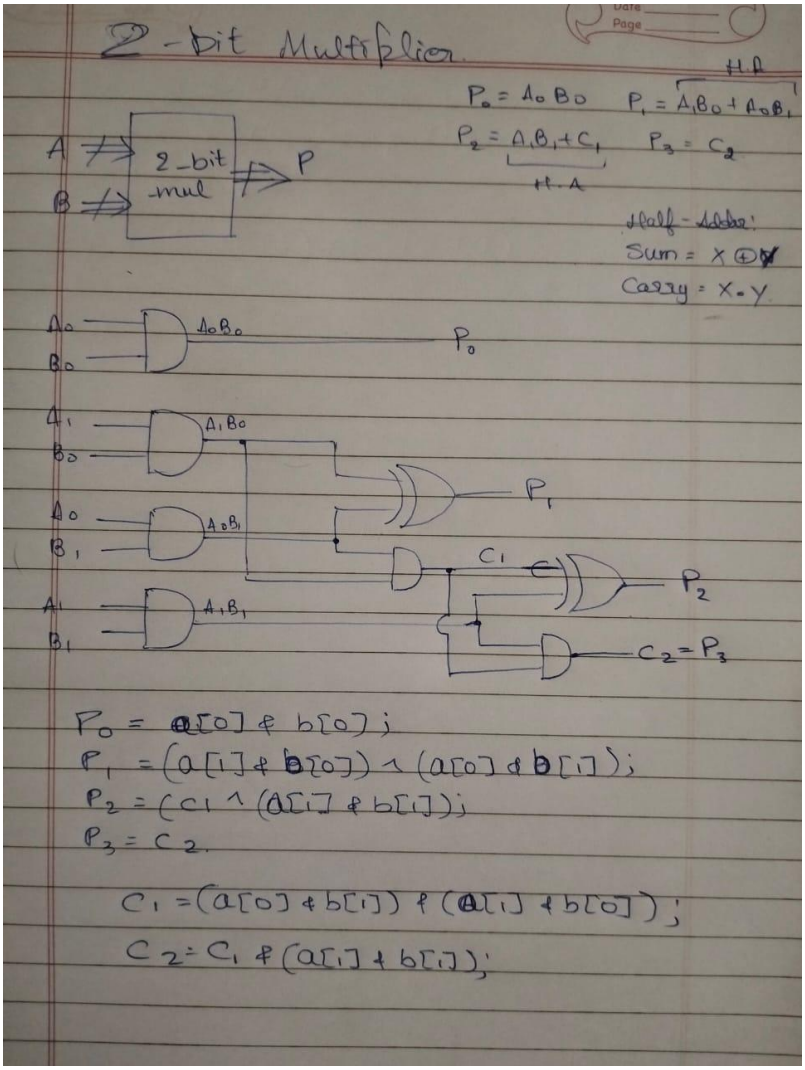
2 Bit Multiplier




```
# run 1000ns
Time    | a    b   => p
   0ns  | 00   00  => 0000
  10ns  | 01   01  => 0001
  20ns  | 10   01  => 0010
  30ns  | 11   01  => 0011
  40ns  | 01   10  => 0010
  50ns  | 10   10  => 0100
  60ns  | 11   10  => 0110
  70ns  | 11   11  => 1001
$finish called at time : 80 ns
```

//Verilog code for 2-bit multiplier

```
module multiplier2Bit(input [1:0] a, b, output [3:0] p );

 wire c1, c2;

   assign c1 = (a[1] & b[0]) & (a[0] & b[1]);

   assign c2 = c1 & (a[1] & b[1]);

   assign p[0] = a[0] & b[0];

   assign p[1] = (a[1] & b[0]) ^ (a[0] & b[1]);

   assign p[2] = c1 ^ (a[1] & b[1]);

   assign p[3] = c2;

endmodule
```

