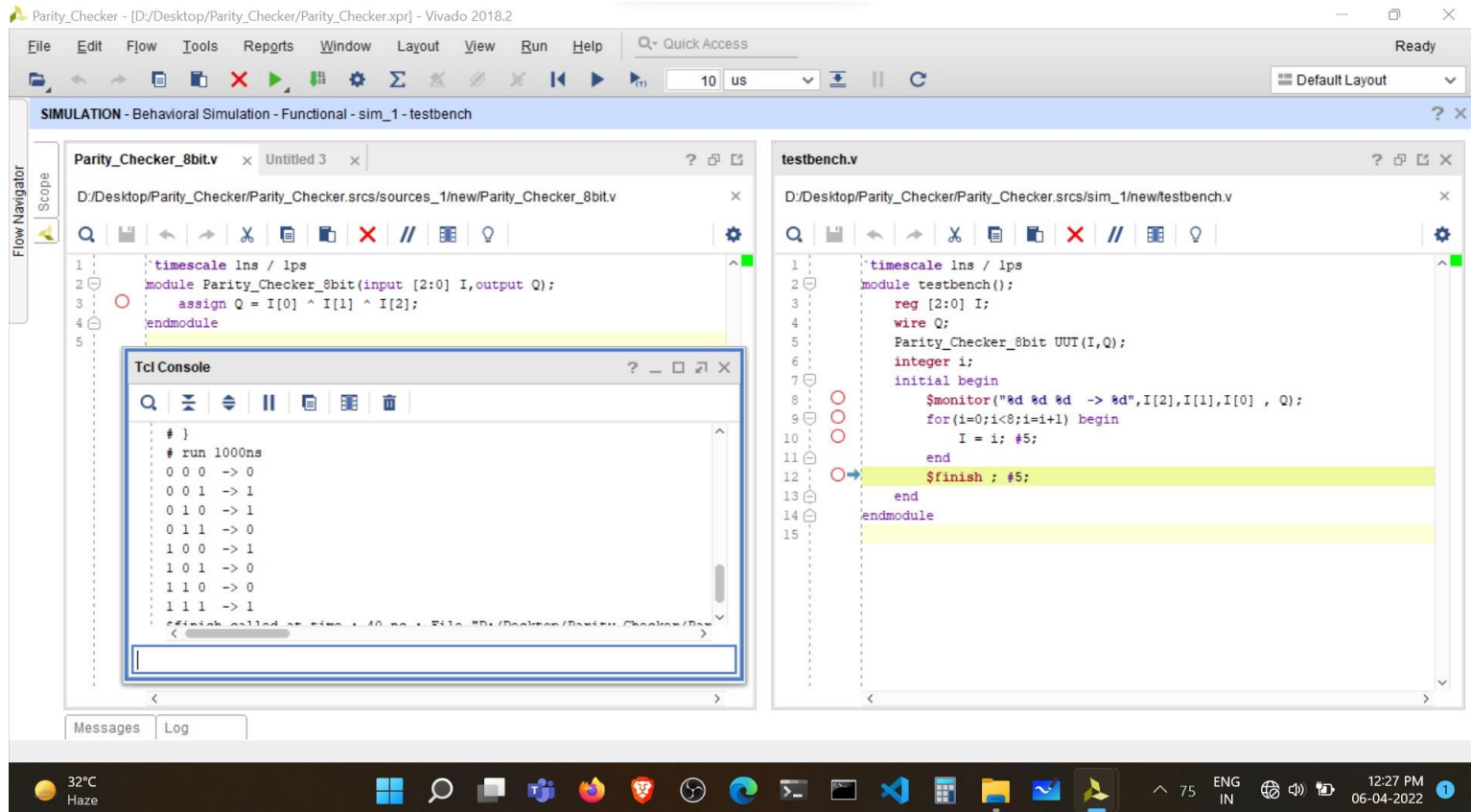
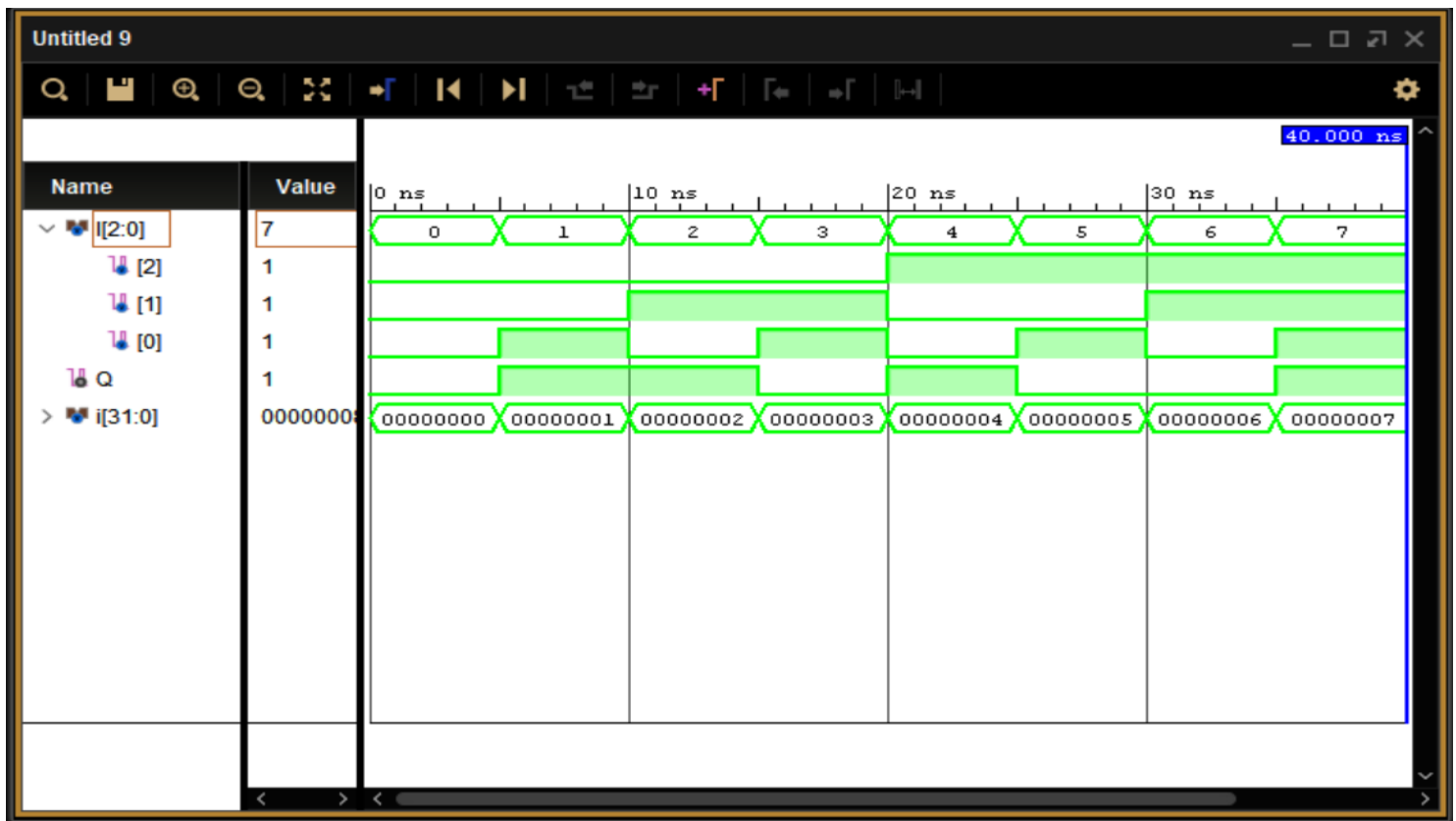


