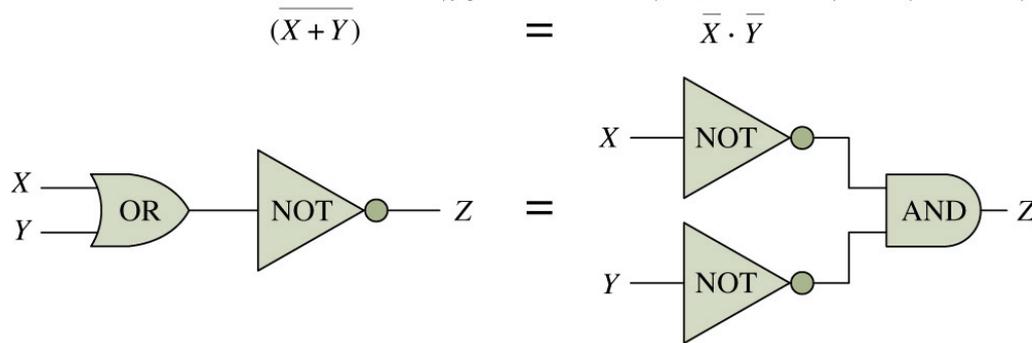
Laws and Theorems

- **LAW OF IDENTITY:** a term that is **TRUE** in one part of an expression will be **TRUE** in all parts of the expression.
- **COMMUTATIVE LAW:-** the order in which terms are written does not affect their value ($AB = BA$, $A+B = B+A$).
- **ASSOCIATIVE LAW:** a simple equality statement $A(BC) = ABC$ or $A+(B+C) = A+B+C$.
- **IDEMPOTENT LAW:** a term **ANDed** with itself or **ORed** with itself is equal to that term ($AA = A$, $A+A = A$).
- **DOUBLE NEGATIVE LAW:** a term that is inverted twice is equal to the term.
- **COMPLEMENTARY LAW:** a term **ANDed** with its complement equals 0, and a term **ORed** with its complement equals 1 ($AA = 0$, $A+A = 1$).
- **LAW OF INTERSECTION:** a term **ANDed** with 1 equals that term and a term **ANDed** with 0 equals 0

De Morgan's laws

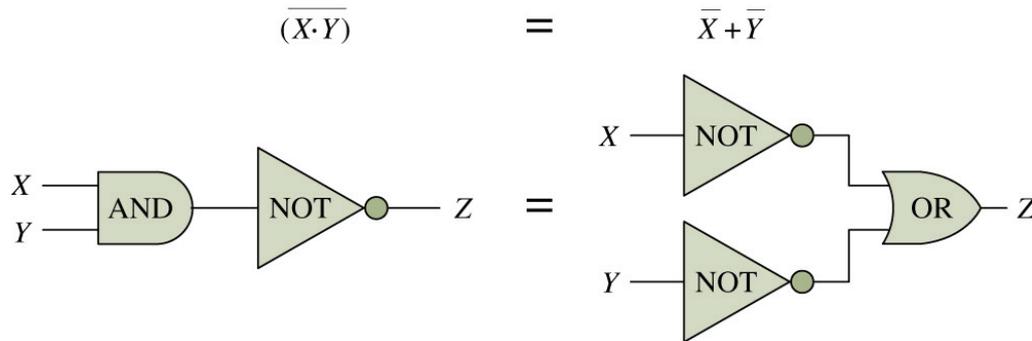
The inverter, often referred to as a NOT gate, is a logic device that has an output opposite of the input.

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.



