

RTL CHALLENGE

DAY-16:- Verilog Code for Implementation of Basic Gates Using CONDITIONAL / BEHAVIORAL MODELLING.

Software used:- Xilinx Vivado(2023.1)

Theory:

Behavioral / Conditional:

In Verilog, conditional statements are constructs used to control the flow of logic based on specified conditions or variable values. The if-else statement is commonly used for simple binary decision-making, while case statements provide multi-way branching based on specific values of variables. These conditional constructs enable designers to create flexible and responsive behavior within their hardware description language (HDL) designs.

Here's a basic syntax example:

```
if (condition)
    // statements to execute if condition is true
else
    // statements to execute if condition is false
```

Code for Basicgates Using Behavioral/Conditional statements:

```
module basicgatesusingconditionals(
    input a,b,
    output reg andout,orout,norout,nandout,xorout,xnorout
);
always@(a,b)
begin
    if(a==0&&b==0)
        begin
            andout=0;orout=0;norout=1;nandout=1;xorout=0;xnorout=1;
        end
    else if(a==0&&b==1)
        begin
            andout=0;orout=1;norout=0;nandout=1;xorout=1;xnorout=0;
        end
    else if(a==1&&b==0)
        begin
            andout=0;orout=1;norout=0;nandout=1;xorout=1;xnorout=0;
        end
    else
        begin
            andout=1;orout=1;norout=0;nandout=0;xorout=0;xnorout=1;
        end
    end
endmodule
```

TESTBENCH:-

```
module basicgatesusingconditionals_tb();
reg a,b;
wire andout,orout,norout,nandout,xorout,xnorout;

basicgatesusingconditionals uut (
    .a(a),.b(b),.andout(andout),.orout(orout),.norout(norout),.nandout(nandout),.xorout(xorout),.xnorout(xnorout)
);
initial
begin
    $monitor("when a=%d, b=%d, andout=%b, realtime", a, b, andout, $realtime);
    //The $monitor function in Verilog is used to display variable values in the
    // simulation output, providing real-time feedback on changes.
    a=0;b=0:#100;
    a=0;b=1:#100;
    a=1;b=0:#100;
    a=1;b=1:#100;
    $display("when a=%d, b=%d, andout=%b, realtime", a, b, xorout, $realtime);
    //The $display function in Verilog is used to display formatted text and variable
    //values in the simulation output, aiding in debugging and verification processes.
    $finish;
    //The $finish system task in Verilog is used to terminate the simulation when it is
    //encountered during runtime, effectively stopping the simulation process.
end
endmodule
```

OUTPUT:

when a=0, b=0, andout=0, realtime0

when a=0, b=1, andout=0, realtime100

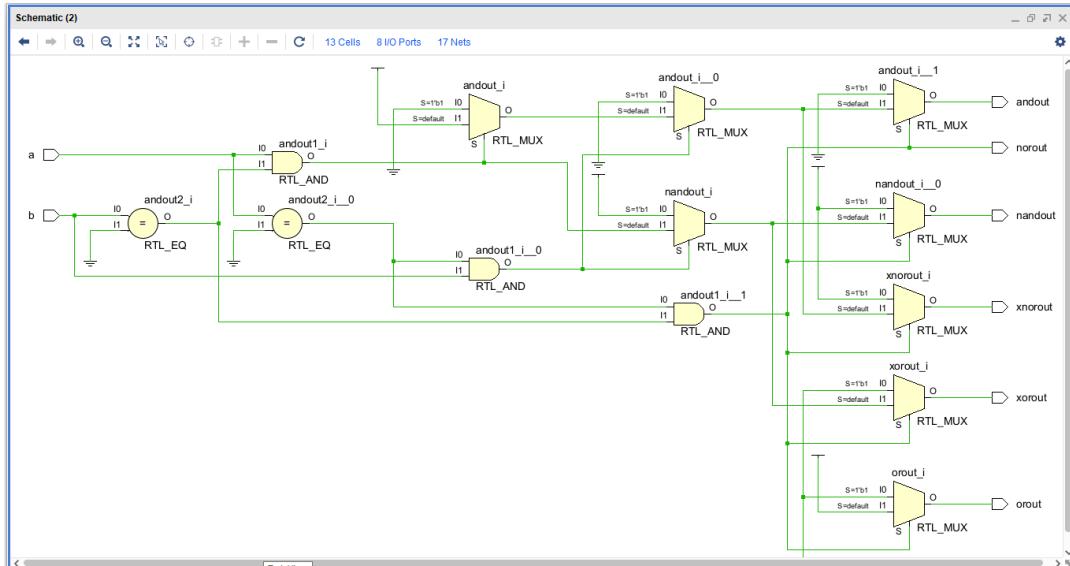
when a=1, b=0, andout=0, realtime200

when a=1, b=1, andout=1, realtime300

when a=1, b=1, andout=1, realtime400

\$finish called at time : 400 ns :

RTL SCHEMATIC:-



OUTPUT WAVEFORM:

