

Code for AND gate

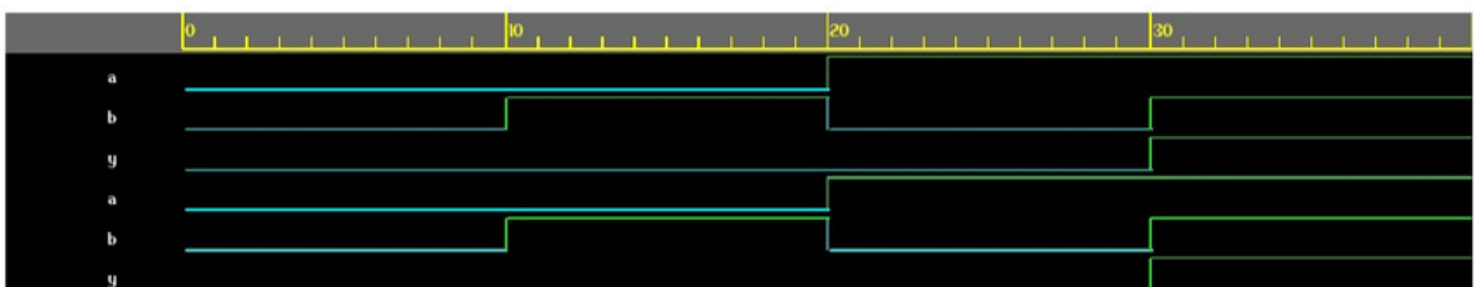
```
3 // Code your design here
4 module and_gate_tb;
5     reg a, b;
6     wire y;
7
8     and_gate uut (.a(a),.b(b),.y(y));
9
10    initial begin
11        $dumpfile("and_gate.vcd");
12        $dumpvars(0, and_gate_tb);
13
14        $display("A B | Y");
15        $display("-----");
16
17        a = 0; b = 0; #10; $display("%b %b
18 | %b", a, b, y);
19        a = 0; b = 1; #10; $display("%b %b
20 | %b", a, b, y);
21        a = 1; b = 0; #10; $display("%b %b
22 | %b", a, b, y);
23        a = 1; b = 1; #10; $display("%b %b
24 | %b", a, b, y);
25
26        $finish;
27    end
28 endmodule
```

```
1 // Code your design here
2 module and_gate (
3     input a,
4     input b,
5     output y
6 );
7     assign y = a & b;
8 endmodule
```

Output:

```
A B | Y
-----
0 0 | 0
0 1 | 0
1 0 | 0
1 1 | 1
```

Waveform:



Code for OR gate

```
1 // Code your testbench here
2 // or browse Examples
3
4 module tb_or_gate;
5     reg a, b;
6     wire y;
7
8     or_gate uut (.a(a),.b(b),.y(y) );
9
10    initial begin
11        $dumpfile("or_gate.vcd");
12        $dumpvars(0, tb_or_gate);
13
14        $display("A B | Y");
15        $display("-----");
16
17        a = 0; b = 0; #10; $display("%b %b
18 | %b", a, b, y);
19        a = 0; b = 1; #10; $display("%b %b
20 | %b", a, b, y);
21        a = 1; b = 0; #10; $display("%b %b
22 | %b", a, b, y);
23        a = 1; b = 1; #10; $display("%b %b
24 | %b", a, b, y);
25
26        $finish;
27    end
28 endmodule
```

```
1 // Code your design here
2 module or_gate (
3     input a,
4     input b,
5     output y
6 );
7     assign y = a|b;
8 endmodule
9
10
```

Output:

A B | Y

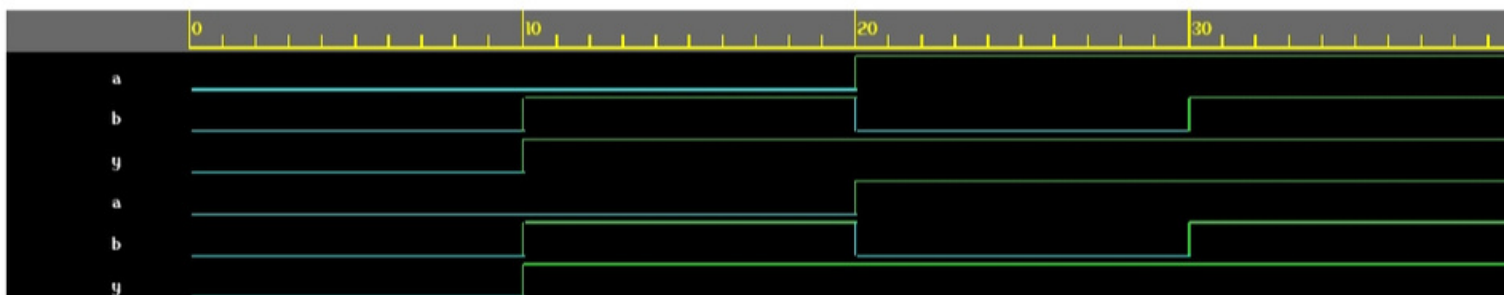
0 0 | 0

0 1 | 1

1 0 | 1

1 1 | 1

Waveform:



Code for NOT gate

```
1 // Code your testbench here
2 // or browse Examples
3
4 module not_gate_tb;
5     reg a;
6     wire y;
7
8     not_gate uut (.a(a),.y(y));
9
10    initial begin
11        $dumpfile("not_gate.vcd");
12        $dumpvars(0, not_gate_tb);
13
14        $display("A | Y");
15        $display("-----");
16
17        a = 0; #10; $display("%b | %b", a,
y);
18        a = 1; #10; $display("%b | %b", a,
y);
19
20        $finish;
21    end
22 endmodule
```

```
1 // Code your design here
2 module not_gate(
3     input a,
4     output y
5 );
6     assign y= ~a;
7 endmodule
```

Output:

A | Y

0 | 1

1 | 0

Waveform:



Code for NAND gate

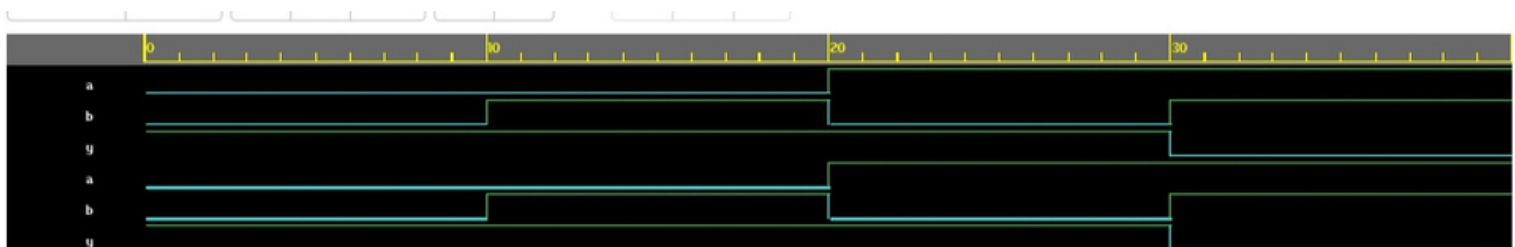
```
2 // or browse Examples
3 module nand_gate_tb;
4 reg a,b;
5 wire y;
6 nand_gate uut (.a(a),.b(b),.y(y));
7
8 initial begin
9     $dumpfile("nand_gate.vcd");
10    $dumpvars(0,nand_gate_tb);
11
12    $display(" A B | Y");
13    $display("-----");
14
15    a=0;b=0;#10; $display("%b %b |
16    %b",a,b,y);
17    a=0;b=1;#10; $display("%b %b |
18    %b",a,b,y);
19    a=1;b=0;#10; $display("%b %b |
20    %b",a,b,y);
21    a=1;b=1;#10; $display("%b %b |
22    %b",a,b,y);
23
24    $finish;
25 end
26 endmodule
```

```
1 // Code your design here
2 module nand_gate(
3 input a,
4 input b,
5 output y
6 );
7
8 assign y= ~(a&b);
9 endmodule
10
```

Output:

```
A B | Y
-----
0 0 | 1
0 1 | 1
1 0 | 1
1 1 | 0
```

Waveform :