Truth table

X	Y	$Z = \overline{(X+Y)} = \bar{X} \cdot \bar{Y}$
0	0	1
0	1	0
1	0	0
1	1	0



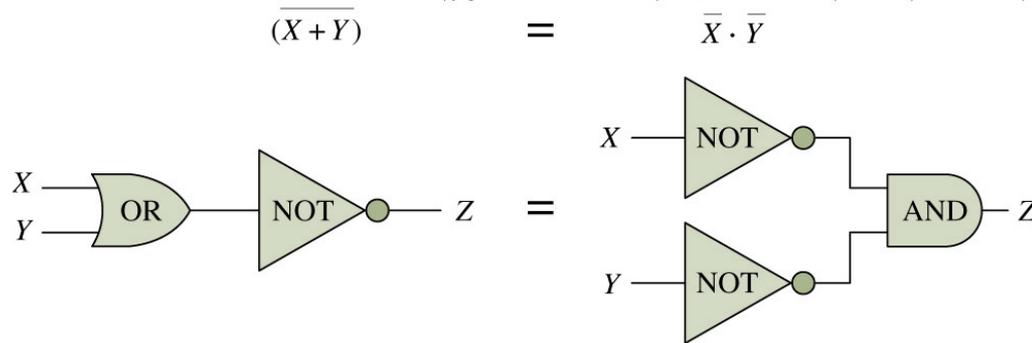
Truth table

X	Y	$Z = \overline{(X \cdot Y)} = \bar{X} + \bar{Y}$
0	0	1
0	1	1
1	0	1
1	1	0

De Morgan's laws

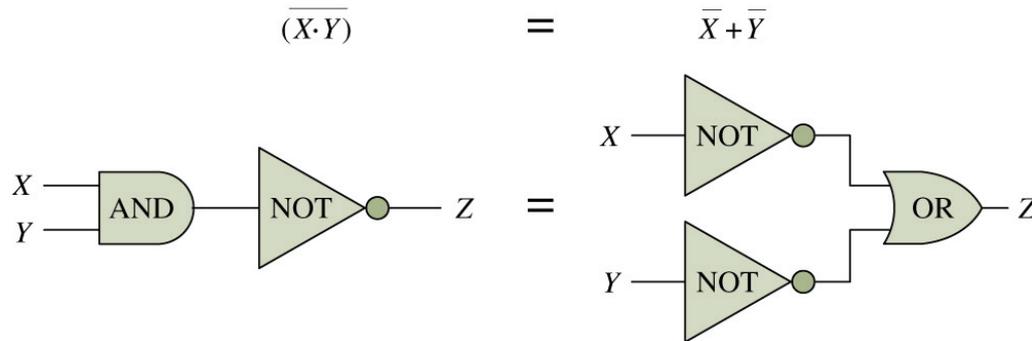
The inverter, often referred to as a NOT gate, is a logic device that has an output opposite of the input.

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.



Truth table

X	Y	$Z = \overline{(X+Y)} = \bar{X} \cdot \bar{Y}$
0	0	1
0	1	0
1	0	0
1	1	0



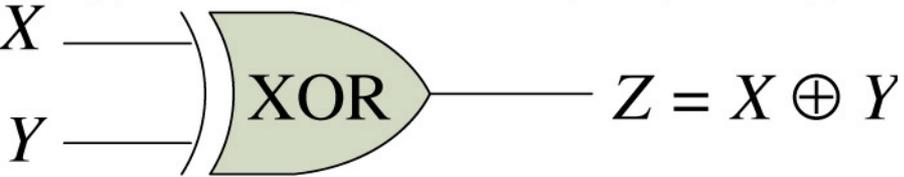
Truth table

X	Y	$Z = \overline{(X \cdot Y)} = \bar{X} + \bar{Y}$
0	0	1
0	1	1
1	0	1
1	1	0

XOR gate

The NAND gate is another logic device commonly found in digital equipment. This gate is simply an AND gate with an inverter (NOT gate) at the output.

