EXPT.-05

AIM: - Design and Simulate 2-bit comparator Verilog HDL.

Software: - Xilinx Vivado 2019.1

Theory: -

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.

Figure-1: Block Diagram of Comparator

1) 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

Figure-2: Truth Table of 1-Bit Comparator

From the above truth table logical expressions for each output can be expressed as follows:

A > B : AB'

A < B : A'B

A = B : A'B' + AB

By using these Boolean expressions, we can implement a logic circuit for this comparator as given below :

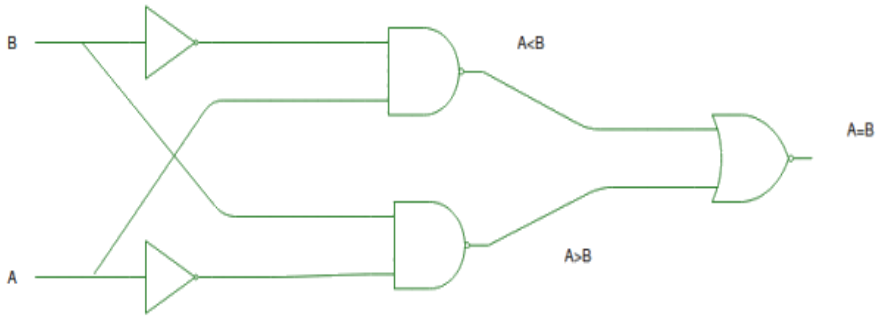


Figure-3: Logic Circuit of 1-Bit Comparator

2) 2-Bit Magnitude Comparator :

A comparator used to compare two binary numbers each of two bits is called a 2-bit magnitude comparator. It consists of four inputs and three outputs to generate less than, equal to and greater than between two binary numbers.

The truth table for a 2-bit comparator is given below:

INPUT				OUTPUT		
A1	A0	B1	B0	A<B	A=B	A>B
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	0

Figure-4: Truth Table of 2-Bit Comparator

From the above truth table logical expressions for each output can be expressed as follows:

$A > B : A_1B_1' + A_0B_1'B_0' + A_1A_0B_0'$

$A = B : A_1'A_0'B_1'B_0' + A_1'A_0B_1'B_0 + A_1A_0B_1B_0 + A_1A_0'B_1B_0'$

$: A_1'B_1' (A_0'B_0' + A_0B_0) + A_1B_1 (A_0B_0 + A_0'B_0')$

$: (A_0B_0 + A_0'B_0') (A_1B_1 + A_1'B_1')$

$: (A_0 \text{ Ex-Nor } B_0) (A_1 \text{ Ex-Nor } B_1)$

$A < B : A_1'B_1 + A_0'B_1B_0 + A_1'A0'B0$

By using these Boolean expressions, we can implement a logic circuit for this comparator as given below :

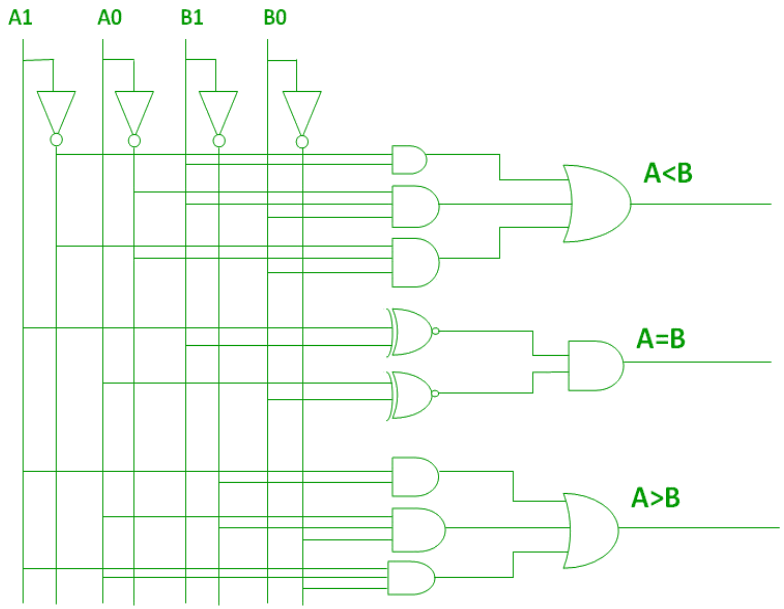


Figure-5: Logic Circuit of 2-Bit Comparator

Applications of Comparators :

1. Comparators are used in central processing units (CPUs) and microcontrollers (MCUs).
2. These are used in control applications in which the binary numbers representing physical variables such as temperature, position, etc. are compared with a reference value.
3. Comparators are also used as process controllers and for Servo motor control.
4. Used in password verification and biometric applications.

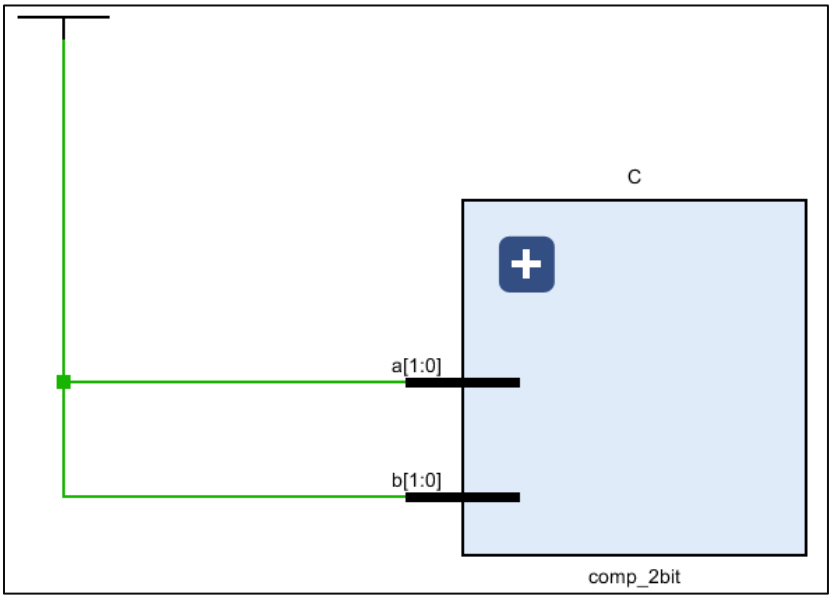
Code: -

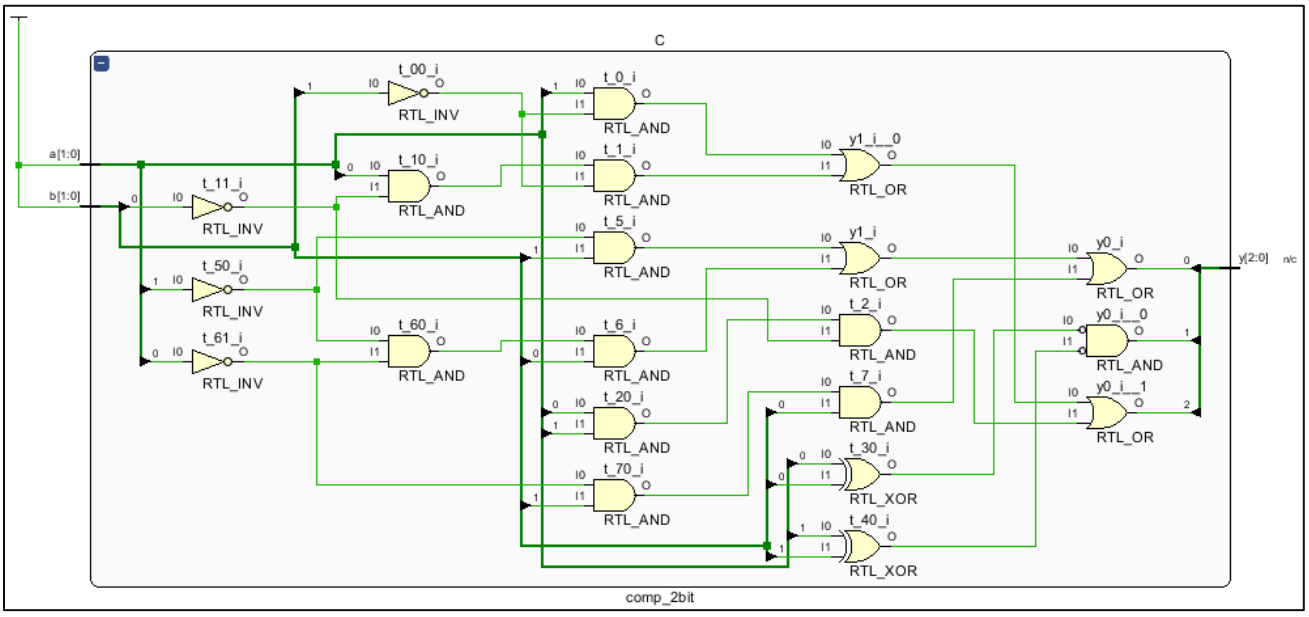
```
//2-bit comparator
module comp_2bit(y,a,b);

//two 2-bit inputs
input [1:0] a,b;

//y[2]-a>b
//y[1]-a=b
//y[0]-a<b
output [2:0] y;

//interconnects
wire [7:0] t;

//a>b
and(t[0],a[1],~b[1]);
and(t[1],a[0],~b[0],~b[1]);
and(t[2],a[0],a[1],~b[0]);
or(y[2],t[0],t[1],t[2]);
//a=b
xnor(t[3],a[0],b[0]);
xnor(t[4],a[1],b[1]);
and(y[1],t[3],t[4]);

//a<b
and(t[5],~a[1],b[1]);
and(t[6],~a[1],~a[0],b[0]);
and(t[7],~a[0],b[1],b[0]);
or(y[0],t[5],t[6],t[7]);

endmodule
```

```
//tb
`timescale 1ns / 1ps

module tb_comp_2bit();
reg [1:0] in1,in2;
wire [2:0] out;

comp_2bit C(out,in1,in2);

integer i;
initial
begin
for(i=0;i<16;i=i+1)
begin
{in1,in2}=i;
#10;
end
end

initial
$monitor($time,"out=%b,in1=%b,in2=%b",out,in1,in2);

initial
#160 $finish;

endmodule
```

