IR Sensor

IR sensor is an electronic device, that emits the light in order to sense some object of the surroundings.

An IR sensor can detects the motion.

The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode .

Photodiode is sensitive to IR light which is emitted by the IR LED.

When IR light falls on the photodiode, the resistances and the output voltages will change in proportion to the magnitude of the IR light received.

IR Sensor

IR Sensor Working Principle

IR Transmitter or IR LED

Infrared Transmitter is a light emitting diode (LED) which emits infrared radiations called as IR LED’s. Even though an IR LED looks like a normal LED, the radiation emitted by it is invisible to the human eye.

The picture of an Infrared LED is shown below.

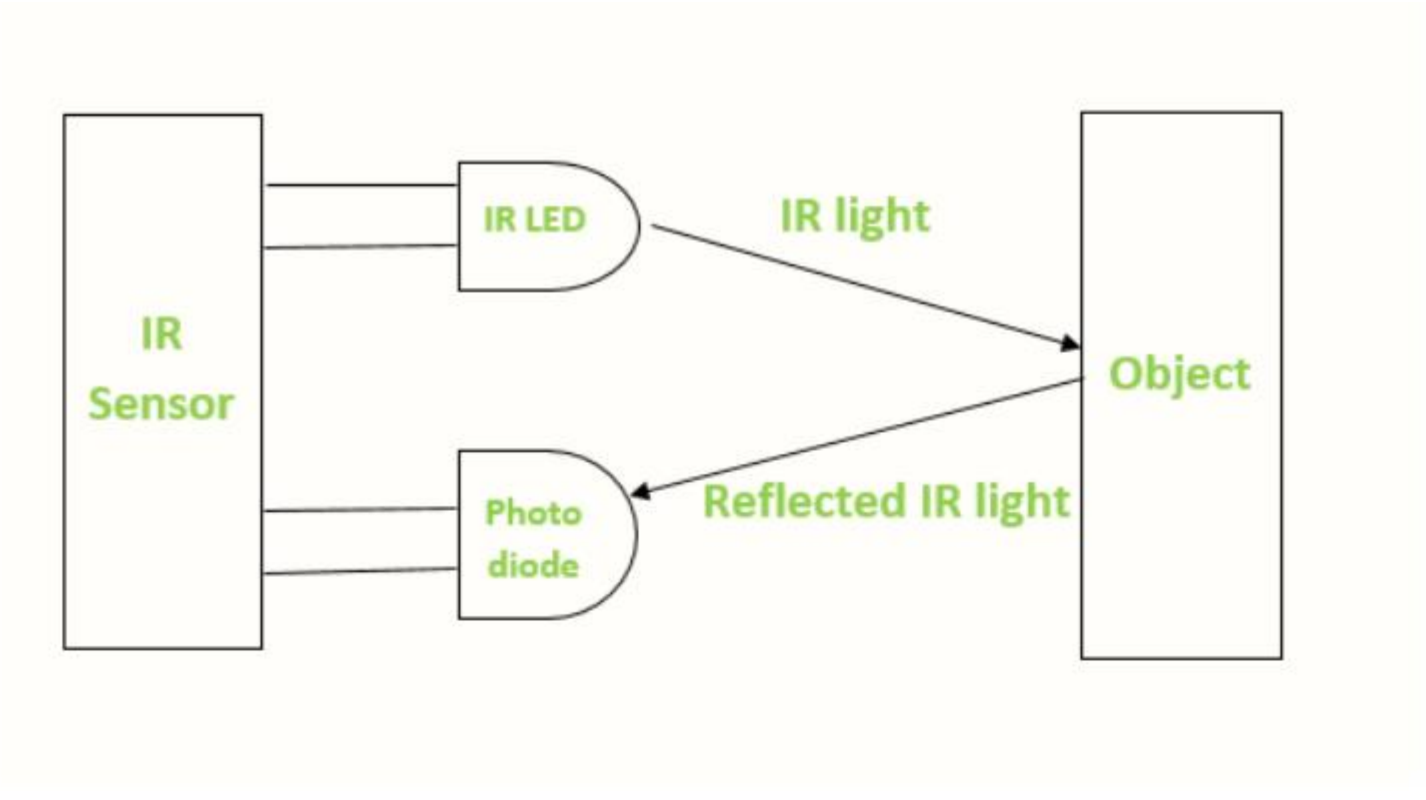
IR Receiver

Infrared receivers or infrared sensors detect the radiation from an IR transmitter.

IR receivers come in the form of photodiodes.

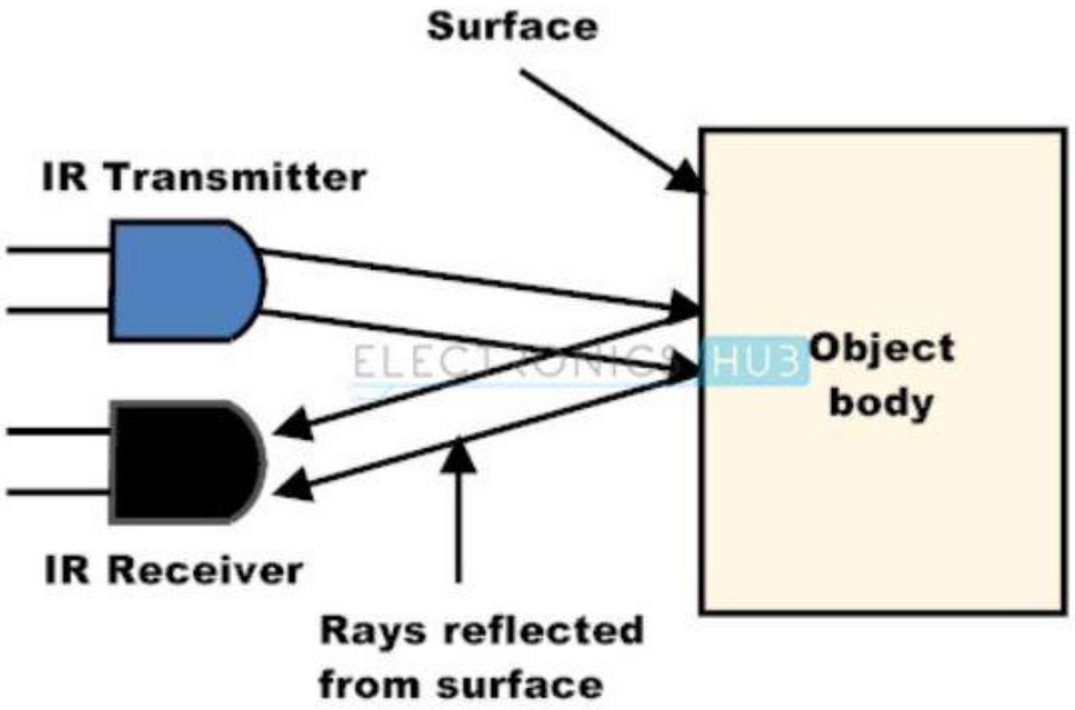
Infrared Photodiodes are different from normal photo diodes as they detect only infrared radiation. Below image shows the picture of an IR receiver or a photodiode,

IR Sensor Working Principle



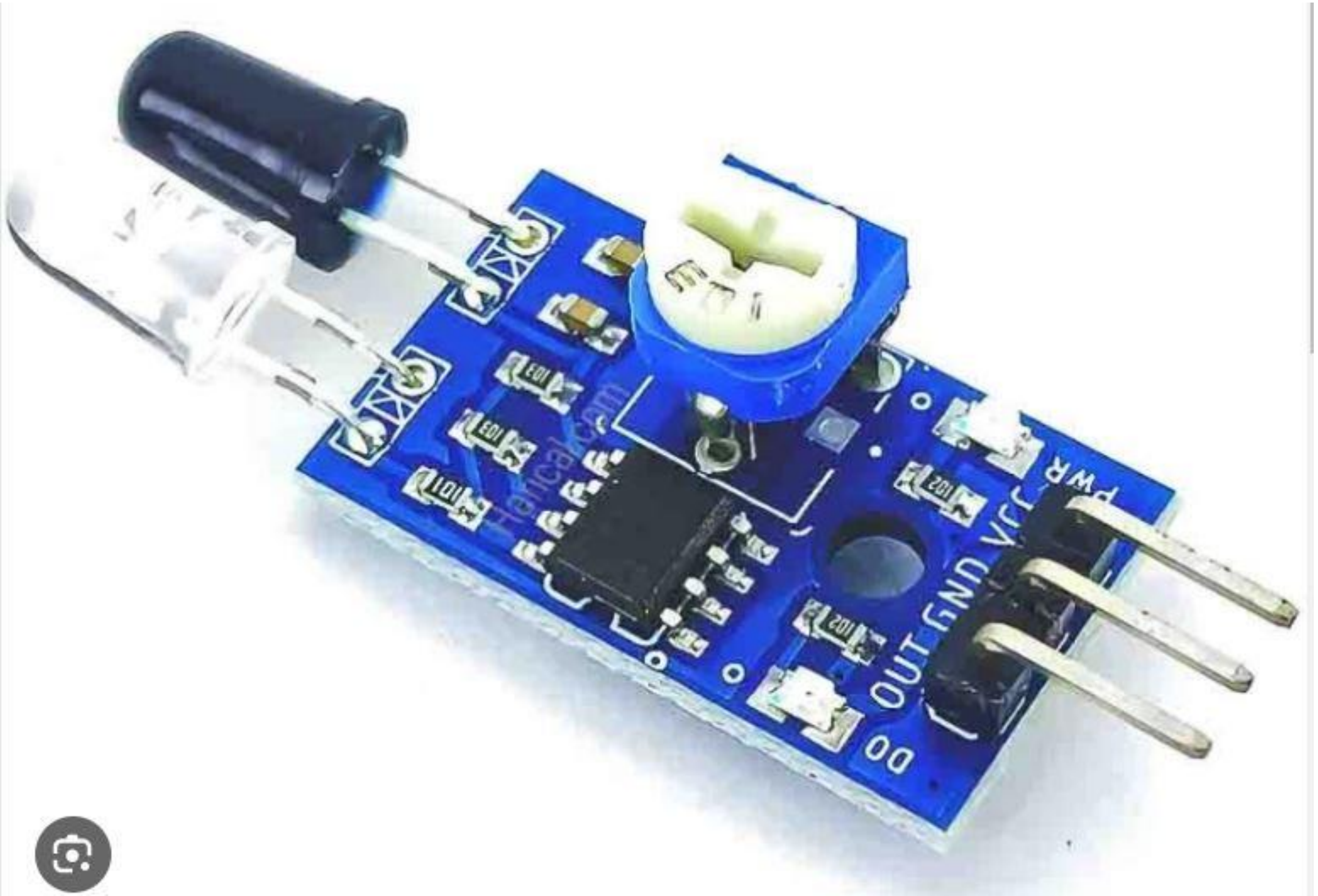
When the IR transmitter emits radiation, it reaches the object and some of the radiation reflects back to the IR receiver. Based on the intensity of the reception by the IR receiver, the output of the **sensor** defines.

IR Sensor

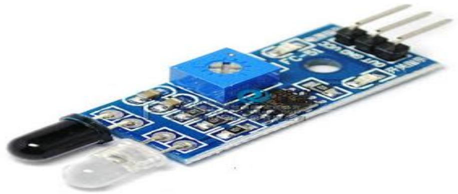


Surface
IR Transmitter
Object body
IR Receiver
Rays reflected from surface

IR Sensor



InfraRed (IR) Sensor

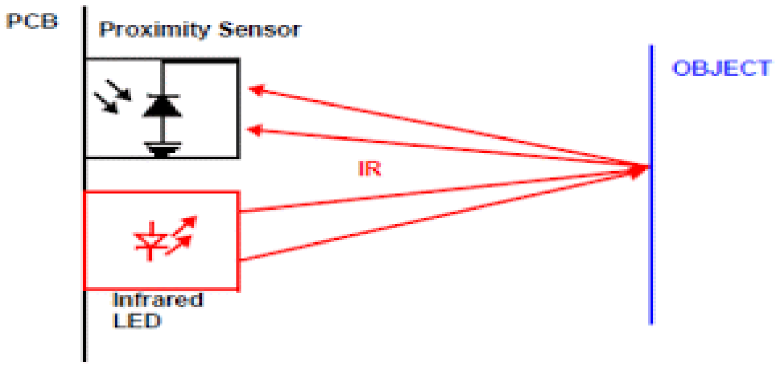


Features-

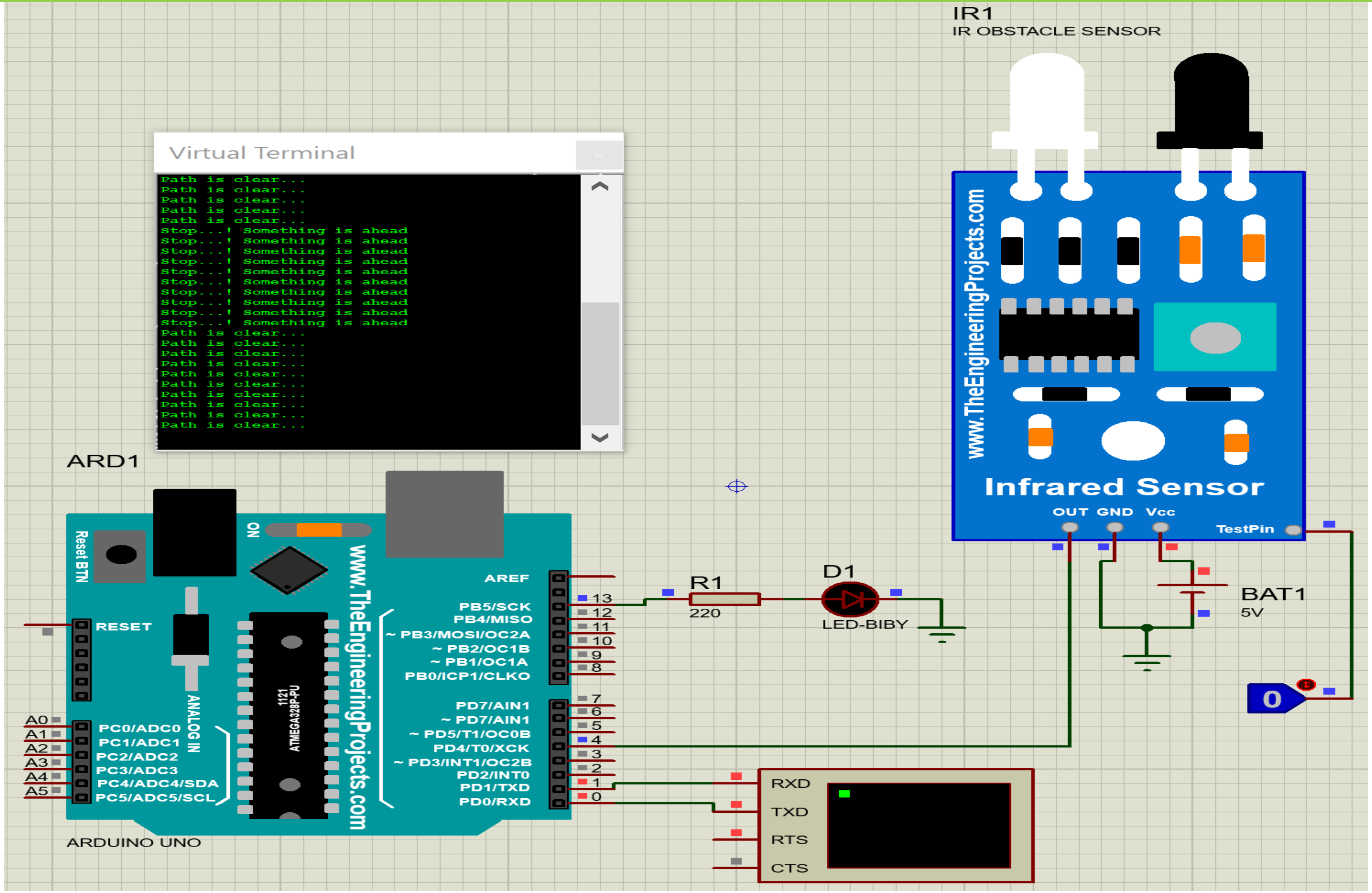
- Can be used for obstacle sensing, fire detection, line sensing, etc
- Input Voltage: 5V DC
- Comes with an easy to use digital output
- Can be used for wireless communication and sensing IR remote signals

IR Sensor have three Pins

1. VCC = +5V DC
2. GND
3. D0 or OUT (Digital Output)



IR Sensor


IR1
IR OBSTACLE SENSOR
Virtual Terminal
Path is clear...
Stop...! Something is ahead
ARD1
www.TheEngineeringProjects.com
Infrared Sensor
OUT GND Vcc
TestPin
AREF
R1
220
D1
LED-BIBY
BAT1
5V
RESET
ANALOG IN
ATMEGA328P-PU
ARDUINO UNO
RXD
TXD
RTS
CTS

DHT-11 Digital Temperature And Humidity Sensor

DHT11 sensor measures and provides humidity and temperature values serially over a single wire.

It can measure relative humidity in percentage (20 to 90%) and temperature in degree Celsius in the range of 0 to 50°C.

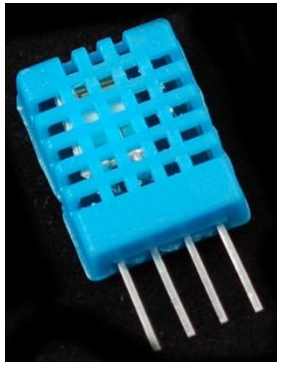
It has 4 pins; one of which is used for data communication in serial form.

Pin diagram of DHT11

Pin No.	Pin Name	Pin Description
1	VCC	Power supply
2	DATA	Digital output pin
3	NC	Not in use
4	GND	Ground

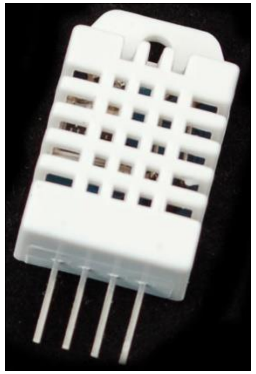
DHT11 Specifications

- Ultra-low cost
- 3 to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)
- Good for 20-80% humidity readings with 5% accuracy
- Good for 0-50°C temperature readings ±2°C accuracy



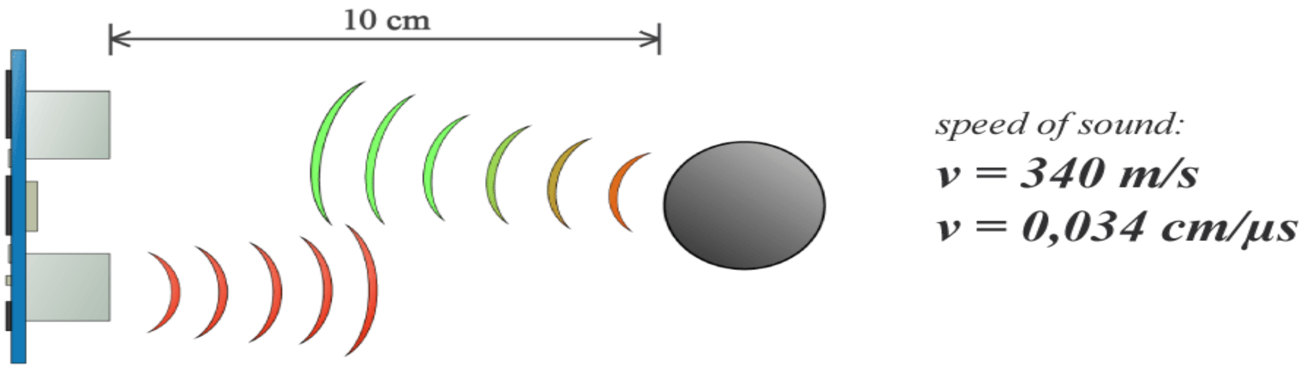
DHT22 Specifications

- Low cost
- 3 to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)
- Good for 0-100% humidity readings with 2-5% accuracy
- Good for -40 to 125°C temperature readings ±0.5°C accuracy



Ultrasonic Sensor

- An Ultrasonic sensor is a device that can <u>measure the distance to an object by using sound waves</u>.
- It measures distance by sending out a sound wave at a specific frequency and listening for that sound wave to bounce back.
- <u>By recording the elapsed time between the sound wave being generated and the sound wave bouncing back,</u> it is possible to calculate the distance between the sonar sensor and the object.