# Simulation Results:

- Parity Checker



Verilog Code and Test bench for 3-bit Parity Checker



Simulation output of 3-bit parity checker

- 8:1 Multiplexer

Assignment\_1B - [D:/Desktop/Mux\_8\_1/Assignment\_1B.xpr] - Vivado 2018.2

**PROJECT MANAGER - Assignment\_1B**

**Mux\_8\_1.v**

```

1 `timescale 1ns / 1ps
2 module Mux_8_1(input [7:0] I , [2:0] S , output Q );
3     assign Q = (~S[2] & ~S[1] & ~S[0] & I[0]) |
4               (~S[2] & ~S[1] & S[0] & I[1]) |
5               (~S[2] & S[1] & ~S[0] & I[2]) |
6               (~S[2] & S[1] & S[0] & I[3]) |
7               (S[2] & ~S[1] & ~S[0] & I[4]) |
8               (S[2] & ~S[1] & S[0] & I[5]) |
9               (S[2] & S[1] & ~S[0] & I[6]) |
10              (S[2] & S[1] & S[0] & I[7]);
11 endmodule

```

**testbench.v**

```

1 `timescale 1ns / 1ps
2 module testbench();
3     reg [7:0] I ;
4     reg [2:0] S ;
5     wire Q;
6     integer i = 0;
7     integer s = 0;
8     Mux_8_1 uut(I,S,Q);
9     initial begin
10        $monitor("%d%d%d%d%d%d%d%d %d %d %d -> %d",I[7],I[6],I[5],I[4],I[3],I[2],I[1],I[0],S[2],S[1],S[0],Q);
11        for(i=0;i<256;i=i+1) begin
12            I = i;
13            for(s=0;s<8;s=s+1) begin
14                S = s; #5;
15            end
16        end
17        $finish;#5;
18    end
19 endmodule
20

```

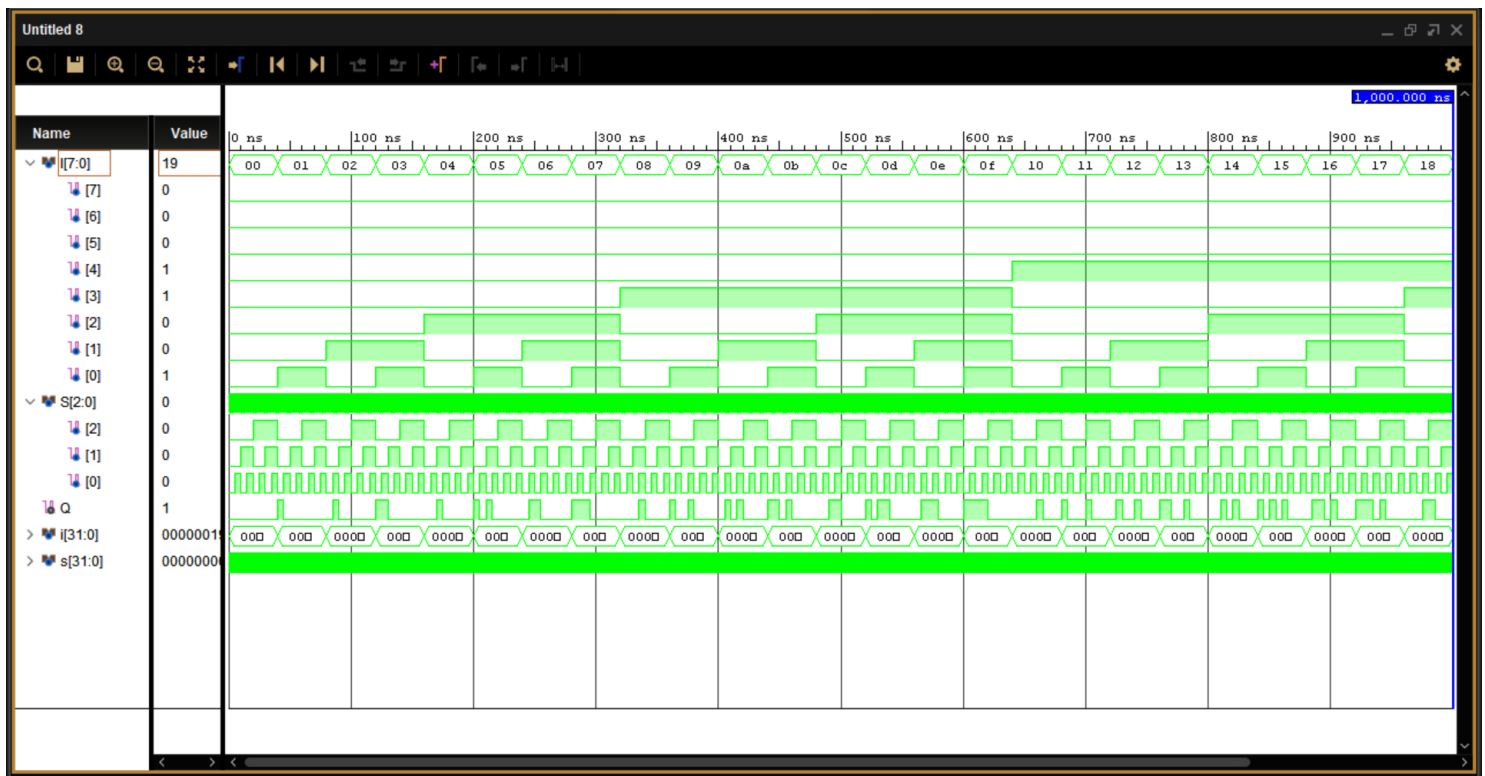
**Tcl Console**

```

00010011 0 0 1 -> 1
00010011 0 1 0 -> 0
00010011 0 1 1 -> 0
00010011 1 0 0 -> 1
00010011 1 0 1 -> 0
00010011 1 1 0 -> 0
00010011 1 1 1 -> 0
00010100 0 0 0 -> 0

```

Verilog Code and Test bench for 8-bit MUX



Simulation output 8:1 MUX

- 1:4 De-Multiplexer

Demux\_1\_4 - [D:/Desktop/Demux\_1\_4/Demux\_1\_4.xpr] - Vivado 2018.2

ELABORATED DESIGN - xc7vx485tffg1157-1 (active)

Project Summary | Schematic | Demux\_1\_4.v

D:/Desktop/Demux\_1\_4/Demux\_1\_4.srcs/sources\_1/new/Demux\_1\_4.v

```

1 `timescale 1ns / 1ps
2 module Demux_1_4(
3     input I,
4     input [1:0] S,
5     output [3:0] Q
6 );
7     assign Q[0] = ~S[1] & ~S[0] & I;
8     assign Q[1] = ~S[1] & S[0] & I;
9     assign Q[2] = S[1] & ~S[0] & I;
10    assign Q[3] = S[1] & S[0] & I;
11 endmodule

```

testbench.v

D:/Desktop/Demux\_1\_4/Demux\_1\_4.srcs/sim\_1/new/testbench.v

```

1 `timescale 1ns / 1ps
2 module testbench();
3     reg I;
4     reg [1:0] S;
5     wire [3:0] Q;
6     Demux_1_4 UUT(I,S,Q);
7     initial begin
8         $monitor("%d %d %d %d -> %d %d %d %d",I,S[1],S[0],Q[3],Q[2],Q[1],Q[0]);
9         I = 0; S = 2'b00; #5;
10        I = 0; S = 2'b01; #5;
11        I = 0; S = 2'b10; #5;
12        I = 0; S = 2'b11; #5;
13        I = 1; S = 2'b00; #5;
14        I = 1; S = 2'b01; #5;
15        I = 1; S = 2'b10; #5;
16        I = 1; S = 2'b11; #5;
17        $finish; #5;
18    end
19 endmodule

```

Tcl Console

```

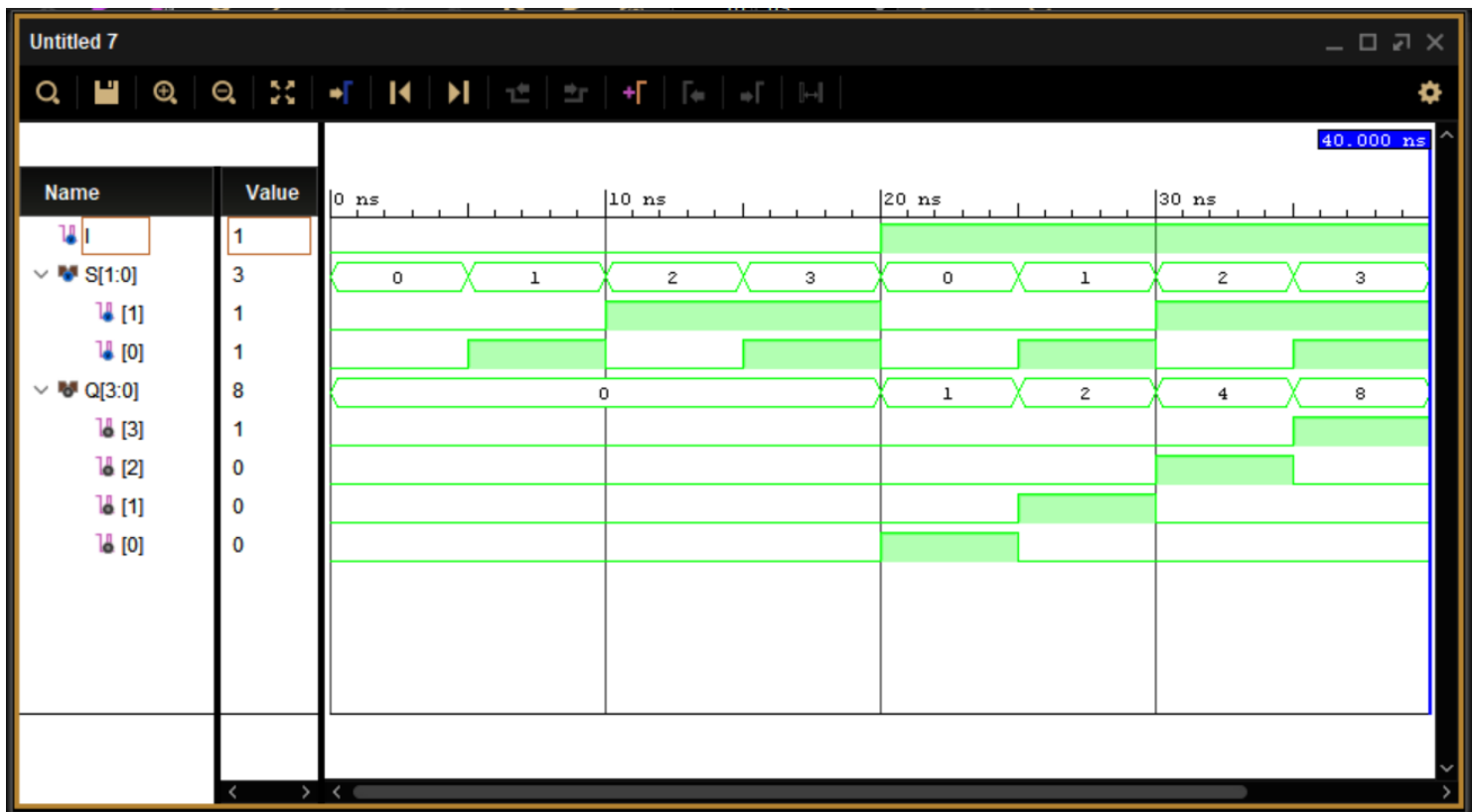
# run 1000ns
0 0 0 -> 0 0 0 0
0 0 1 -> 0 0 0 0
0 1 0 -> 0 0 0 0
0 1 1 -> 0 0 0 0
1 0 0 -> 0 0 0 1
1 0 1 -> 0 0 1 0
1 1 0 -> 0 1 0 0
1 1 1 -> 1 0 0 0

```

32°C Haze

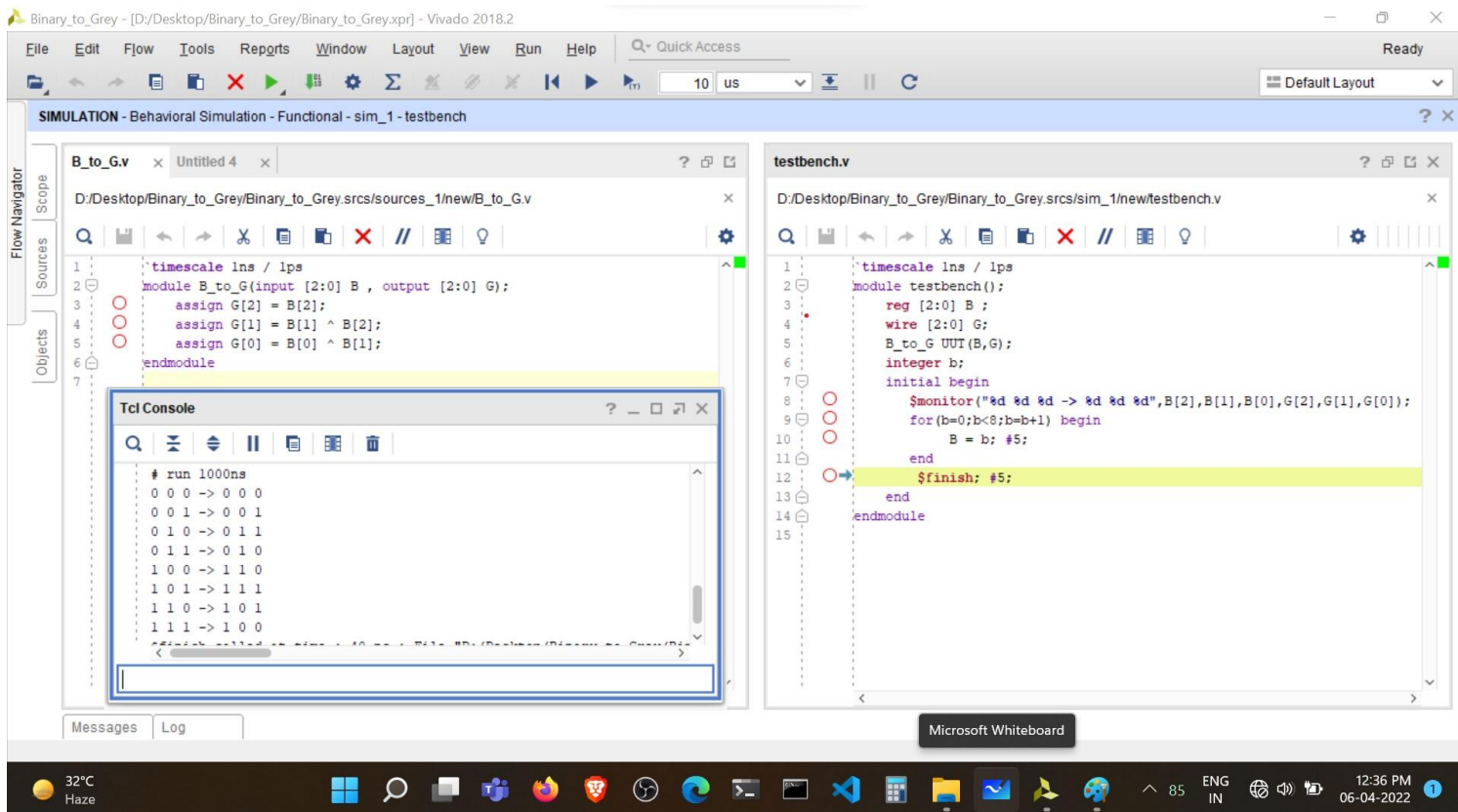
95 ENG IN 12:58 PM 06-04-2022

Verilog Code and Test bench for 1:4 De-MUX

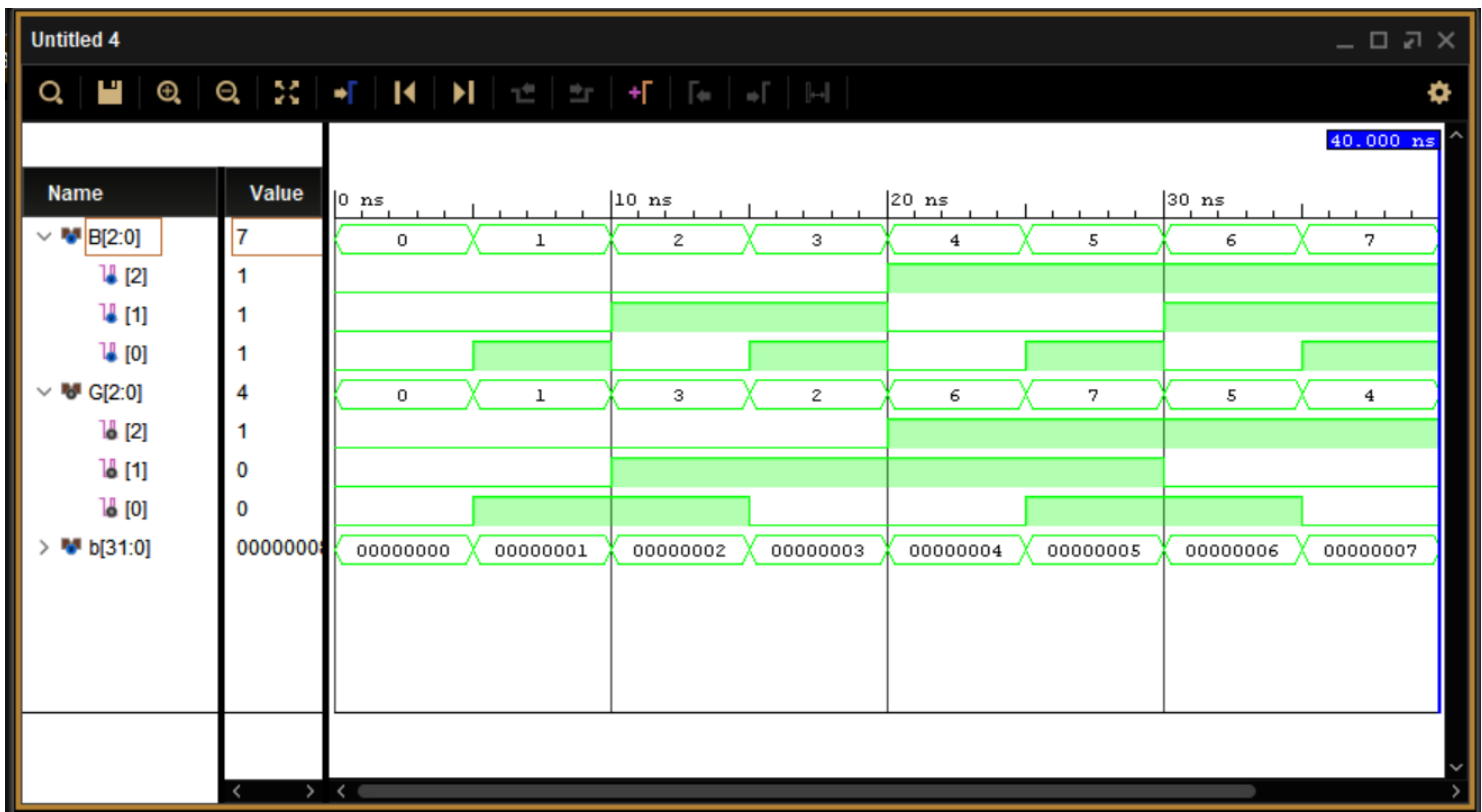


