DAY – 4

HALF ADDER

Theory : -

A half adder is a fundamental digital circuit used in computer hardware and digital electronics to perform basic binary addition. It adds two single-digit binary numbers and produces two outputs: the sum (S) and the carry (C). Half adders are essential building blocks for more complex arithmetic circuits like full adders, which can add multiple bits.

Here's some information about a half adder:

Inputs:

A: The first binary input digit (0 or 1).

B: The second binary input digit (0 or 1).

Outputs:

1. Sum (S): This output represents the least significant bit of the addition result. The sum is obtained by taking the XOR (exclusive OR) of A and B. S = A XOR B

2. Carry (C): This output represents the carry-out of the addition operation. The carry is obtained by taking the AND operation of A and B. C = A AND B

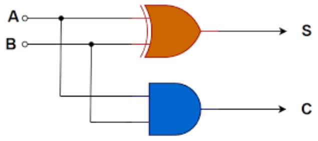
Truth Table: The following truth table summarizes the behavior of a half adder

A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	1	1

Logical Equations: The logical equations for the sum (S) and carry (C) can be expressed as follows:

1. Sum (S) = A XOR B
2. Carry (C) = A AND B

Circuit Diagram



Gate Modelling

```
module HalfAdder(A,B,Sum,Carry);
 input A,B;
 output Sum,Cary;
 xor G1(Sum,A,B);
 and G2 (Carry,A,B);
endmodule
```

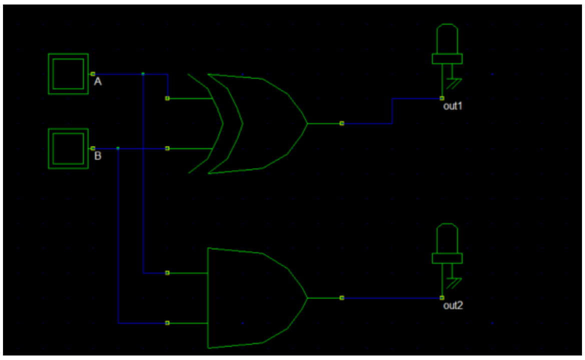
DataFlow Modelling

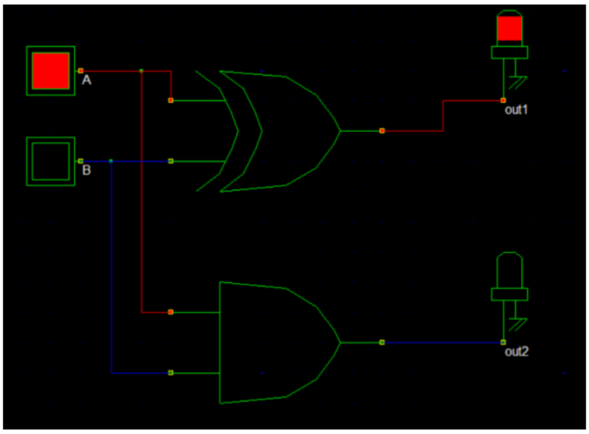
```
 module HalfAdder(A,B,Sum,Carry);
  input A,B;
  output Sum,Carry;
  assign Sum = (A ^ B);
  assign Carry = (A && B);
endmodule
```

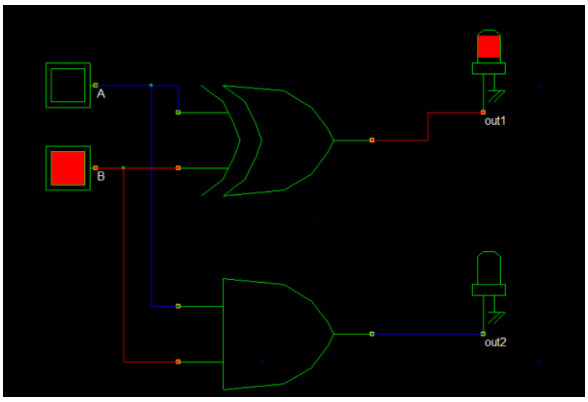
Behavioral Modelling

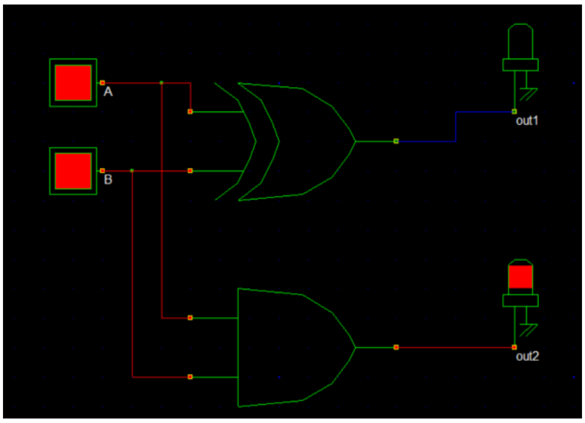
```
module HalfAdder(A,B,Sum,Carry);
 input A,B;
 output Sum,Carry;

always @(A,B)
  begin
  case ({A,B})
  3'b00: sum = 0;
  3'b01: sum = 1;
  3'b10: sum = 1;
  3'b11: sum = 0;
  default : sum = 0;
  endcase

  case ({A,B})
  3'b00: cout = 0;
  3'b01: cout = 0;
  3'b10: cout = 0;
  3'b11: cout = 1;
  default : cout = 0;
  endcase
  end
endmodule
```

Dsch – Using Xor and AND

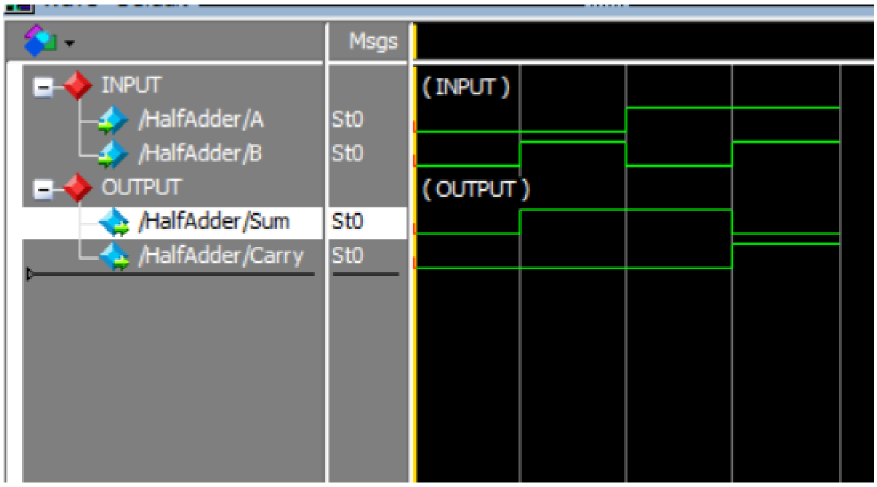








Simulation waveform:


Msgs
INPUT
/HalfAdder/A
St0
/HalfAdder/B
St0
OUTPUT
/HalfAdder/Sum
St0
/HalfAdder/Carry
St0
(INPUT)
(OUTPUT)