$$distance = \frac{speed\ of\ sound\ \times time\ taken}{2}$$

Distance:

$$s = t \cdot 0{,}034 / 2$$

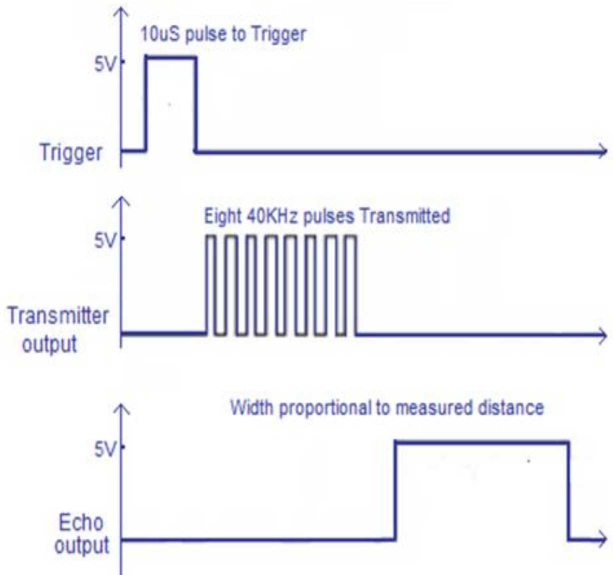
Working principle

The <u>Trig pin</u> will be used to send the signal and
the <u>Echo pin</u> will be used to listen for returning signal

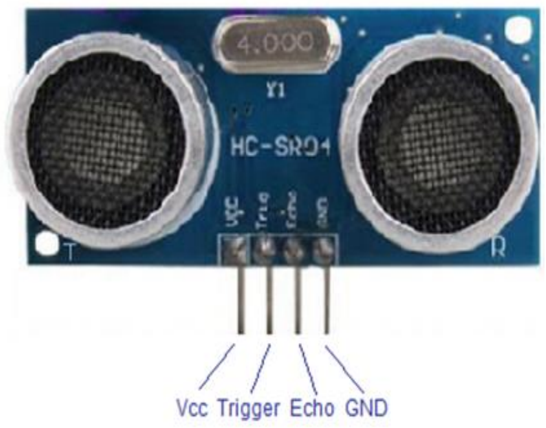
(1) Using IO trigger for <u>at least 10us high level signal</u>, Trig -> Pin-9 (o/p) of Arduino

(2) The Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.

(3) IF the signal back, through high level , time of high output IO duration is the time from sending ultrasonic to returning.



HC-SR04 Timing diagram



HC-SR04 Pinout

$$distance = \frac{speed\ of\ sound\ \times time\ taken}{2}$$

Working Voltage	DC 5 V
Working Current	15mA
Working Frequency	40Hz
Max Range	4m
Min Range	2cm
MeasuringAngle	15 degree
Trigger Input Signal	10uS TTL pulse
Echo Output Signal	Input TTL lever signal and the range in proportion
Dimension	45*20*15mm

Arduino Data Types

- But sometimes you do need to choose a type that specifically suits your application.

Numeric types	Bytes	Range	Use
`int`	2	-32768 to 32767	Represents positive and negative integer values.
`unsigned int`	2	0 to 65535	Represents only positive values; otherwise, similar to `int`.
`long`	4	-2147483648 to 2147483647	Represents a very large range of positive and negative values.
`unsigned long`	4	4294967295	Represents a very large range of positive values.

Arduino Data Types (Cont..)

Numeric types	Bytes	Range	Use
`float`	4	3.4028235E+38 to -3.4028235E+38	Represents numbers with fractions; use to approximate real-world measurements.
`double`	4	Same as `float`	In Arduino, `double` is just another name for `float`.
`boolean`	1	`false` (0) or `true` (1)	Represents true and false values.
`char`	1	-128 to 127	Represents a single character. Can also represent a signed value between -128 and 127.
`byte`	1	0 to 255	Similar to `char`, but for unsigned values.
Other types			
`string`	Represents arrays of `chars` (characters) typically used to contain text.		
`void`	Used only in function declarations where no value is returned.		

Data Types and Operators

Integer: used with integer variables with value between 2147483647 and -2147483647.

Eg: int x=1200;

Character: used with single character, represent value from - 127 to 128.

Eg. char c='r';

Long: Long variables are extended size variables for number storage, and store 32 bits (4 bytes), from -2,147,483,648 to 2,147,483,647.

Eg. long u=199203;

Floating-point numbers can be as large as 3.4028235E+38 and as low as -3.4028235E+38. They are stored as 32 bits (4 bytes) of information.

Eg. float num=1.291; [The same as **double** type]

Statements and operators:

Statement represents a command, it ends with **;**

Eg: **int x; x=13;**

Operators are symbols that used to indicate a specific function:

Math operators: [**+,-,*,/,%,^**]

Logic operators: [**==, !=, &&, ||**]

Comparison/Boolean operators: [**==, >, <, !=, <=, >=**]

Syntax:

; Semicolon, **{}** curly braces,

// single line comment,

/*Multi-line comments****/***

Boolean Operators:

== (is equal?)

!= (is not equal?)

\> (greater than)

\>= (greater than or equal)

< (less than)

<= (less than or equal)

Compound Operators:

++ (increment)

-- (decrement)

+= (compound addition)

-= (compound subtraction)

*= (compound multiplication)

/= (compound division)

Control statements:

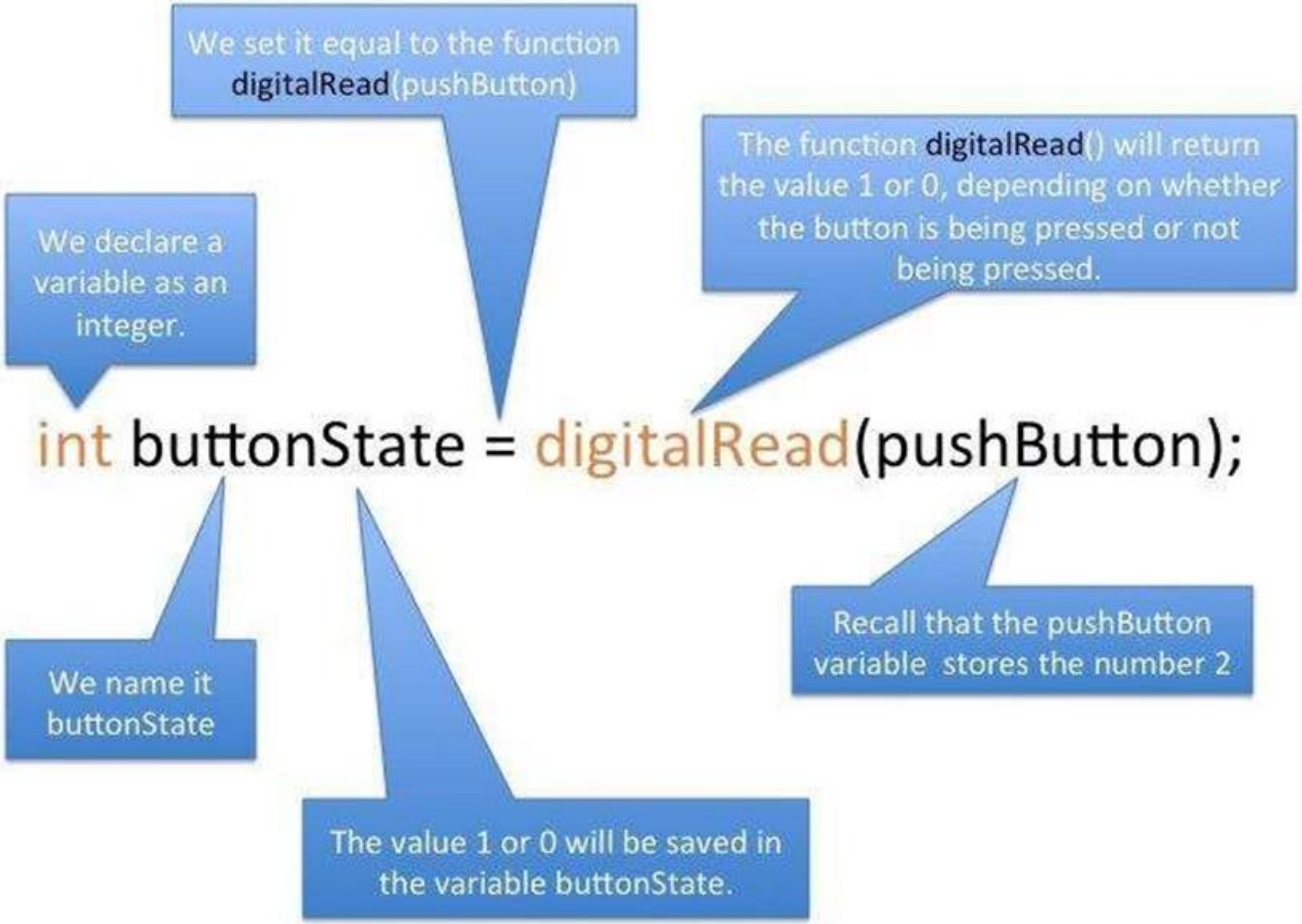
If Conditioning:

```
if(condition)
{
statements-1;
…
Statement-N;
}
else if(condition2)
{
Statements;
}
else
{
statements;
}
```

Switch case:

```
switch (var)
{
        case 1:
//do something when var equals 1  break;
case 2:
//do something when var equals 2  break;
default:
// if nothing else matches, do the default
// default is optional
}
```

Digital Input (Cont..)



Combinational Circuits

- Combinational circuit is a circuit in which we combine the different gates in the circuit, for example encoder, decoder, multiplexer and demultiplexer.

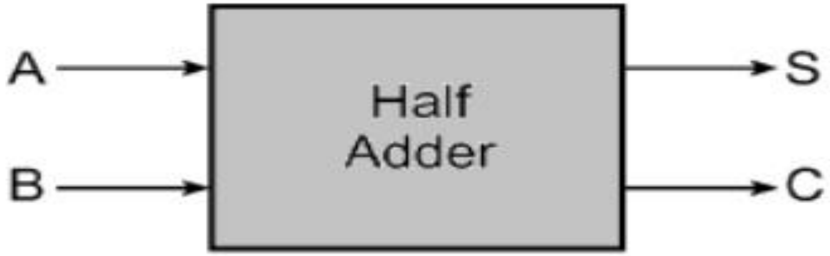
- Some of the characteristics of combinational circuits are following −

- The output of combinational circuit at any instant of time, depends only on the levels present at input terminals.

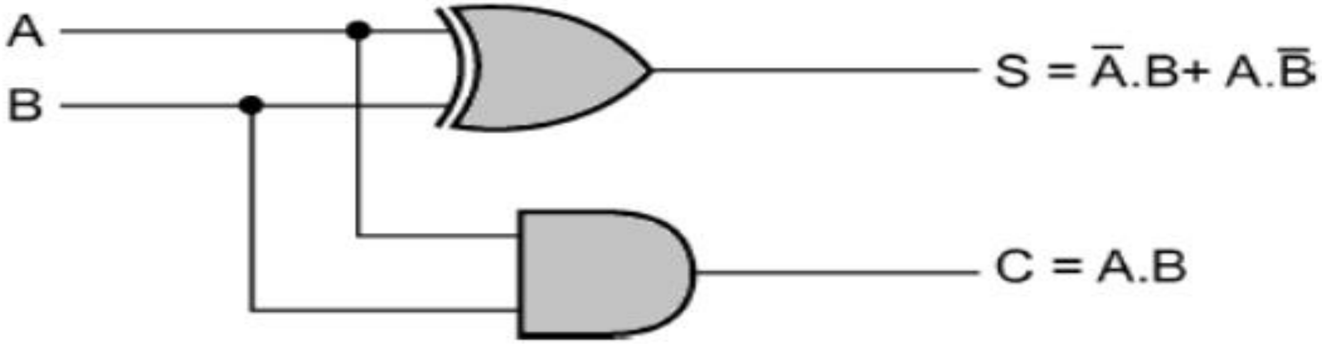
Adders

- Addition of binary bits is the most basic operation.