Simulation output of 1:4 De-MUX

## • Binary to Gray Converter

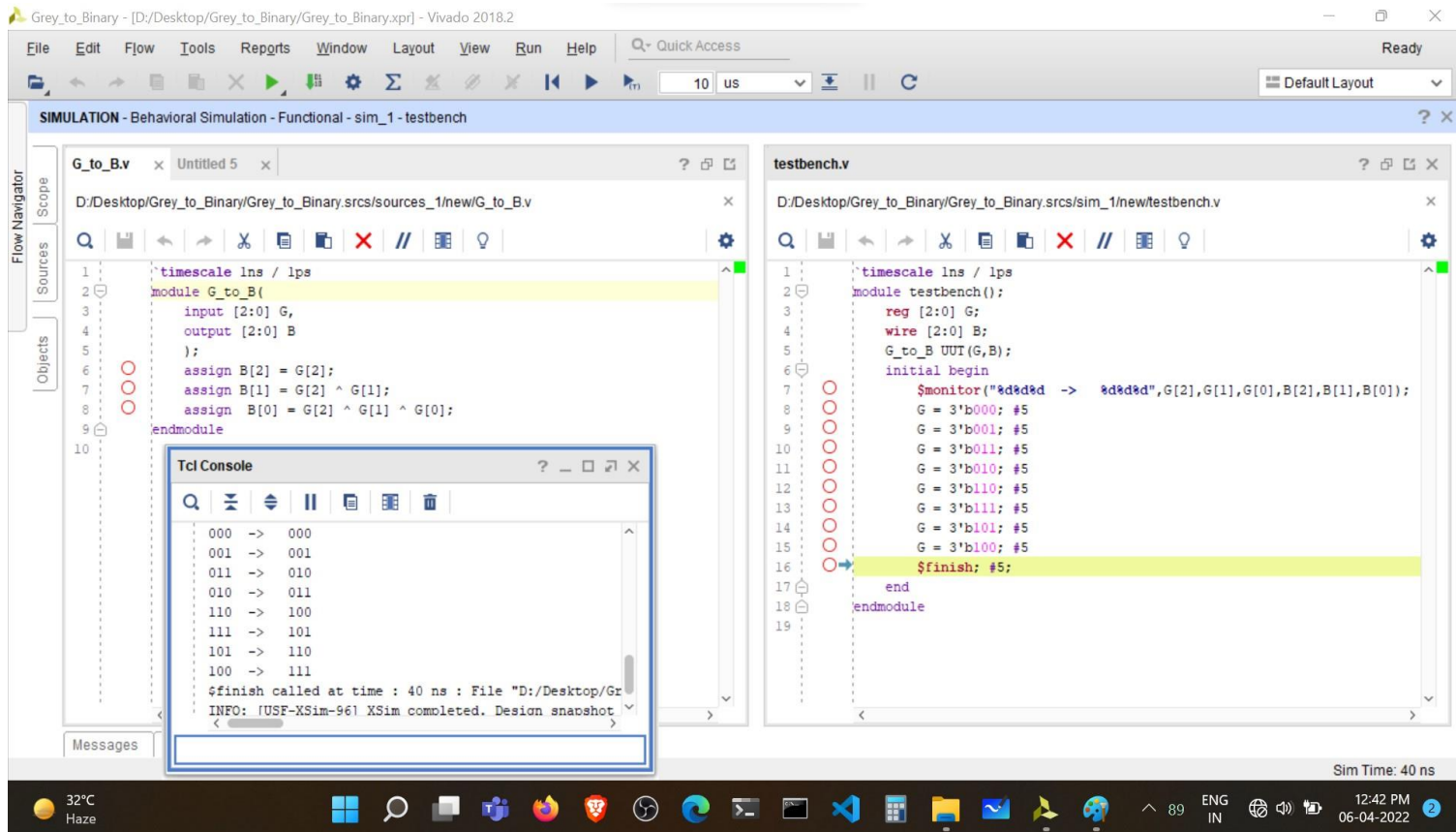


Verilog Code and Test bench of Binary to Gray Converter