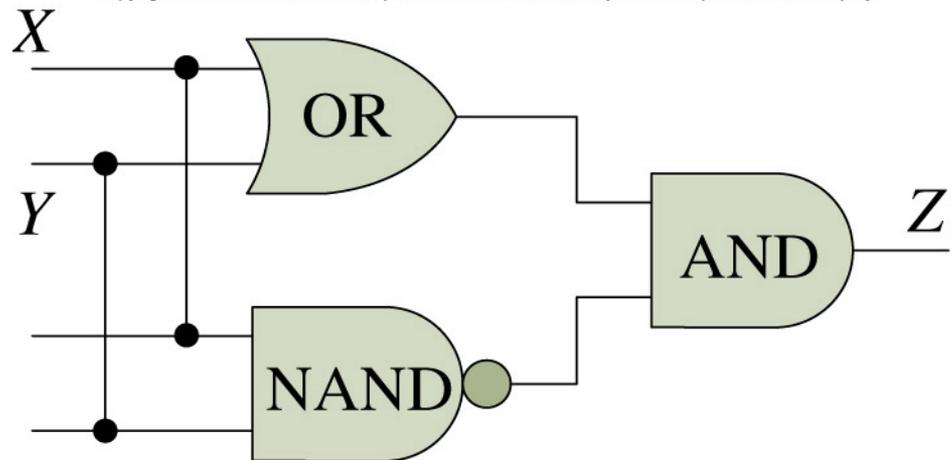
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.



Realization of an XOR gate

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

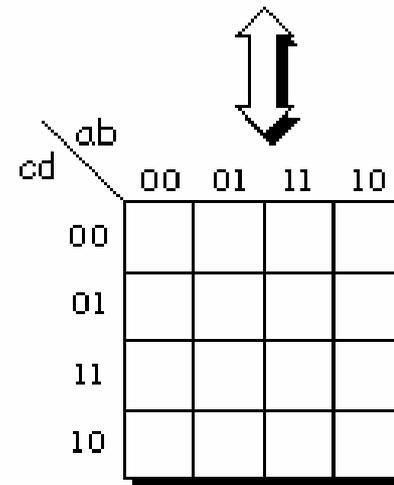
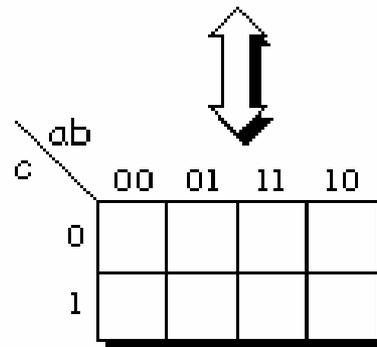
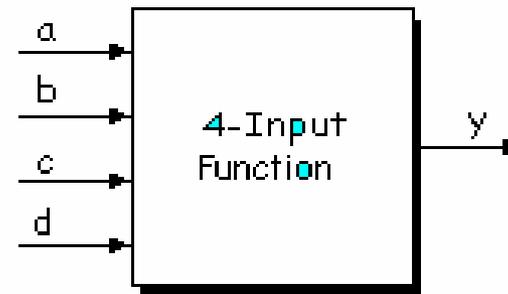
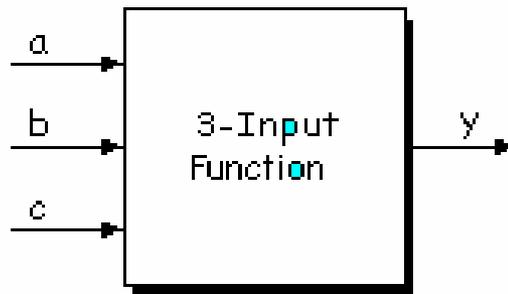


Truth table

Karnaugh Map

Provides an Alternative Technique for Representing Boolean Functions

<http://www.maxmon.com/kmaps1.htm>



Truth table and Karnaugh Map Representations of a Logic Function

Copyright © The McGraw-Hill Companies, Inc.
Permission required for reproduction or display.

Read Examples: 13.10; 13.12; and 13.14

	$\overline{X}Y$	$X\overline{Y}$	XY	$\overline{X}\overline{Y}$
\overline{Z}	0	1	1	0
Z	0	1	1	0

Karnaugh map

X	Y	Z	Desired Function
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Truth table

Karnaugh map for a four-variable expression

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

X	Y	Y	Z	Desired Function
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

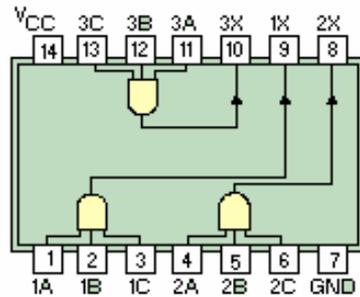
Truth table for four-variable expression

	$\bar{W} \cdot \bar{X}$	$\bar{W} \cdot X$	$W \cdot X$	$W \cdot \bar{X}$
$\bar{Y} \cdot \bar{Z}$	1	0	0	0
$\bar{Y} \cdot Z$	1	1	0	1
$Y \cdot Z$	0	0	1	0
$Y \cdot \bar{Z}$	0	1	0	1

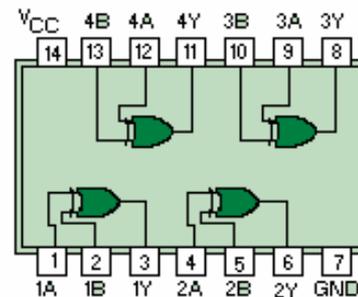
Logic Families

Logic families are groups of logic circuits that are based on particular types of elements (resistors, transistors, and so forth). Families are identified by the manner in which the elements are connected, and, in some cases, by the types of elements used.

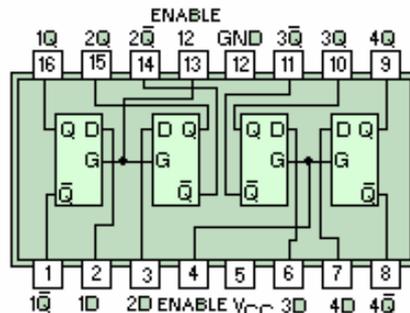
<http://www.tpub.com/neets/book13/55k.htm>



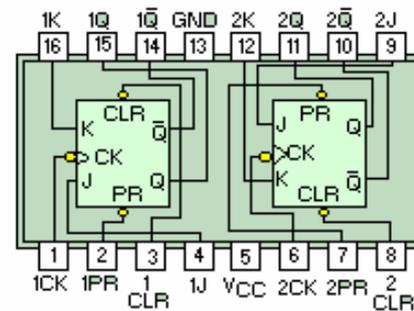
TRIPLE THREE-INPUT AND



QUAD EXCLUSIVE-OR



4-BIT LATCH (D REGISTER)