Code for NOR gate

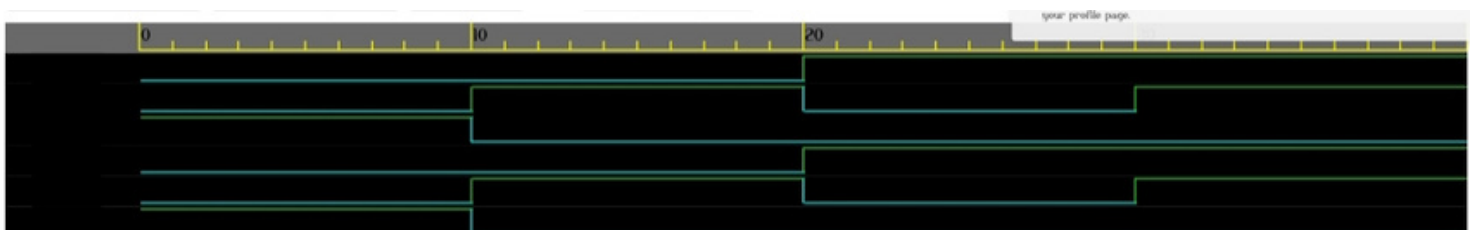
```
3 module nor_gate_tb;
4 reg a,b;
5 wire y;
6 nor_gate uut(.a(a),.b(b),.y(y));
7
8 initial begin
9     $dumpfile("nor_gate.vcd");
10    $dumpvars(0,nor_gate_tb);
11
12    $display("A B | Y");
13    $display("-----");
14
15    a = 0; b = 0; #10; $display("%b %b |
16    %b", a, b, y);
17    a = 0; b = 1; #10; $display("%b %b |
18    %b", a, b, y);
19    a = 1; b = 0; #10; $display("%b %b |
20    %b", a, b, y);
21    a = 1; b = 1; #10; $display("%b %b |
22    %b", a, b, y);
23
24    $finish;
25 end
26 endmodule
```

```
1 // Code your design here
2 module nor_gate (
3     input a,b,
4     output y
5 );
6
7     assign y=~(a|b);
8 endmodule
9
```

Output:

```
A B | Y
-----
0 0 | 1
0 1 | 0
1 0 | 0
1 1 | 0
```

Waveform:



Code for XOR gate

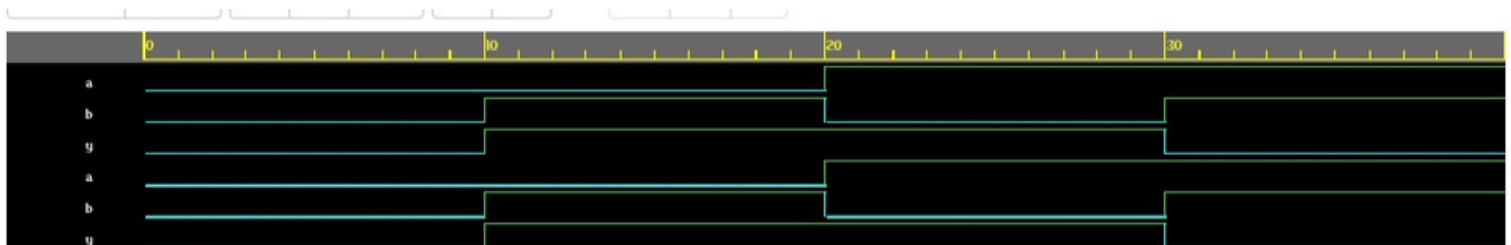
```
3
4 module tb_xor_gate;
5     reg a, b;
6     wire y;
7
8     xor_gate uut (.a(a),.b(b),.y(y) );
9
10    initial begin
11        $dumpfile("xor_gate.vcd");
12        $dumpvars(0, tb_xor_gate);
13
14        $display("A B | Y");
15        $display("-----");
16
17        a = 0; b = 0; #10; $display("%b %b
18 | %b", a, b, y);
19        a = 0; b = 1; #10; $display("%b %b
20 | %b", a, b, y);
21        a = 1; b = 0; #10; $display("%b %b
22 | %b", a, b, y);
23        a = 1; b = 1; #10; $display("%b %b
24 | %b", a, b, y);
25
26        $finish;
27    end
28 endmodule
```

```
1 // Code your design here
2 module xor_gate (
3     input a,
4     input b,
5     output y
6 );
7     assign y = a^b;
8 endmodule
9
10
```

Output:

```
A B | Y
-----
0 0 | 0
0 1 | 1
1 0 | 1
1 1 | 0
```

Waveform:



Code for XOR gate

```
3
4 module tb_xnor_gate;
5     reg a, b;
6     wire y;
7
8     xnor_gate uut (.a(a),.b(b),.y(y) );
9
10    initial begin
11        $dumpfile("xnor_gate.vcd");
12        $dumpvars(0, tb_xnor_gate);
13
14        $display("A B | Y");
15        $display("-----");
16
17        a = 0; b = 0; #10; $display("%b %b
| %b", a, b, y);
18        a = 0; b = 1; #10; $display("%b %b
| %b", a, b, y);
19        a = 1; b = 0; #10; $display("%b %b
| %b", a, b, y);
20        a = 1; b = 1; #10; $display("%b %b
| %b", a, b, y);
21
22        $finish;
23    end
24 endmodule
```

```
1 // Code your design here
2 module xnor_gate (
3     input a,
4     input b,
5     output y
6 );
7     assign y = ~(a^b);
8 endmodule
9
10
```

Output:

A B | Y

0 0 | 1

0 1 | 0

1 0 | 0

1 1 | 1

Waveform:

