COA - Lab 2

Created	@Jan 21, 2021 5:43 PM
Tags	

Basic Gates:

- AND
- OR
- NOT
- NOR
- NAND
- XOR
- XNOR

AND gate:

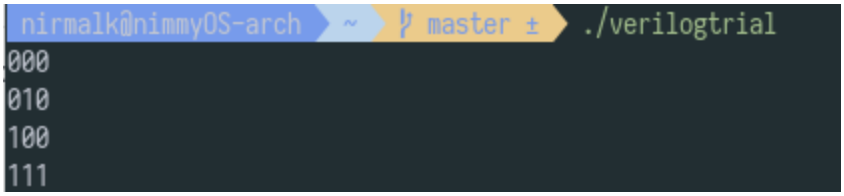
Module:

```
module anddemo(x,y,z);
input x,y;
output z;
assign z = x&y;
endmodule
```

Testbench:

```
module andgate_demo_tb;
reg t_a, t_b;
wire t_y;
```

```
anddemo my_gate(t_a, t_b, t_y);



initial
begin
    $monitor(t_a,t_b,t_y);
    t_a=0;t_b=0;#10;
    t_a=0;t_b=1;#10;
    t_a=1;t_b=0;#10;
    t_a=1;t_b=1;#10;
    $finish;
end
endmodule
```



OR Gate:

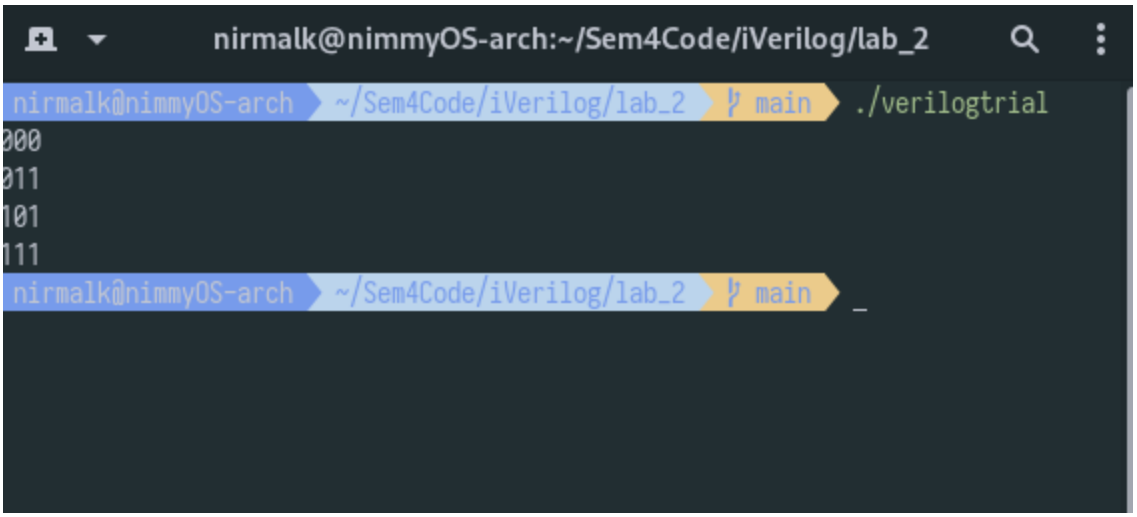
Module:

```
module orgate(a,b,y);
input a,b;
output y;
assign y = a | b;
endmodule
```

Testbench:

```
module orgate_test;
wire t_y;
reg t_a,t_b;

orgate test_gate(t_a,t_b,t_y);

initial
begin
```

```
$monitor(t_a,t_b,t_y);
t_a = 1'b0;
t_b = 1'b0;

#5

t_a = 1'b0;
t_b = 1'b1;

#5


t_a = 1'b1;
t_b = 1'b0;

#5


t_a = 1'b1;
t_b = 1'b1;

end
endmodule
```



NOT Gate:

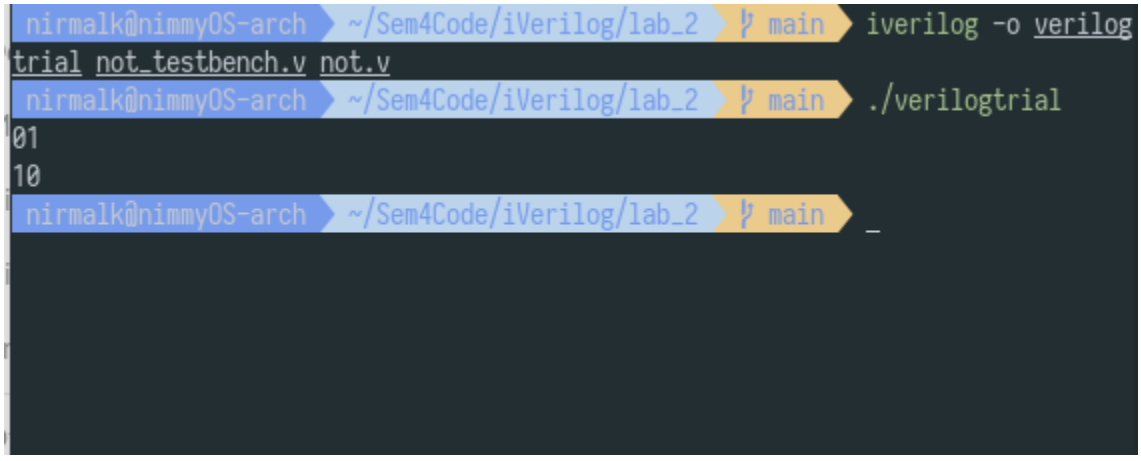
Module:

```
module notgate(a,y);
input a;
```

```
output y;
assign y = !a;
endmodule
```

Testbench:

```
module notgate_test;
wire t_y;
reg t_a;

notgate test_gate(t_a,t_y);

initial
begin
$monitor(t_a,t_y);
t_a = 1'b0;

#5

t_a = 1'b1;

end
endmodule
```



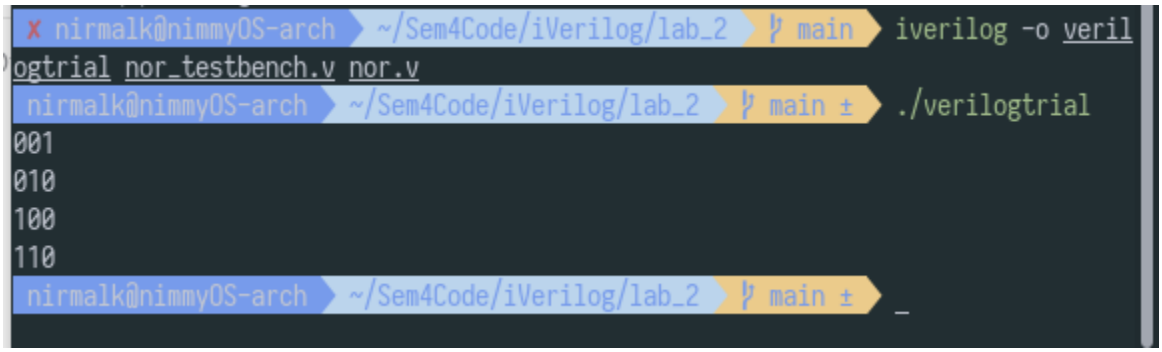
NOR Gate

Module:

```
module norgate(a,b,y);
input a,b;
output y;
assign y = !(a | b);
endmodule
```

Testbench:

```
module norgate_test;
wire t_y;
reg t_a,t_b;


norgate test_gate(t_a,t_b,t_y);

initial
begin

    t_y = 0;
$monitor(t_a,t_b,t_y);
t_a = 1'b0;
t_b = 1'b0;

#5

t_a = 1'b0;
t_b = 1'b1;

#5


t_a = 1'b1;
t_b = 1'b0;

#5


t_a = 1'b1;
t_b = 1'b1;

end
endmodule
```



NAND gate:

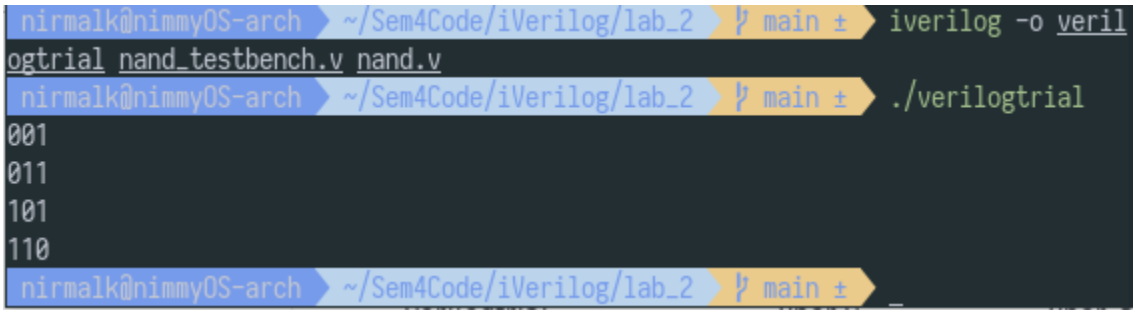
Module:

```
module nandgate(a,b,y);
input a,b;
output y;
assign y = !(a & b);
endmodule
```

Testbench:

```
module nandgate_test;
wire t_y;
reg t_a,t_b;

nandgate test_gate(t_a,t_b,t_y);

initial
begin
$monitor(t_a,t_b,t_y);
t_a = 1'b0;
t_b = 1'b0;

#5

t_a = 1'b0;
t_b = 1'b1;

#5


t_a = 1'b1;
t_b = 1'b0;
```

```
#5


t_a = 1'b1;
t_b = 1'b1;

end
endmodule
```



XOR Gate:

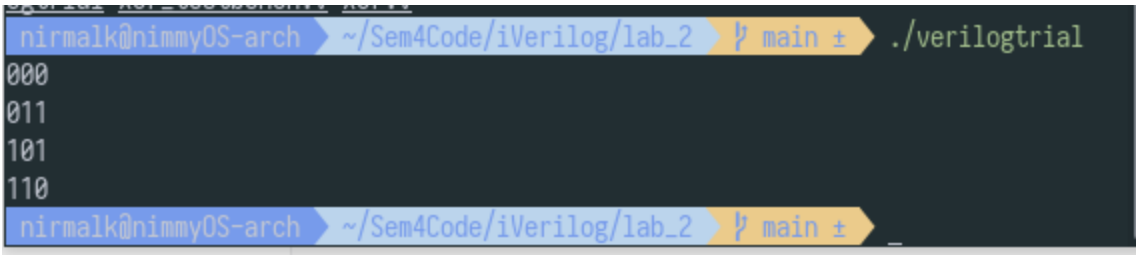
Module:

```
module xorgate(a,b,y);
input a,b;
output y;
assign y = ((a&!b) | (b&!a));
endmodule
```

Testbench:

```
module xorgate_test;
wire t_y;
reg t_a,t_b;

xorgate test_gate(t_a,t_b,t_y);

initial
begin
$monitor(t_a,t_b,t_y);
t_a = 1'b0;
```

```
t_b = 1'b0;

#5

t_a = 1'b0;
t_b = 1'b1;

#5


t_a = 1'b1;
t_b = 1'b0;

#5


t_a = 1'b1;
t_b = 1'b1;

end
endmodule
```



XNOR Gate:

Module:

```
module xnorgate(a,b,y);
input a,b;
output y;
assign y = !((a&!b) | (b&!a));
endmodule
```

Testbench:

```
module xnorgate_test;
wire t_y;
reg t_a,t_b;

xnorgate test_gate(t_a,t_b,t_y);

initial
begin
$monitor(t_a,t_b,t_y);
t_a = 1'b0;
t_b = 1'b0;

#5

t_a = 1'b0;
t_b = 1'b1;

#5


t_a = 1'b1;
t_b = 1'b0;

#5


t_a = 1'b1;
t_b = 1'b1;

end
endmodule
```