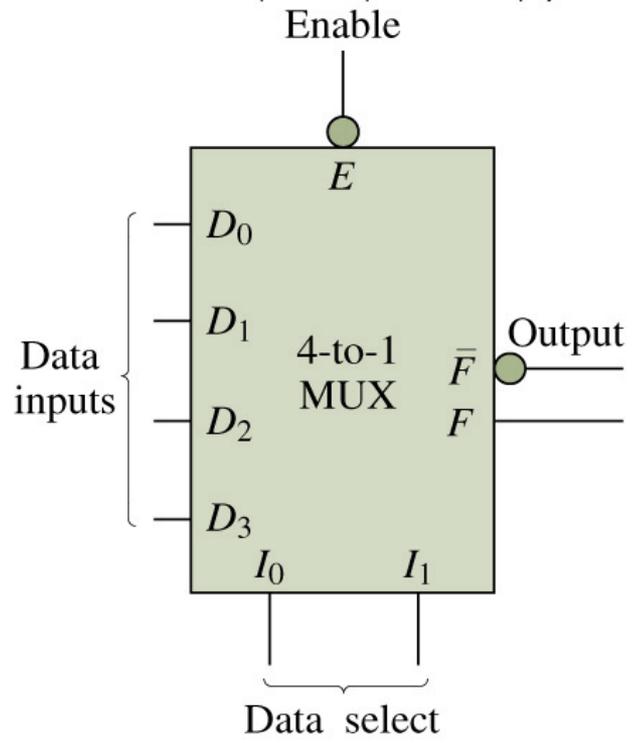
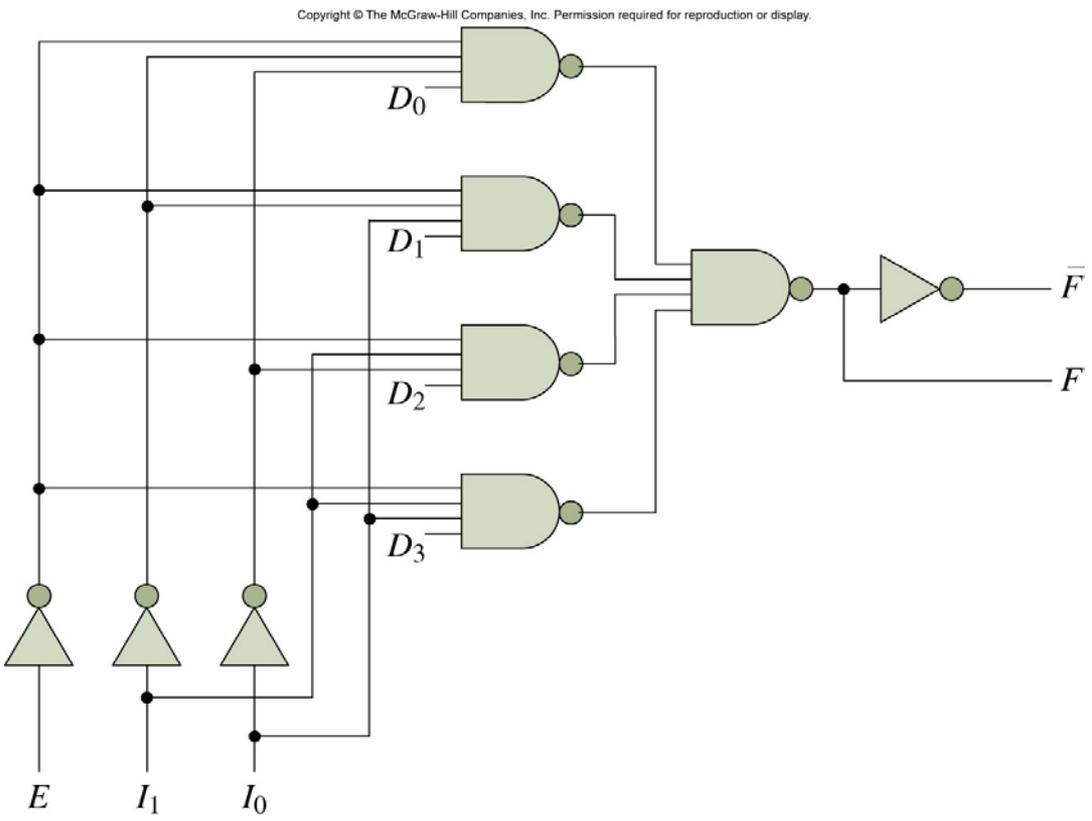


DUAL J-K FLIP-FLOP

Combinational Logic Gates: 4-to-1 Multiplexer

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Multiplexer (MUX), or data selectors, are combinational logic circuits that permit the selection of many inputs. A typical MUX has 2^n data lines, n address (data select) lines and one output. Also, other control inputs may exist.



