REPORT ON ADDER CIRCUIT

An adder is a digital circuit that performs addition of numbers. In many kinds of processors adders are used in the arithmetic logic units They are also used in other parts of the processor, where they are used to calculate addresses, table indices, increment and decrement operators and similar operations. These can be built for many numerical representations like excess-3 or binary coded decimal. Adders are classified into two types***:*** ***half adder***

full adder.

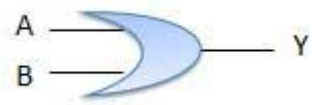
In this project we are designing a full adder by only using IC's. So first we need to know about these IC's (logic gates) and the process of addition.

- ***LOGIC GATE:-*** Logic gates are the basic building blocks of any digital system. It is an electronic circuit having one or more than one input and only one output. The relationship between the input and the output is based on a certain logic.
  In this project we are using ***XOR, AND, OR*** LOGIC GATES.

OR GATE: - A circuit which performs an OR operation is shown in figure. It has n input (n >= 2) and one output.

$$Y = A \text{ OR } B \text{ OR } C \ldots\ldots N$$
$$Y = A + B + C \ldots\ldots N$$

Logic diagram



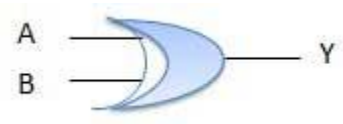
Truth Table

Inputs		Output
A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

- ***XOR GATE: -*** XOR or Ex-OR gate is a special type of gate. It can be used in the half adder, full adder and subtractor. The exclusive-OR gate is abbreviated as EX-OR gate or sometime as X-OR gate.
  It has n input (n >= 2) and one output.

$$Y = A \text{ XOR } B \text{ XOR } C \ldots\ldots N$$
$$Y = A \oplus B \oplus C \ldots\ldots N$$
$$Y = \overline{A}B + A\overline{B}$$

Logic diagram



Truth Table

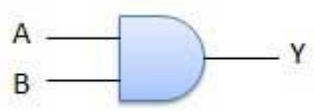
Inputs		Output
A	B	A ⊕ B
0	0	0
0	1	1
1	0	1
1	1	0

-

- ***AND GATE: -*** A circuit which performs an AND operation is shown in figure. It has n input (n >= 2) and one output.

Y	=	A AND B AND C ..... N
Y	=	A.B.C ..... N
Y	=	ABC ..... N

Logic diagram



Truth Table

Inputs		Output
A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

- ***BINARY ADDITION: -*** The binary addition operation works similarly to the base 10 decimal system, except that it is a base 2 system. The binary system consists of only two digits, 1 and 0. The binary code uses the digits 1's and 0's to make certain processes turn off or on.

The four rules of binary addition are:

- 0 + 0 = 0
- 0 + 1 = 1
- 1 + 0 = 1
- 1 + 1 =10

Lets take an example, we have to add (10001)=17 and (11101)=29 . So, sum must be 46

1 0 0 0 1

(+) 1 1 1 0 1

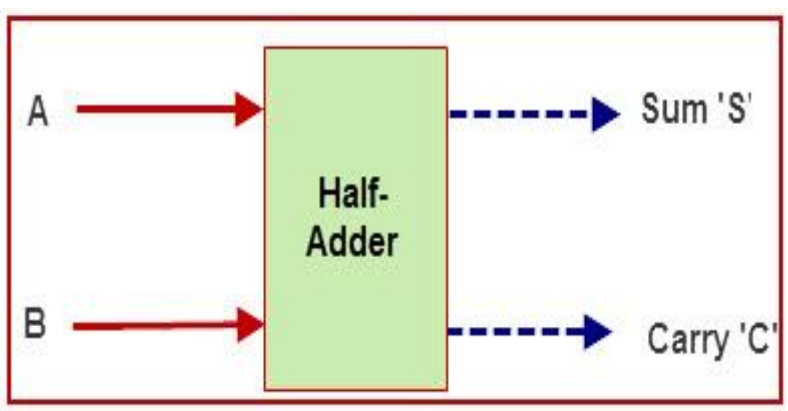
———————

(1 0 1 1 1 0) = 46

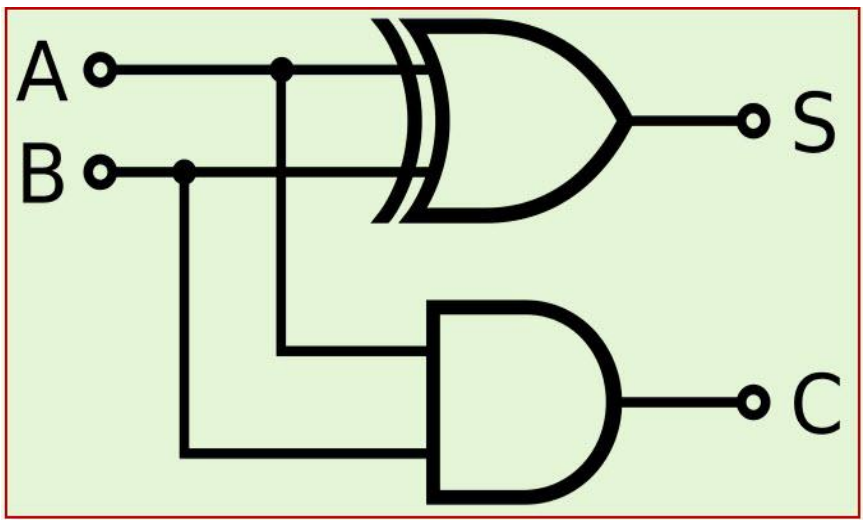
———————

HALF ADDER

An adder is a digital circuit that performs addition of numbers. The half adder adds two binary digits called as augend and addend and produces two outputs as sum and carry



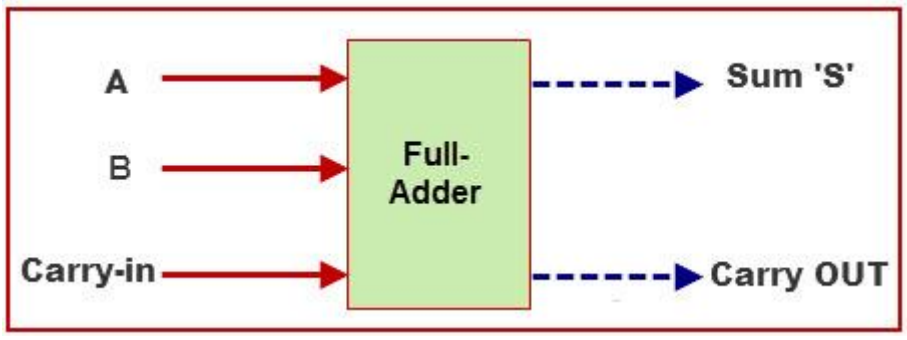
; XOR is applied to both inputs to produce sum and AND gate is applied to both inputs to produce carry.

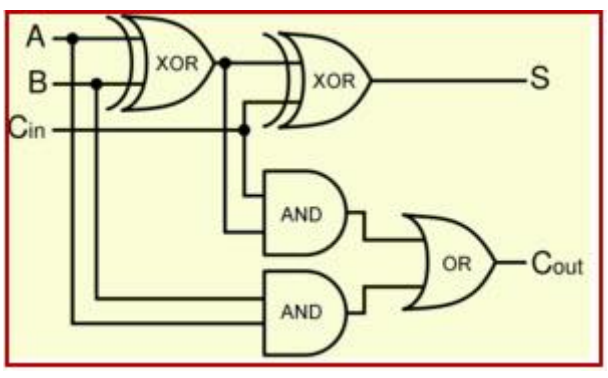


TRUTH TABLE

INPUTS		OUTPUTS	
A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

FULL ADDER: **-** This adder is difficult to implement than a half-adder. The difference between a half-adder and a full-adder is that the full-adder has three inputs and two outputs, whereas half adder has only two inputs and two outputs. The first two inputs are A and B and the third input is an input carry as C-IN. When a full-adder logic is designed, you string eight of them together to create a byte-wide adder and cascade the carry bit from one adder to the next.

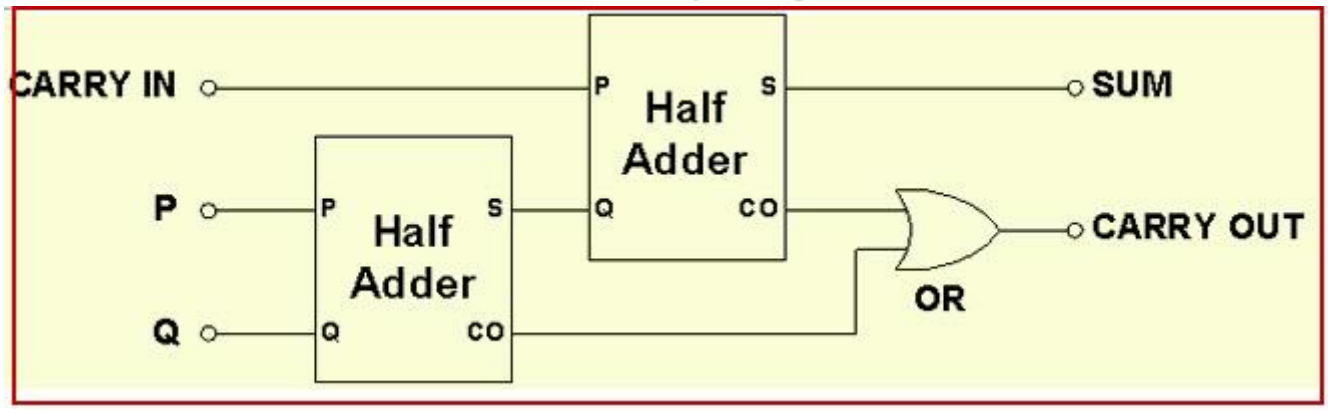




TRUTH TABLE

INPUTS			OUTPUT	
A	B	C-IN	C-OUT	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

NOTE— we can make a full adder by using two half adders as shown



PRACTICAL IMPLEMENTATION USING PROTEUS SOFTWARE

COMPONENTS USED: - XOR *GATE (IC 4070),* AND GATE (IC 4081 or 7408), OR GATE (IC 4071), LOGICPROBE (binary output) and LOGICSTATE (binary input).

In this project I took 2 numbers A= (1111) = 15 and B= (1110) = 14. So, by adding these two numbers we must get 29(in binary form) as a result. So, in the binaryprobe we must get (11101) as output.

$$\begin{array}{r} 1\ 1\ 1\ 1 \\ (+)1\ 1\ 1\ 0 \\ \hline (11101) = 29 \end{array}$$