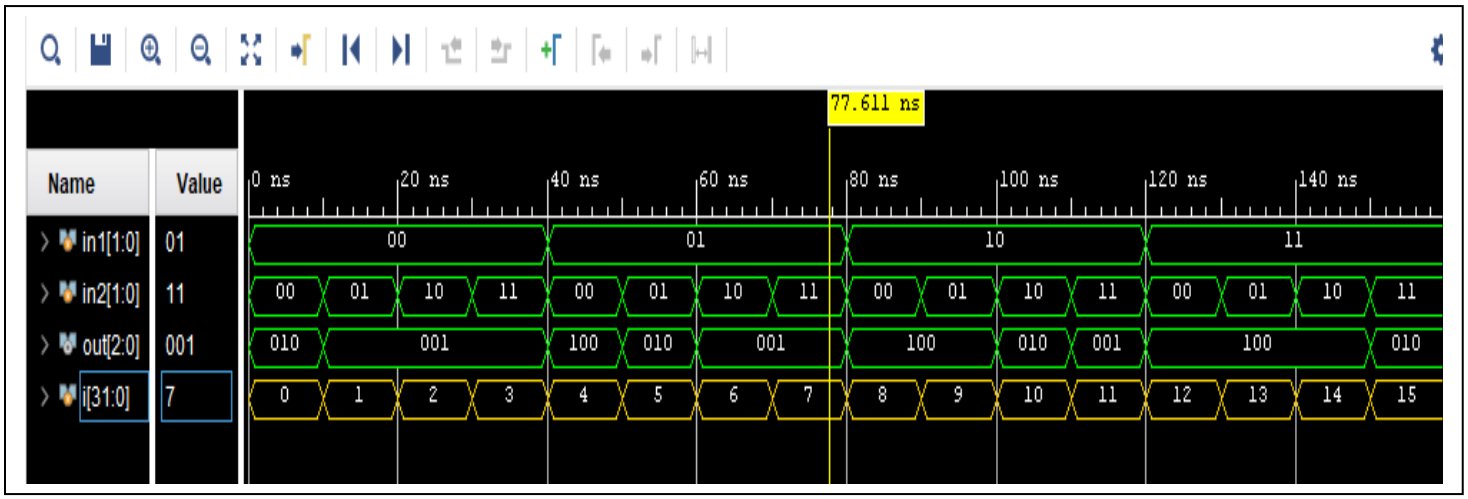
Schematics and Simulated Waveforms: -

Schematic:





Waveform:



Conclusion: -

Hence , we have successfully designed and simulated 2-bit comparator using gate-level modeling style in Verilog HDL. Also verified simulated waveforms with actual truth tables using Xilinx Vivado 2019.1