Simulation output of Binary to Gray Converter

- Gray to Binary Converter



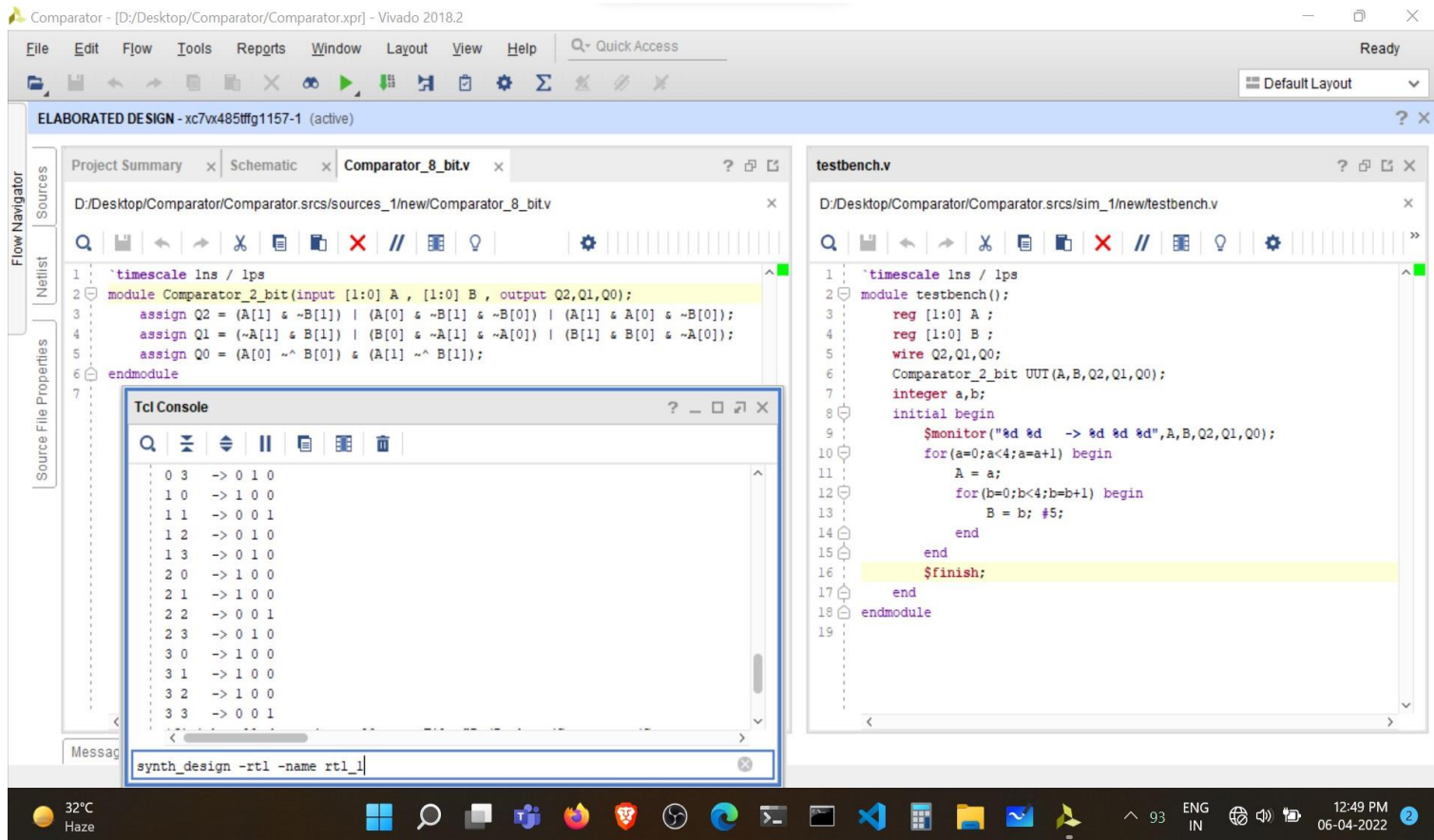
Verilog Code and Test bench for Gray to Binary Converter