1) Half Adder:- Half Adder is a combination logic circuit with two inputs and two outputs.
2) It is the basic building blocks for addition of two single bit numbers.
3) This circuit has two outputs mainly- SUM and CARRY.

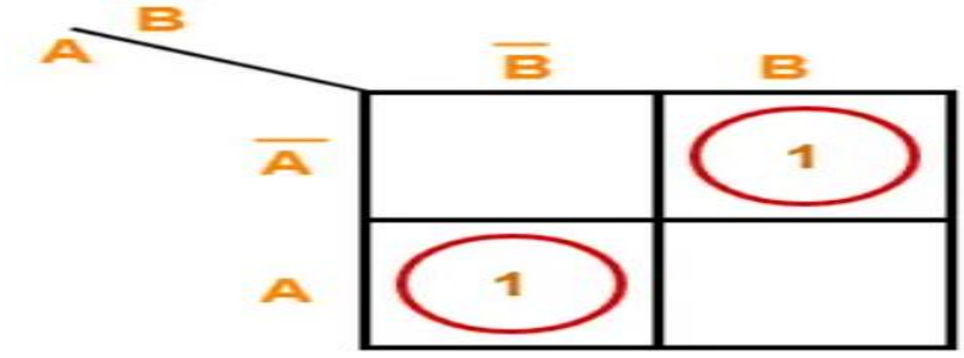


A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



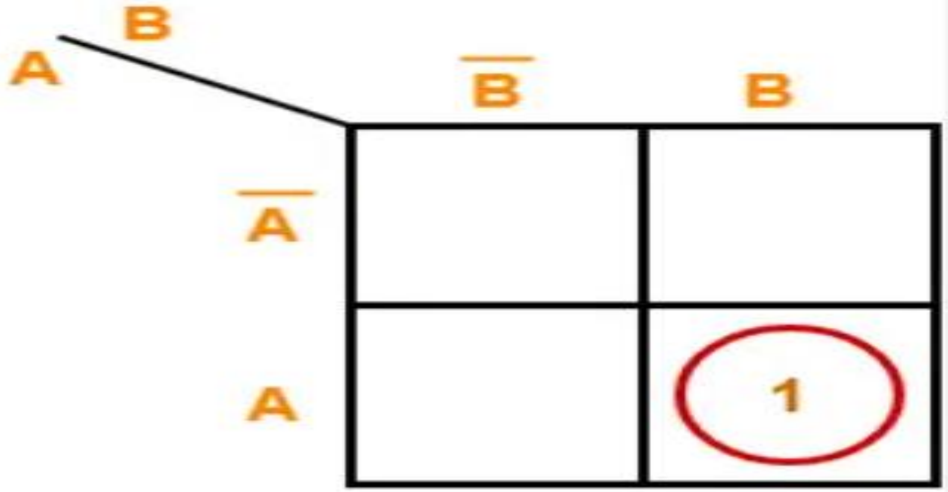
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

For S:



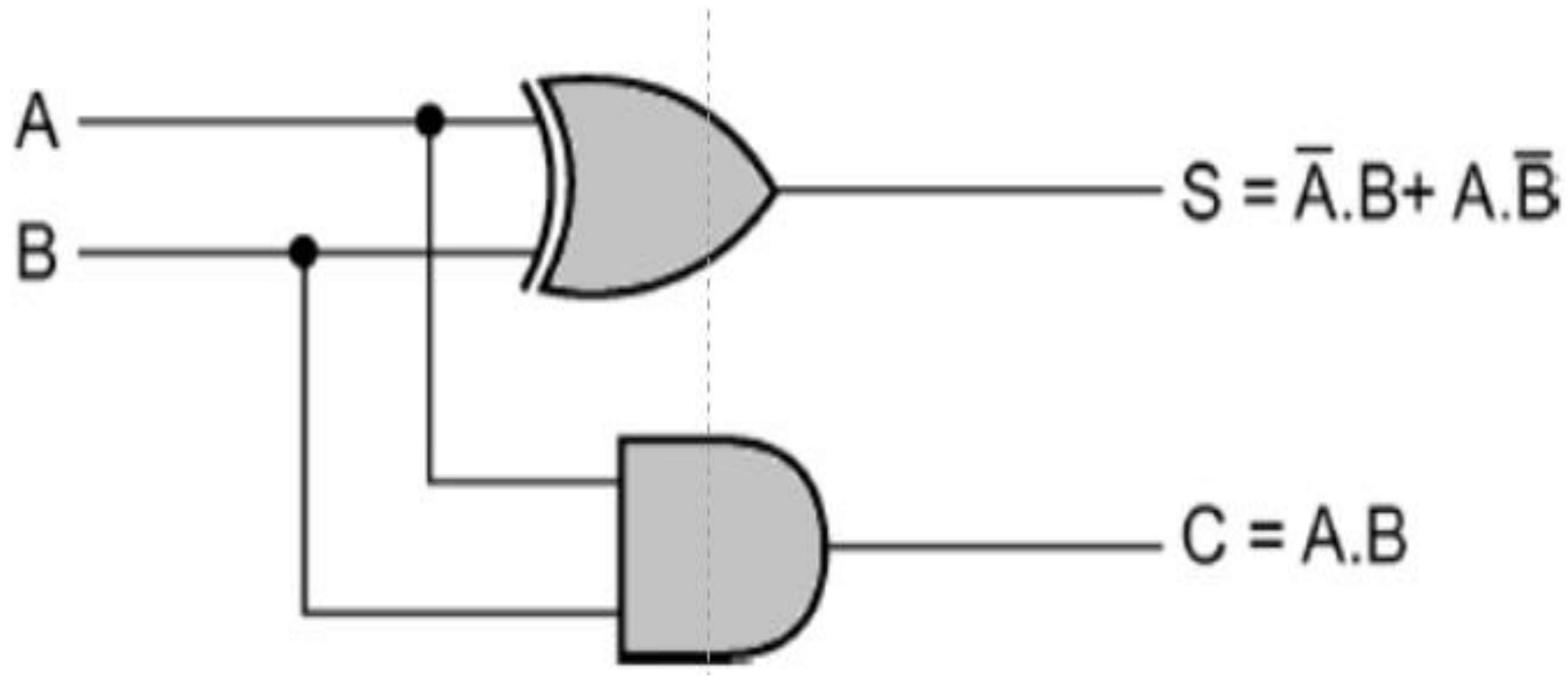
$\boxed{S = A \oplus B}$

For C:



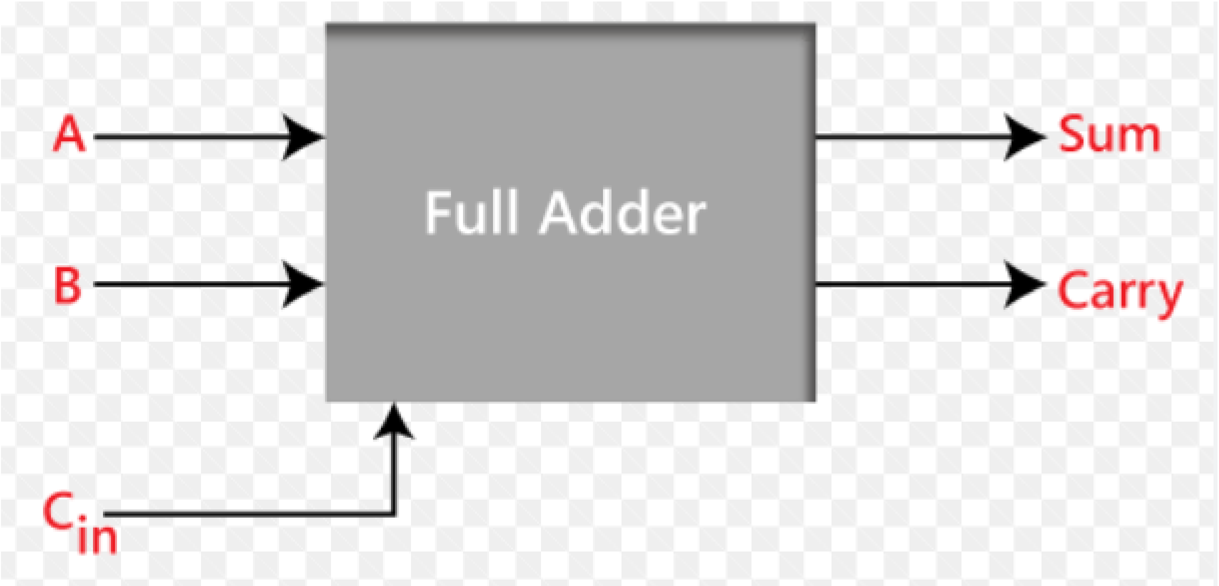
$\boxed{C = A . B}$

Diagram



Full adder

1. Full adder is a 3 single bits adder circuit.
2. It can add two one bit no. A&B and carry Cin.
3. The full adder is a three inputs and two outputs combinations circuits.



Truth table

Inputs			Outputs	
A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Kmap

For sum

A \ BC	00	01	11	10
0		1		1
1	1		1	

$$\therefore \text{Sum} = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$$

$$= \bar{A}(\bar{B}C + B\bar{C}) + A(\bar{B}\bar{C} + BC)$$

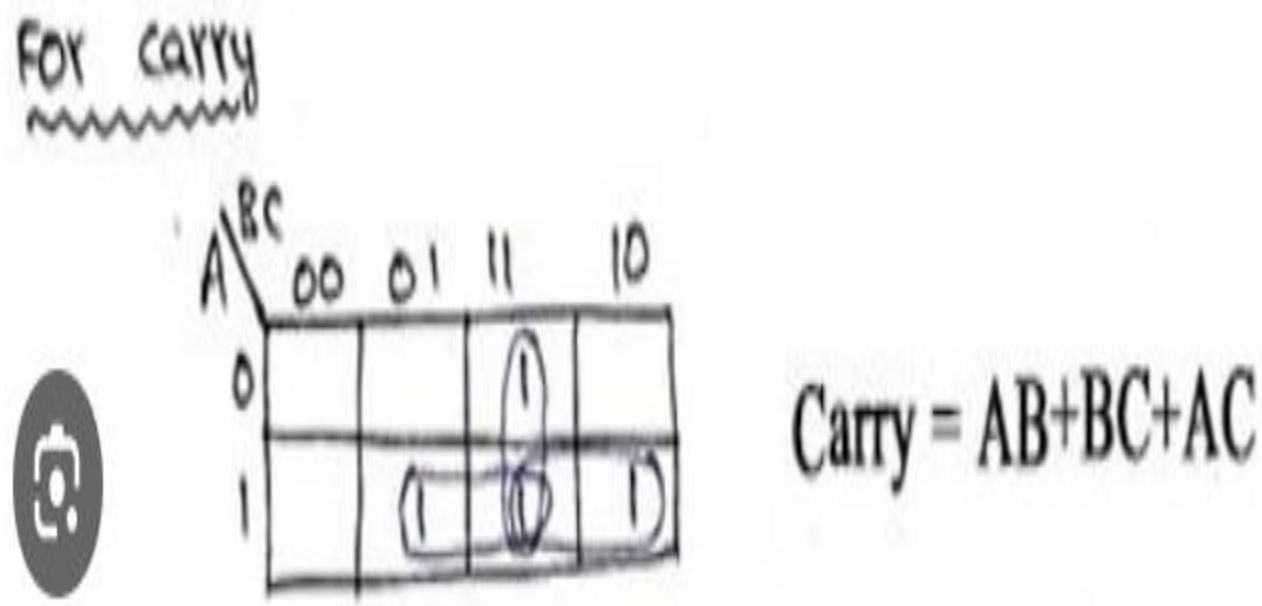
$$= \bar{A}(B \oplus C) + A(\overline{B \oplus C}) \qquad (\because \bar{x}y + x\bar{y} = x \oplus y)$$

$$= A \oplus B \oplus C$$

Inputs			Outputs	
A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

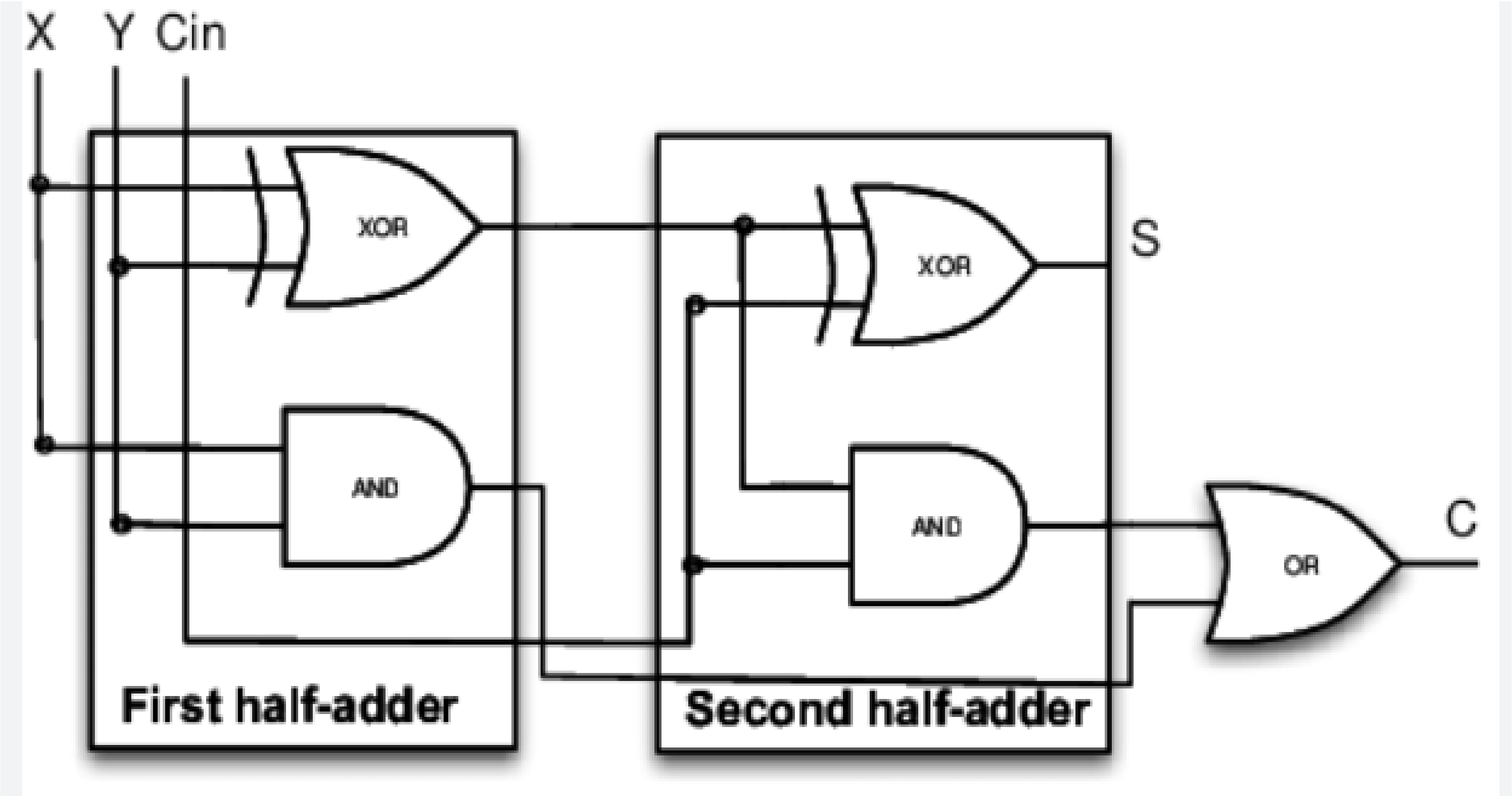
723 × 349

Kmap



Inputs			Outputs	
A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Full adders using half adders



Carry calculations

$$C_o = (A \oplus B)C_{in} + AB$$

$$\Rightarrow \bar{A}BC_{in} + A\bar{B}C_{in} + AB(1 + C_{in})$$

$$\Rightarrow \bar{A}BC_{in} + A\bar{B}C_{in} + AB + ABC_{in}$$

$$\Rightarrow \bar{A}BC_{in} + ABC_{in} + A\bar{B}C_{in} + AB$$

$$\Rightarrow BC_{in}(\bar{A} + A) + A\bar{B}C_{in} + AB$$

$$\Rightarrow BC_{in} + A\bar{B}C_{in} + AB(1 + C_{in})$$

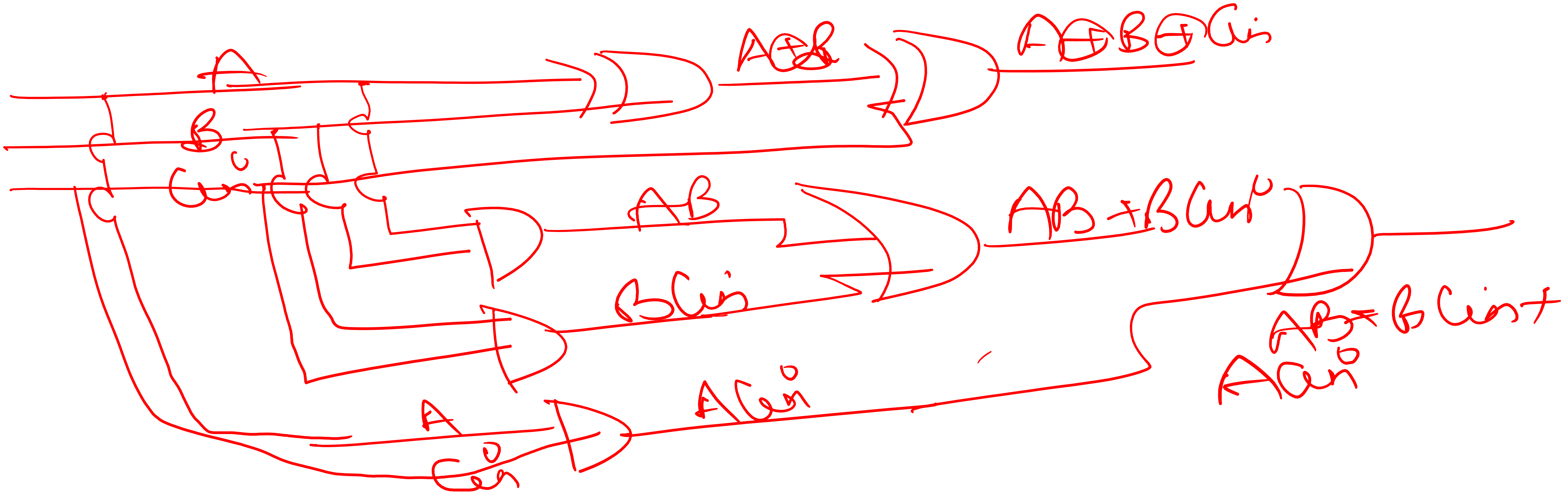
$$\Rightarrow BC_{in} + A\bar{B}C_{in} + AB + ABC_{in}$$

$$\Rightarrow BC_{in} + AB + AC_{in}(B + \bar{B})$$

$$\boxed{\text{Carry} \Rightarrow AB + BC_{in} + AC_{in}}$$

Sum → ~~AB~~ $A \oplus B \oplus C_{in}$

Carry → $AB + BC_{in} + AC_{in}$



Poll

2. Total number of inputs in a half adder is ________

a) 2

b) 3

c) 4

d) 1

Solutions

2. Total number of inputs in a half adder is ________

a) 2

b) 3

c) 4

d) 1

View Answer

Answer: a

Explanation: Total number of inputs in a half adder is two. Since an EXOR gates has 2 inputs and carry is connected with the input of EXOR gates. The output of half-adder is also 2, them being, SUM and CARRY. The output of EXOR gives SUM and that of AND gives carry.

Poll

4. If A and B are the inputs of a half adder, the sum is given by ________

a) A AND B

b) A OR B

c) A XOR B

d) A EX-NOR B

4. If A and B are the inputs of a half adder, the sum is given by ________

a) A AND B

b) A OR B

c) A XOR B

d) A EX-NOR B

View Answer

Answer: c

Explanation: If A and B are the inputs of a half adder, the sum is given by A XOR B, while the carry is given by A AND B.

Poll

5. If A and B are the inputs of a half adder, the carry is given by __________

a) A AND B

b) A OR B

c) A XOR B

d) A EX-NOR B

5. If A and B are the inputs of a half adder, the carry is given by _________

a) A AND B

b) A OR B

c) A XOR B

d) A EX-NOR B

^ View Answer

Answer: a

Explanation: If A and B are the inputs of a half adder, the carry is given by: A(AND)B, while the sum is given by A XOR B.