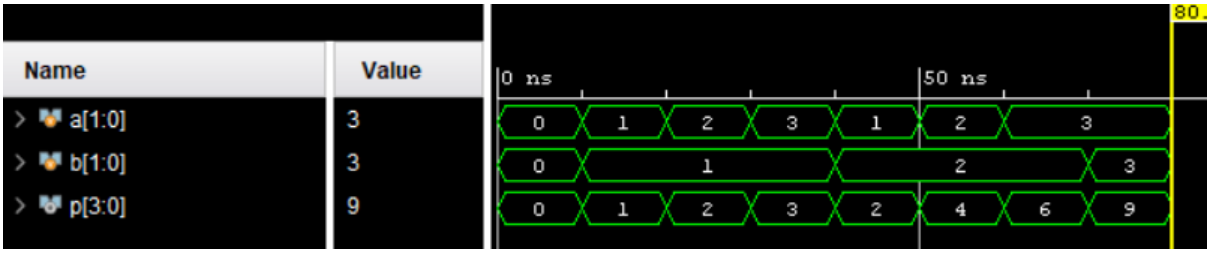


Figure 1: Wave form for 2-bit Multiplier

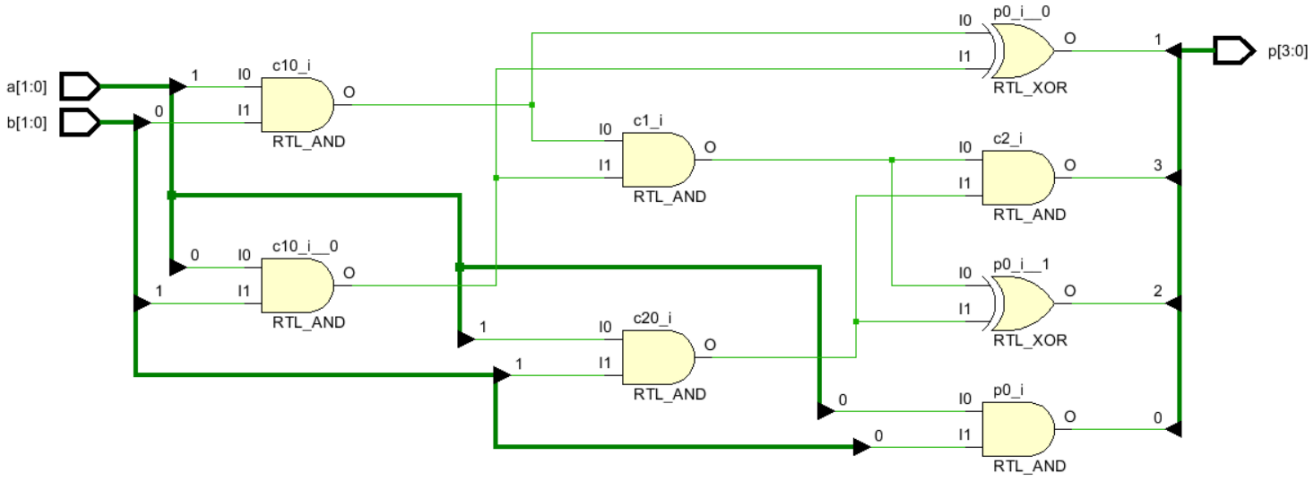


Figure 2: RTL Analysis schematic

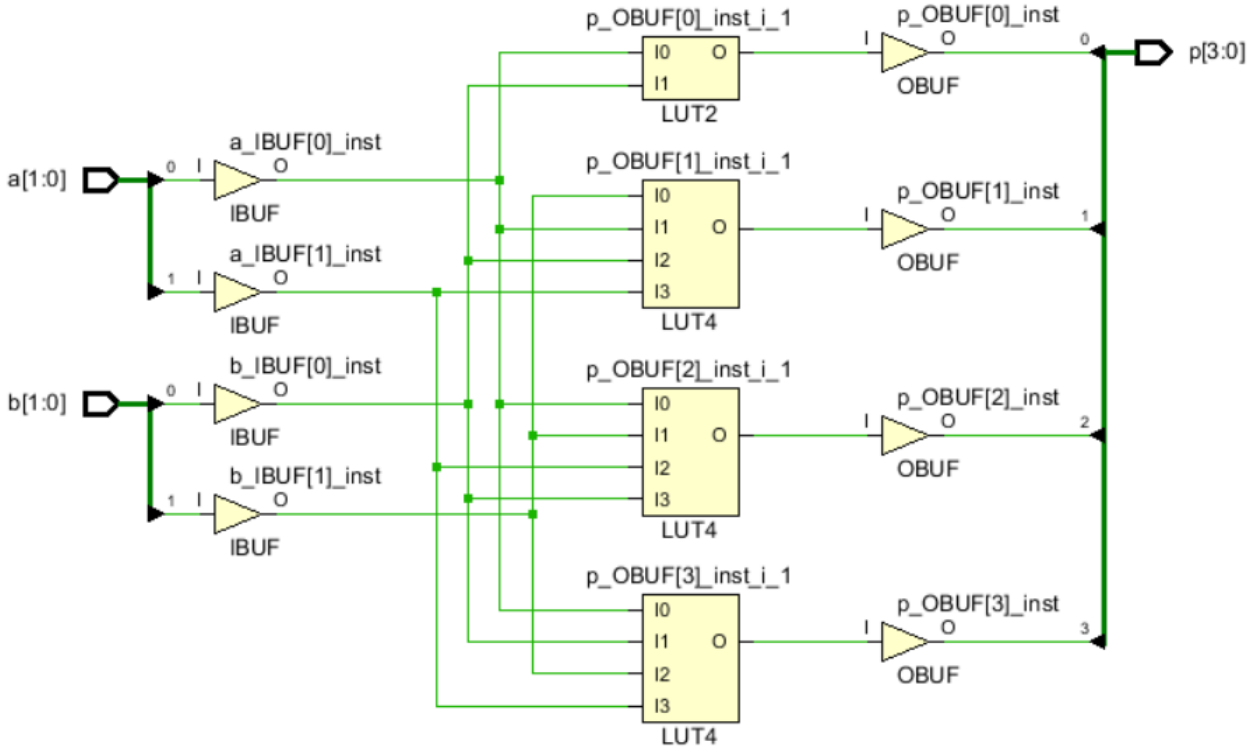


Figure 3: Synthesis schematic

Cell Properties

p_OBUF[0]_inst_i_1

I1	I0	O=I0 & I1
0	0	0
0	1	0
1	0	0
1	1	1

Table 1: LUT for p[0]

Cell Properties

p_OBUF[1]_inst_i_1

I3	I2	I1	I0	O=I0 & I1 & !I3 + I0 & I1 & !I2 + !I1 & I2 & I3 + !I0 & I2 & I3
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Table 2: LUT for p[1]

Cell Properties

p_OBUF[2]_inst_i_1

I3	I2	I1	I0	O=I1 & I2 & !I3 + !I0 & I1 & I2
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

Table 3: LUT for p[2]

Cell Properties

p_OBUF[3]_inst_i_1

I3	I2	I1	I0	O=I0 & I1 & I2 & I3
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Table 4: LUT for p[3]