Block diagram of 4-to-1 MUX

I_1	I_0	F
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

Truth table of 4-to-1 MUX

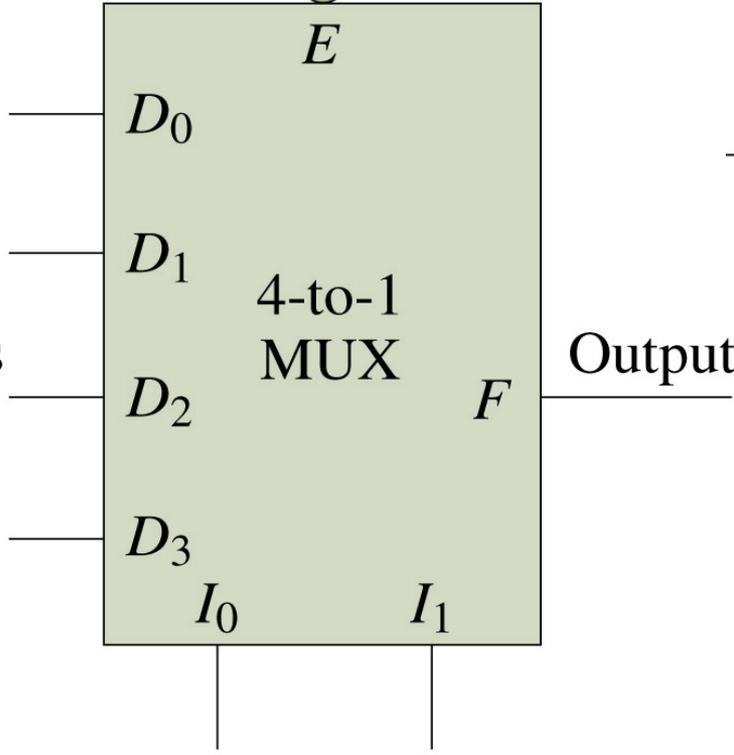
Functional diagram of four-input MUX

Copyright © The McGraw-Hill Companies, Inc.
Permission required for reproduction or display.

Enable



E



Data inputs

Output

Select lines

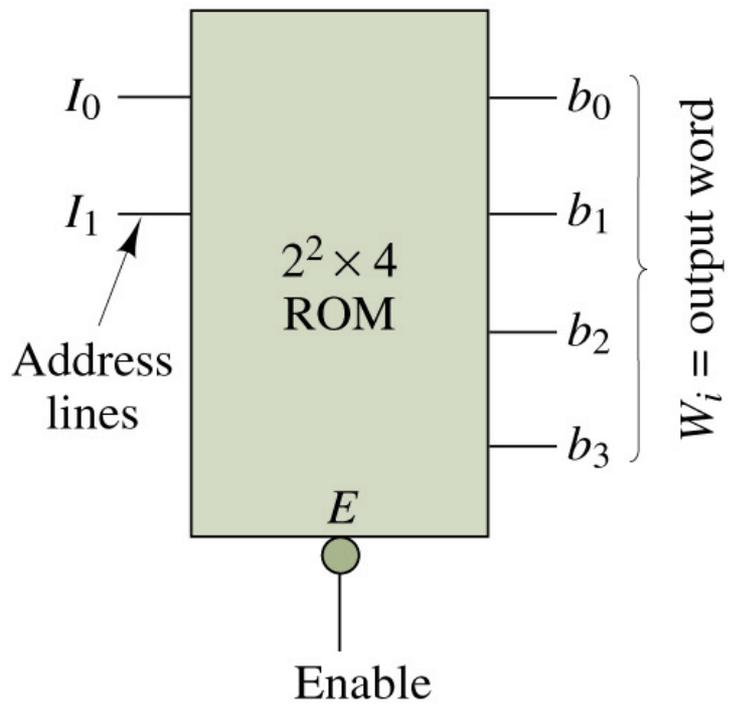
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

I_1	I_0	D_3	D_2	D_1	D_0	F
0	0	d	d	d	0	0
0	0	d	d	d	1	1
0	1	d	d	0	d	0
0	1	d	d	1	d	1
1	0	d	0	d	d	0
1	0	d	1	d	d	1
1	1	0	d	d	d	0
1	1	1	d	d	d	1

Read-only memory

ROM is a logic circuit that holds in storage (memory) information in the form of binary numbers-that cannot be altered by can be read by logic circuit. A ROM is an array of memory cells, each of which can store either a 1 or a 0. The array consists of $2^m \times n$ cells, where n is the number of bits in each word stored in ROM, m address lines are required. To access the information stored in ROM, m address lines are required. When address is selected, the binary word corresponding to the address selected appears at the output, which consists of n bits, that is, the same number of bits as the stored words.