POll

7. The difference between half adder and full adder is ________

a) Half adder has two inputs while full adder has four inputs

b) Half adder has one output while full adder has two outputs

c) Half adder has two inputs while full adder has three inputs

d) All of the Mentioned

Solutions

7. The difference between half adder and full adder is ________

a) Half adder has two inputs while full adder has four inputs

b) Half adder has one output while full adder has two outputs

c) Half adder has two inputs while full adder has three inputs

d) All of the Mentioned

^ View Answer

Answer: c

Explanation: Half adder has two inputs while full adder has three outputs; this is the difference between them, while both have two outputs SUM and CARRY.

Binary Subtractor

1. Half subtractor is a combinational circuit with two inputs and two output.

2. It produces the difference between two binary bits.

3. It subtract (A-B); A is called minuend bit & B is called as subtrahend bit

Inputs		Outputs	
A	B	Diff	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

K-Map Simplification :-

for Difference :-

A \ B	$\bar{B}$ 0	B 1
$\bar{A}$ 0	$_0$	1 $_1$
A 1	1 $_2$	$_3$

$\therefore$ Diff. $= \bar{A}B + A\bar{B}$

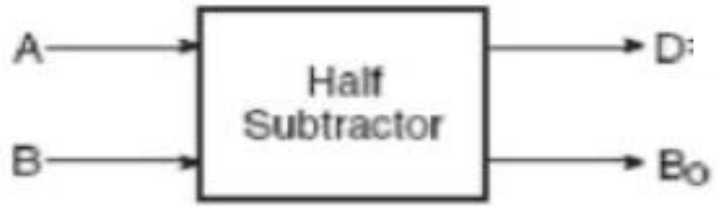
for Borrow :-

A \ B	$\bar{B}$ 0	B 1
$\bar{A}$ 0	$_0$	1 $_1$
A 1	$_2$	$_3$

$\therefore$ Borrow $= \bar{A}B$

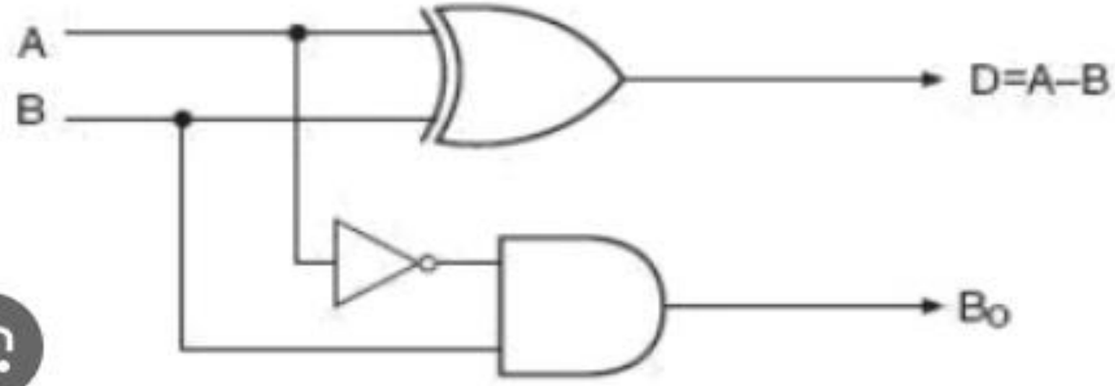
$$D = \overline{A}.B + A.\overline{B}$$

$$B_o = \overline{A}.B$$



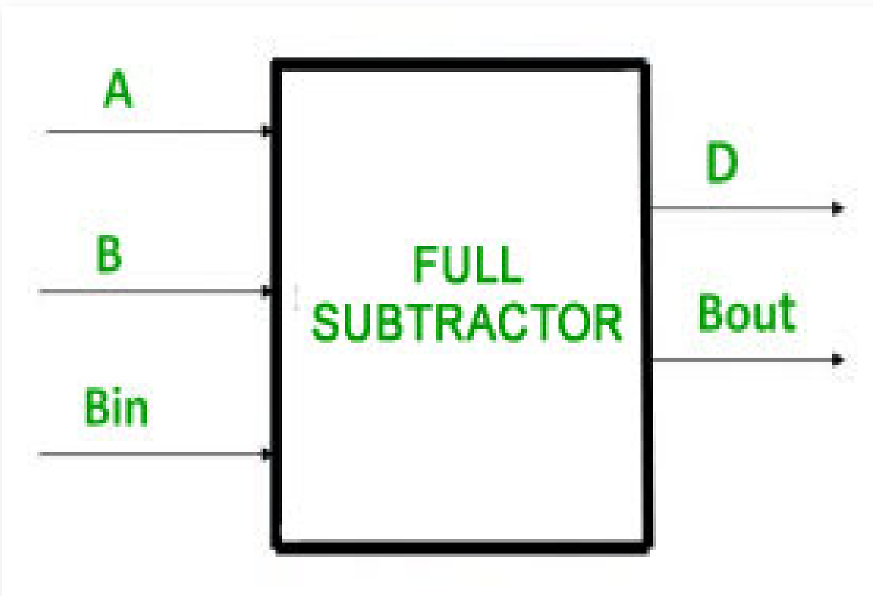
A	B	D	B_o
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Half Subtractor



Full Subtractor

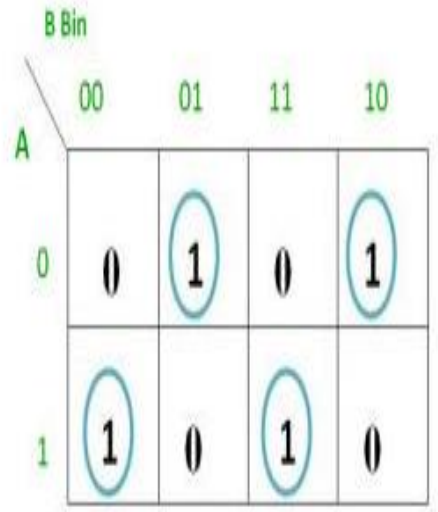
A full subtractor is a **combinational circuit** that performs subtraction of two bits, one is minuend and other is subtrahend, taking into account borrow of the previous adjacent lower minuend bit. This circuit **has three inputs and two outputs**. The three inputs A, B and Bin, denote the minuend, subtrahend, and previous borrow, respectively. The two outputs, D and Bout represent the difference and output borrow, respectively.



Truth Table

From above table we can draw the K-Map as shown for "difference" and "borrow".

INPUT			OUTPUT	
A	B	Bin	D	Bout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1



D=A'B'Bin + AB'Bin' + A'BBin' + ABBin

Logical expression for difference -

```
D   = A'B'Bin + A'BBin' + AB'Bin' + ABBin

    = Bin(A'B' + AB)  + Bin'(AB' + A'B)

    = Bin( A XNOR B) + Bin'(A XOR B)

    = Bin (A XOR B)'  +  Bin'(A XOR B)

    = Bin XOR (A XOR B)

    = (A XOR B) XOR Bin
```

INPUT			OUTPUT	
A	B	Bin	D	Bout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

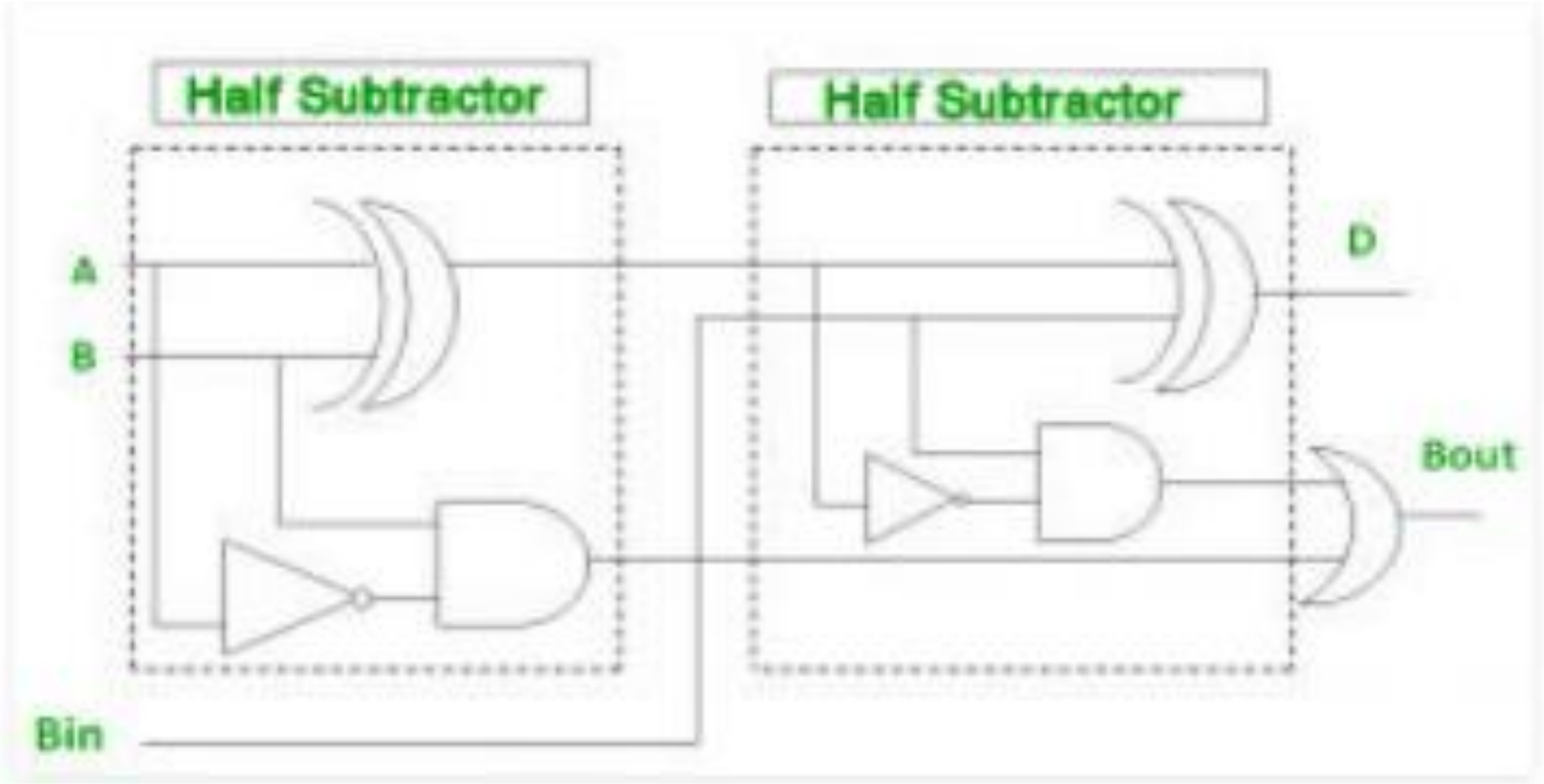
A \ B Bin	00	01	11	10
0	0	1	1	1
1	0	0	1	0

