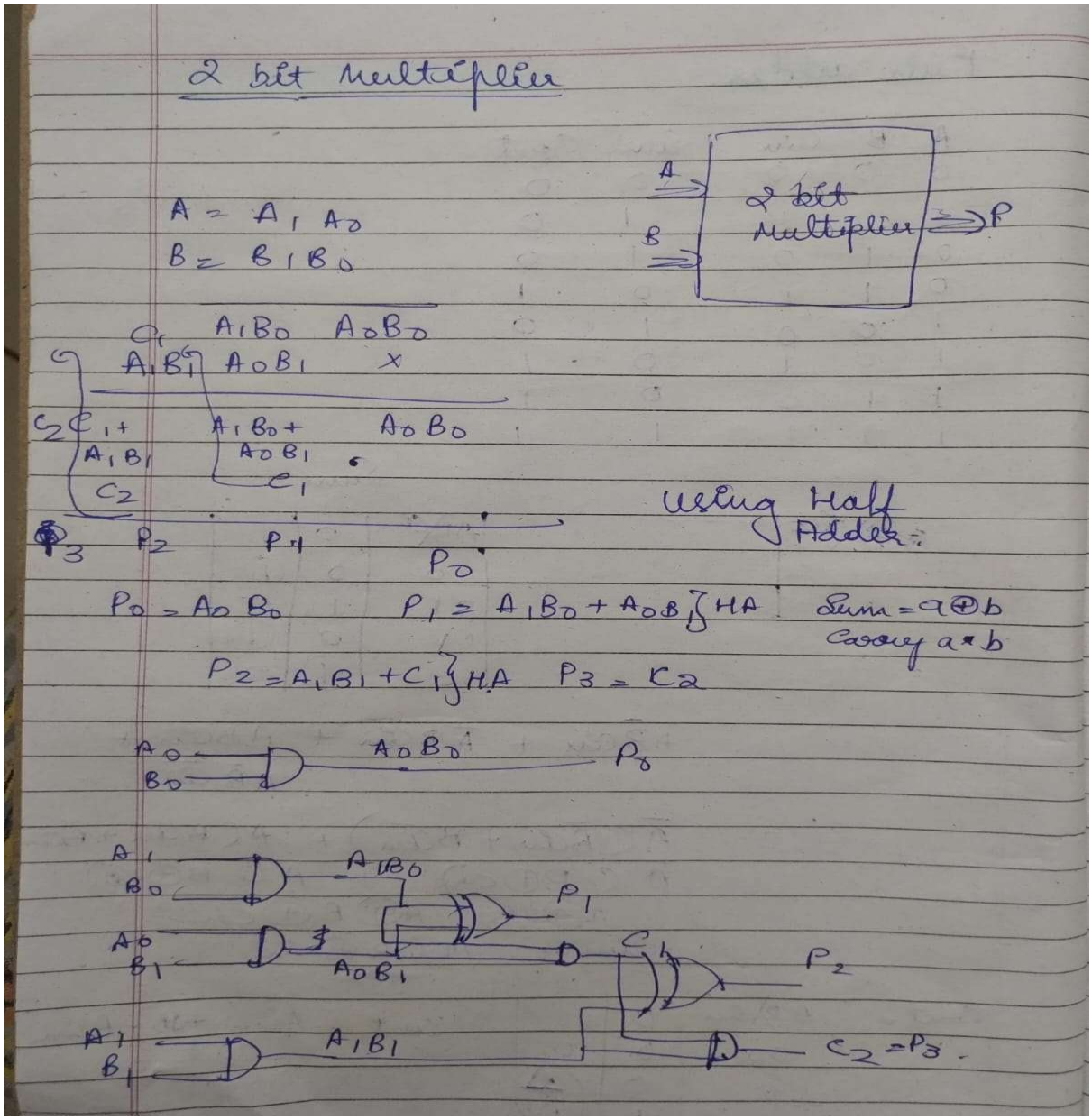
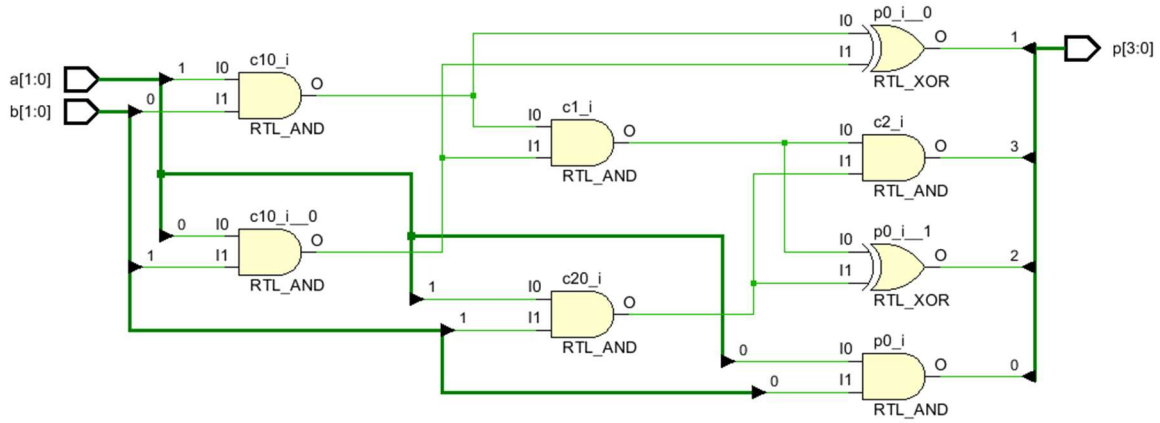
2 Bit Multiplier



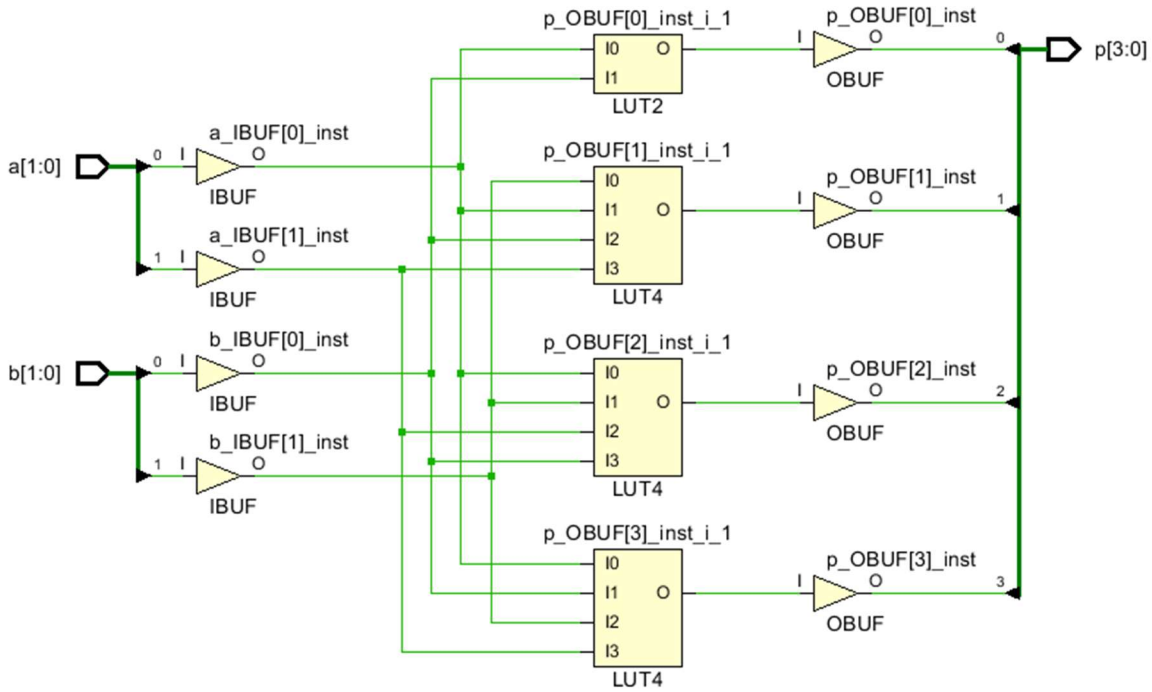
Verilog code

```
module multiplier(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
```

RTL analysis schematic:



Synthesis Schematic:



Look Up Table:

Cell Properties

p_OBUF[0]_inst_i_1

I1	I0	O=I0 ...
0	0	0
0	1	0
1	0	0
1	1	1

Figure 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

Figure 2: LUT for p[1]

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

Figure 3: LUT for p[3]

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

Figure 4: LUT for p[2]