ROM address		ROM content (4-bit words)				
I_1	I_0	b_3	b_2	b_1	b_0	
0	0	0	1	1	0	W_0
0	1	1	0	0	1	W_1
1	0	0	1	1	0	W_2
1	1	1	1	1	1	W_3



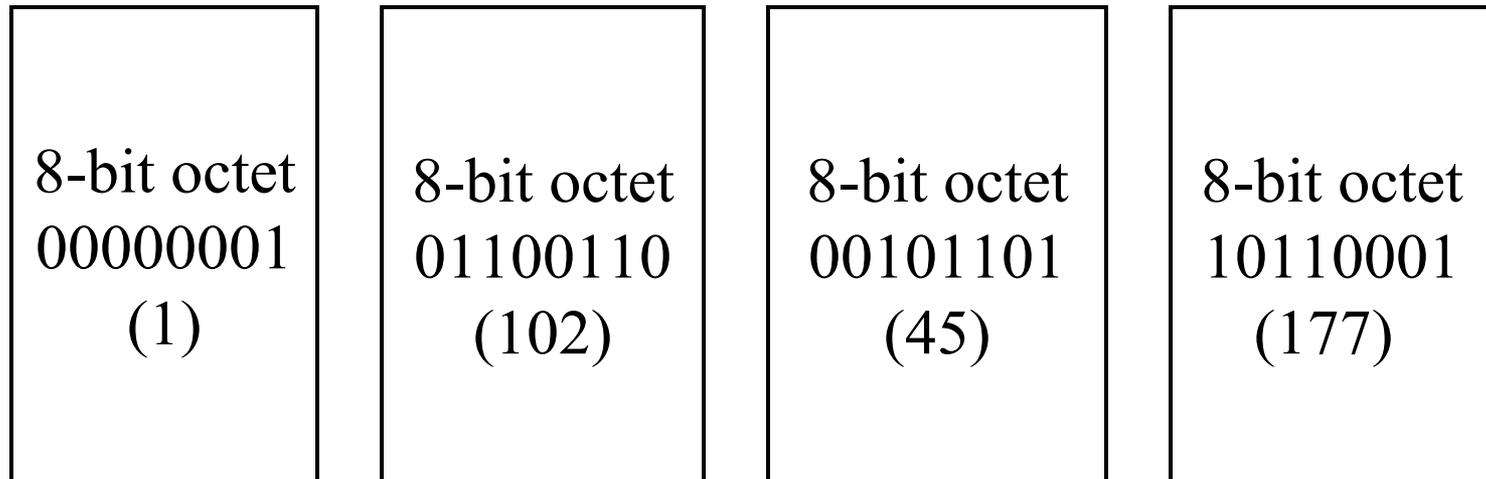
Application: Internet Protocol (IP)

- The Internet Protocol (IP) determine the addressing system that connects all computers to the Internet. The IP addressing scheme provides logical addressing of data packets so they can be routed both between local networks and throughout the Internet.
- The IP address is made up of two parts, the network portion and the host (computer portion). The network portion routes data packets between networks while the host portion determines the location of the particular device within a network.
- An IP address is a 32-bit number that is divided into four groups called octets. Each octet contains eight binary numbers (a byte) consisting of 1s and 0s. Because of the difficulty in remembering a bit pattern of a 32-bit binary number, IP addressing is simplified through the use of a numbering system called *decimal notation*.