Bout=A’Bin + A’B + BBin

Logical expression for borrow -

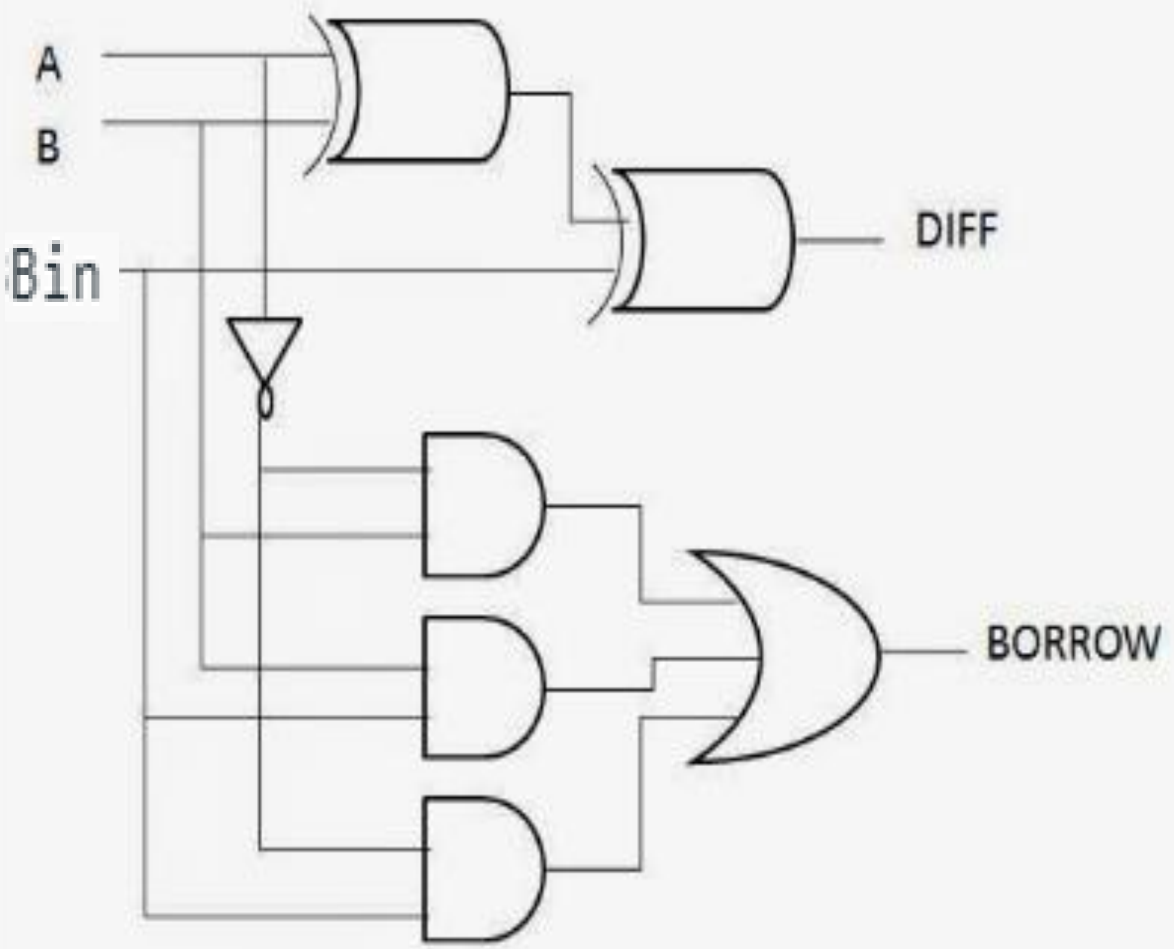
```
Bout = A'B'Bin + A'BBin' + A'BBin + ABBin

     = A'B'Bin +A'BBin' + A'BBin + A'BBin + A'BBin + ABBin

     = A'Bin(B + B') + A'B(Bin + Bin') + BBin(A + A')

     = A'Bin + A'B + BBin


OR


Bout = A'B'Bin + A'BBin' + A'BBin + ABBin

     = Bin(AB + A'B') + A'B(Bin + Bin')

     = Bin( A XNOR B) + A'B

     = Bin (A XOR B)' + A'B
```

Diagram-Full Subtractor using Half Subtractor



Full subtractor using general logic diagram

FULL SUBTRACTOR



A	B	C	D	BO
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Poll

1. Half subtractor is used to perform subtraction of ________

a) 2 bits

b) 3 bits

c) 4 bits

d) 5 bits

1. Half subtractor is used to perform subtraction of __________

a) 2 bits

b) 3 bits

c) 4 bits

d) 5 bits

View Answer

Answer: a

Explanation: Half subtractor is a combinational circuit which is used to perform subtraction of two bits, namely minuend and subtrahend and produces two outputs, borrow and difference.

2. For subtracting 1 from 0, we use to take a _______ from neighbouring bits.

a) Carry

b) Borrow

c) Input

d) Output

2. For subtracting 1 from 0, we use to take a _______ from neighbouring bits.

a) Carry

b) Borrow

c) Input

d) Output

View Answer

Answer: b

Explanation: For subtracting 1 from 0, we use to take a borrow from neighbouring bits because carry is taken into consideration during addition process.

3. How many outputs are required for the implementation of a subtractor?

a) 1

b) 2

c) 3

d) 4

3. How many outputs are required for the implementation of a subtractor?

a) 1

b) 2

c) 3

d) 4

View Answer

Answer: b

Explanation: There are two outputs required for the implementation of a subtractor. One for the difference and another for borrow.

4. Let the input of a subtractor is A and B then what the output will be if A = B?

a) 0

b) 1

c) A

d) B

4. Let the input of a subtractor is A and B then what the output will be if A = B?

a) 0

b) 1

c) A

d) B

View Answer

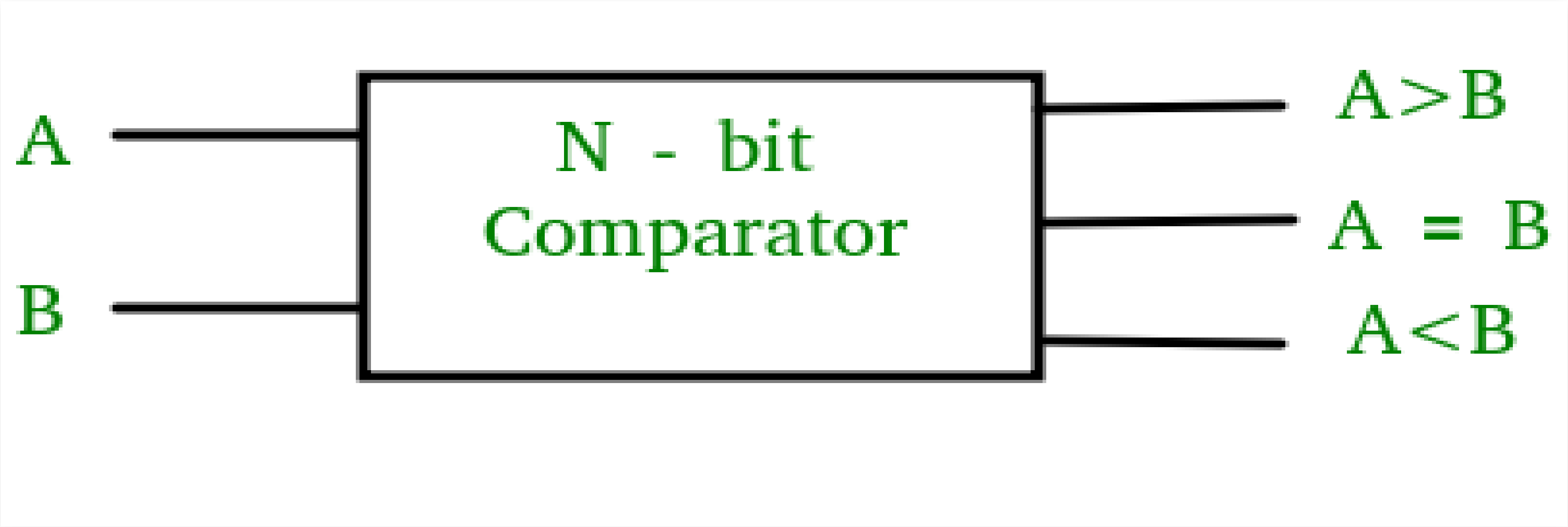
Answer: a

Explanation: The output for A = B will be 0. If A = B, it means that A = B = 0 or A = B = 1. In both of the situation subtractor gives 0 as the output.

Magnitude Comparator in Digital Logic

A magnitude digital Comparator is a combinational circuit that **compares two digital or binary numbers** in order to find out whether one binary number is equal, less than or greater than the other binary number. We logically design a circuit for which we will have two inputs one for A and other for B and have three output terminals, one for A > B condition, one for A = B condition and one for A < B condition.

Diagram



1-Bit Magnitude Comparator –

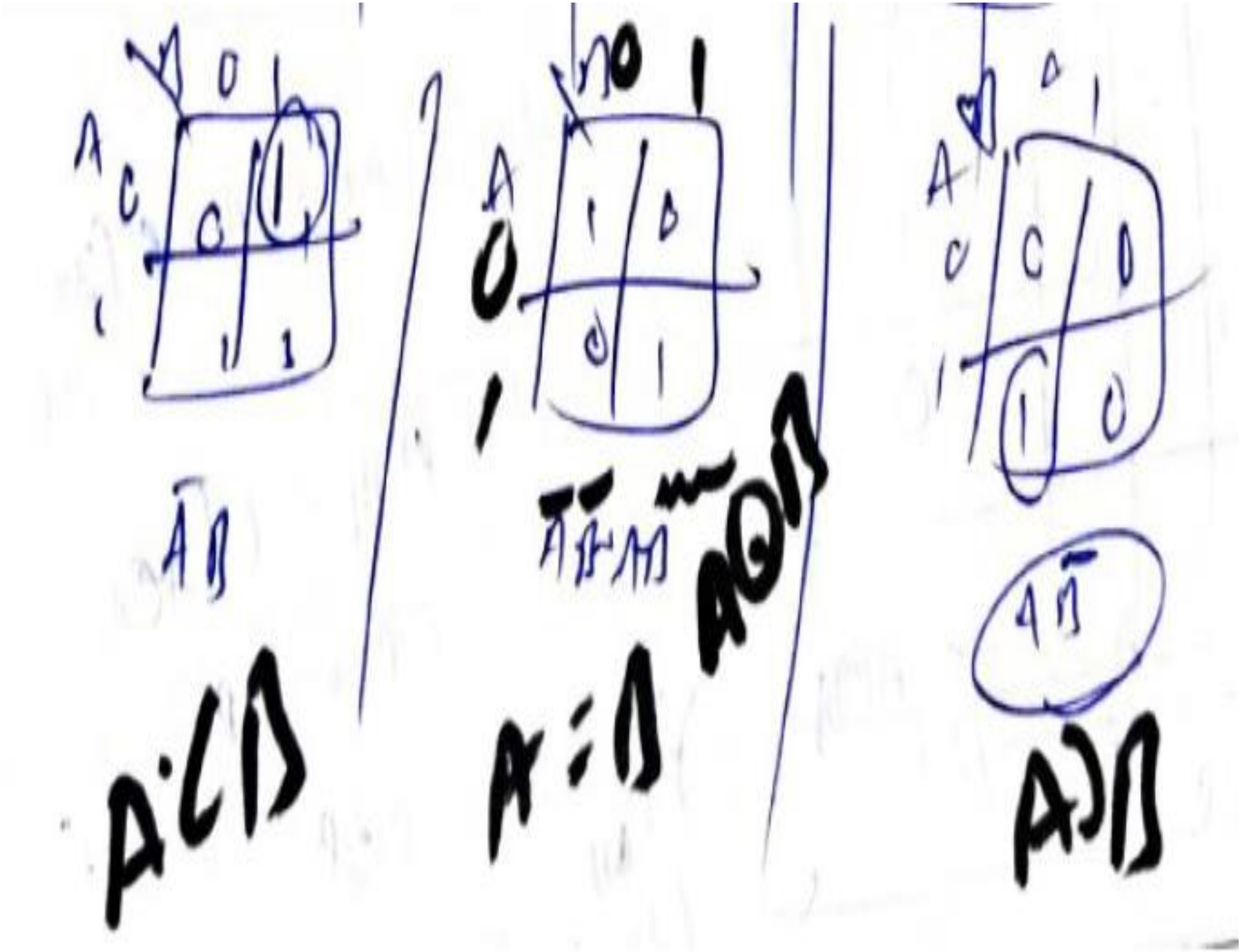
A comparator used to compare two bits is called a single bit comparator. It consists of two inputs each for two single bit numbers and three outputs to generate less than, equal to and greater than between two binary numbers.

The truth table for a 1-bit comparator is given below:

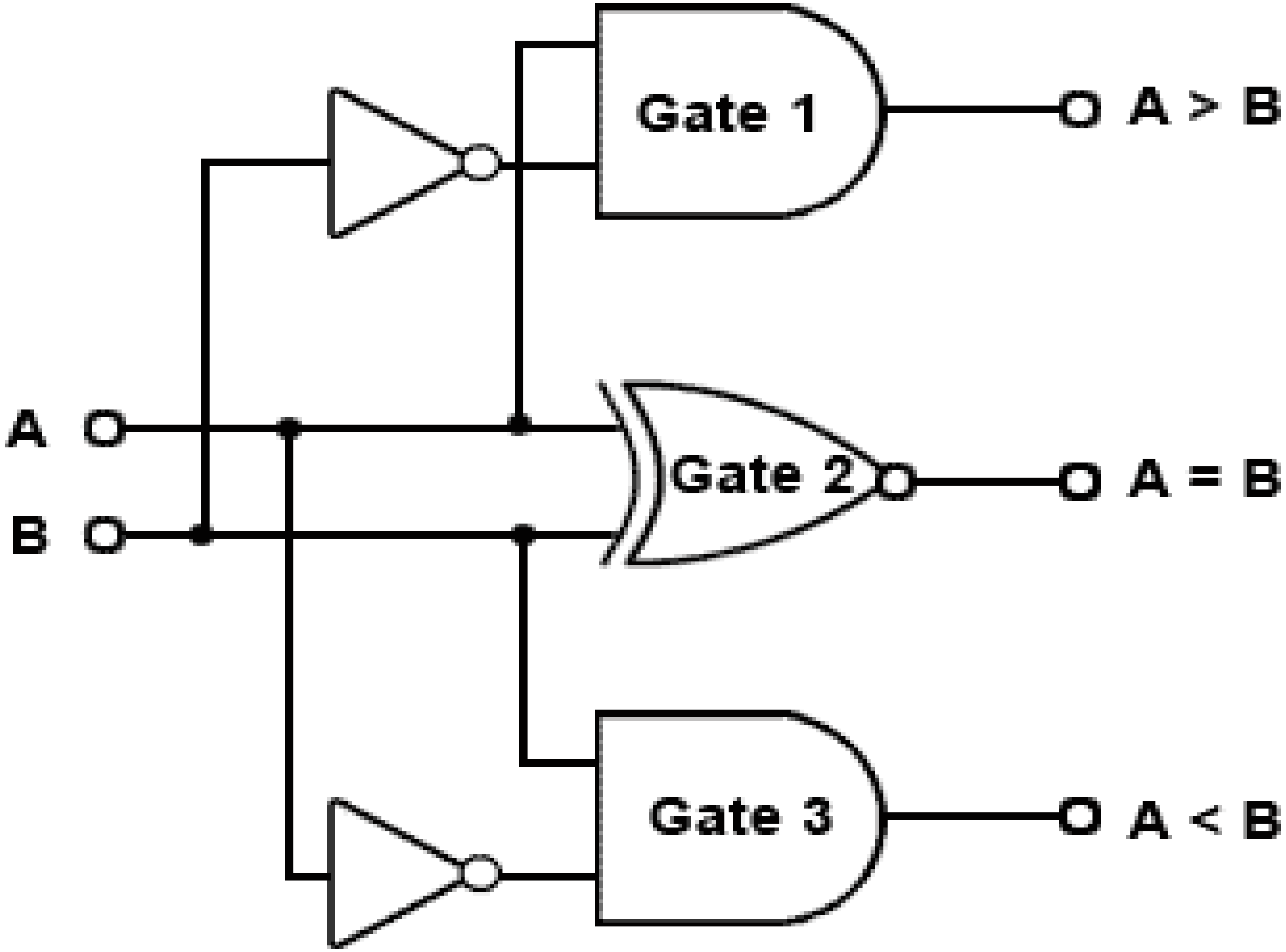
A	B	A<B	A=B	A>B
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0

The truth table for a 1-bit comparator is given below:

A	B	A<B	A=B	A>B
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0



Circuit Diagram



2 bit magnitude comparator

Inputs				Outputs		
A_1	A_0	B_1	B_0	A>B	A=B	A<B
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

Kmap

A>B

A_1A_0 \ B_1B_0	00	01	11	10
00	0	0	0	0
01	1	0	0	0
11	1	1	0	1
10	1	1	0	0

A=B

A_1A_0 \ B_1B_0	00	01	11	10
00	1	0	0	0
01	0	1	0	0
11	0	0	1	0
10	0	0	0	1

From the above k-map simplification, each output can be expressed as

$$A>B: G = A0\,\overline{B1}\;\overline{B0} + A1\,\overline{B1} + A1\;A0\,\overline{B0}$$

$$A=B: E = \overline{A1}\;\overline{A0}\;\overline{B1}\;\overline{B0} + \overline{A1}\;A0\;\overline{B1}\;B0 + A1\;A0\;B1\;B0 + A1\;\overline{A0}\;B1\;\overline{B0}$$

$$= \overline{A1}\;\overline{B1}\;(\overline{A0}\;\overline{B0} + A0\,B0) + A1\;B1\;(A0\,B0 + \overline{A0}\;\overline{B0}\,)$$

$$= (A0\,B0 + \overline{A0}\;\overline{B0})\;(A1\;B1 + \overline{A1}\;\overline{B1})$$

$$= (A0 \text{ Ex-NOR } B0)\;(A1 \text{ Ex-NOR } B1)$$

$$A<B: L = \overline{A1}\;\;B1 + \overline{A0}\;B1\;B0 + \overline{A1}\;\;\overline{A0}\;B0$$

Poll

1. All the comparisons made by comparator is done using ___________

a) 1 circuit

b) 2 circuits

c) 3 circuits

d) 4 circuits

Solutions

1. All the comparisons made by comparator is done using ___________

a) 1 circuit

b) 2 circuits

c) 3 circuits

d) 4 circuits

View Answer

Answer: a

Explanation: A comparator is a combinational circuit that takes two numbers as input in binary form and results whether one input is greater, lesser or equal to the other input. Because, all the input is compared to each other, therefore it is possible only by using 1 circuit.

Poll

2. One that is not the outcome of magnitude comparator is __________

a) a > b

b) a - b

c) a < b

d) a = b

Solutions

2. One that is not the outcome of magnitude comparator is ___________

a) a > b

b) a – b

c) a < b

d) a = b

View Answer

Answer: b

Explanation: A comparator is a combinational circuit that takes two numbers as input in binary form and results whether one input is greater, lesser or equal to the other input. In a digital comparator, only 3 outputs are possible (i.e. A = B, A > B, A < B). So, a – b is an incorrect option.

Poll

5. In a comparator, if we get input as A>B then the output will be ___________

a) 1

b) 0

c) A

d) B

Solutions

5. In a comparator, if we get input as A>B then the output will be ___________

a) 1

b) 0

c) A

d) B

View Answer

Answer: a

Explanation: A comparator is a combinational circuit that takes two numbers as input in binary form and results whether one input is greater, lesser or equal to the other input. If A > B, it means that it satisfies one of the condition among three. Hence the output will be 1.

Parity Generator

It is combinational circuit that accepts an n-1 bit stream data and generates the additional bit that is to be transmitted with the bit stream. This additional or extra bit is termed as a parity bit.

In **even parity** bit scheme, the parity bit is '**0**' if there are **even number of 1s** in the data stream and the parity bit is '**1**' if there are **odd number of 1s** in the data stream.

In **odd parity** bit scheme, the parity bit is '**1**' if there are **even number of 1**s in the data stream and the parity bit is '**0**' if there are **odd number of 1s** in the data stream. Let us discuss both even and odd parity generators.

Even Parity Generator

Let us assume that a 3-bit message is to be transmitted with an even parity bit. Let the three inputs A, B and C are applied to the circuits and output bit is the parity bit P. The total number of 1s must be even, to generate the even parity bit P.

The figure below shows the truth table of even parity generator in which 1 is placed as parity bit in order to make all 1s as even when the number of 1s in the truth table is odd.