00000001.01100110.00101101.10110001

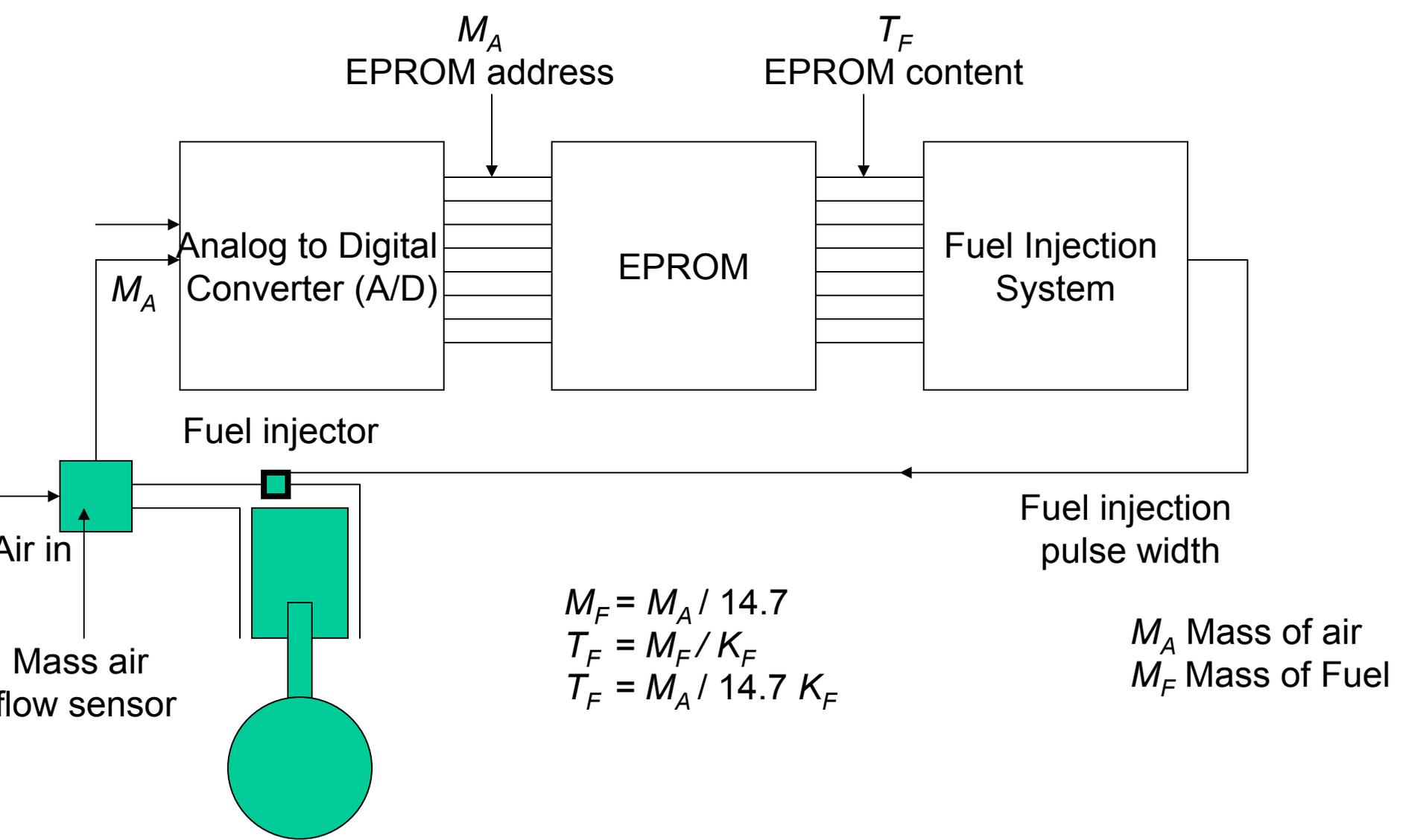
1.102.45.177

- An IP address with four 1-byte (8-bit) octets



Homework

- A microwave oven requires that the main power supply be on, the door be latched, and the timer set to a time other than zero before the oven will function. Draw a logic diagram for the operation of the oven.
- A lighter is being sold that uses a light beam to light the lighter. The lighter has a cutout on its side that has been fixed with a light source and a light sensor. Opening the lid of the lighter activates the light source; interrupting the light source causes the lighter to light. Draw a logic diagram for the function of the lighter.
- An automatic parking lot gate extends a ticket to the customer when the customer's car drives over a sensor S and the driver presses button B on the ticket vendor. Draw the logic diagram to control the release of the ticket. Once the extended ticket is taken from dispenser D , the arm A will raise to let the car through. Draw the logic diagram that control the raising of the arm.



Application: Automotive Fuel Injection System Control