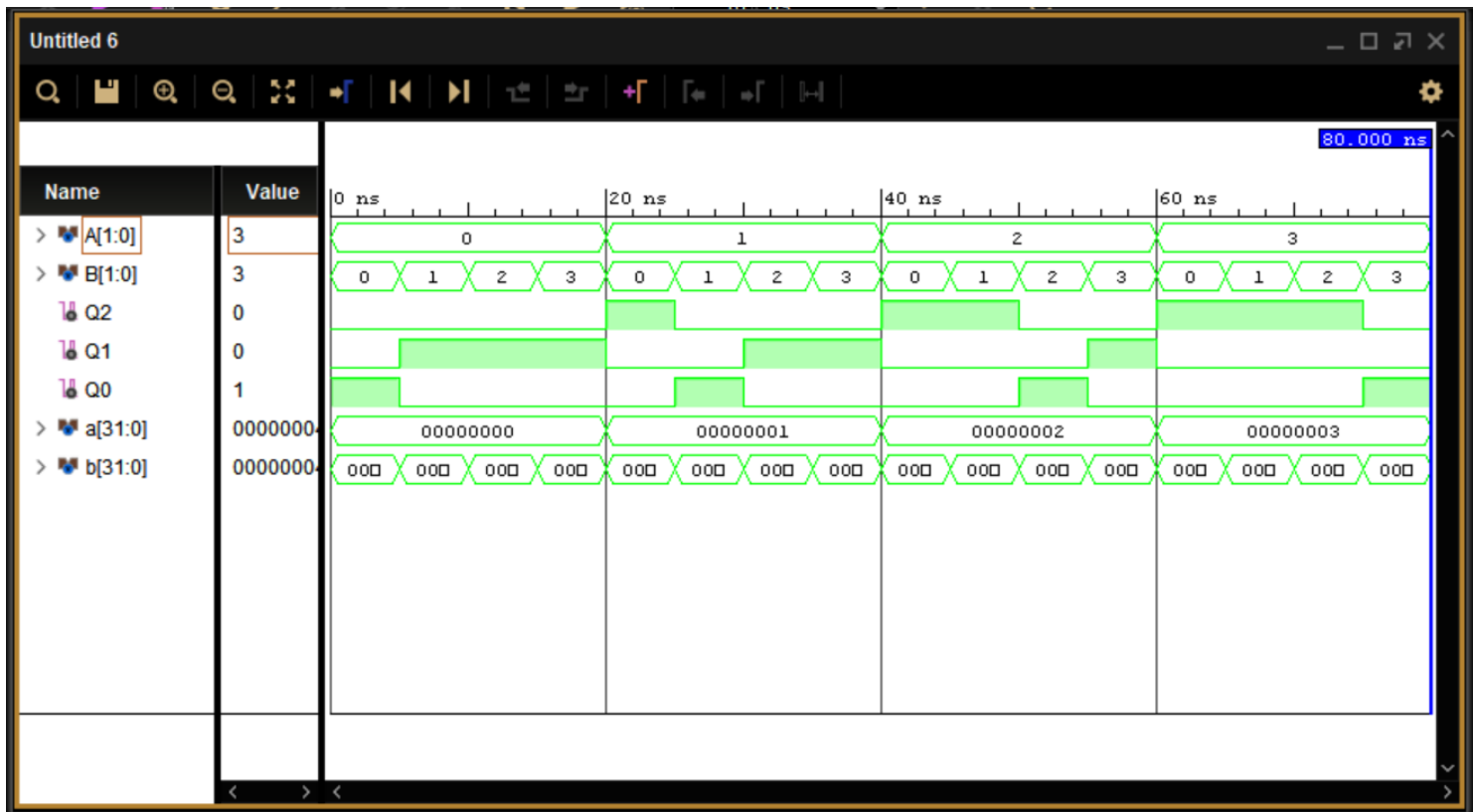
Simulation output of Gray to Binary Converter



## • 2-bit Magnitude Comparator



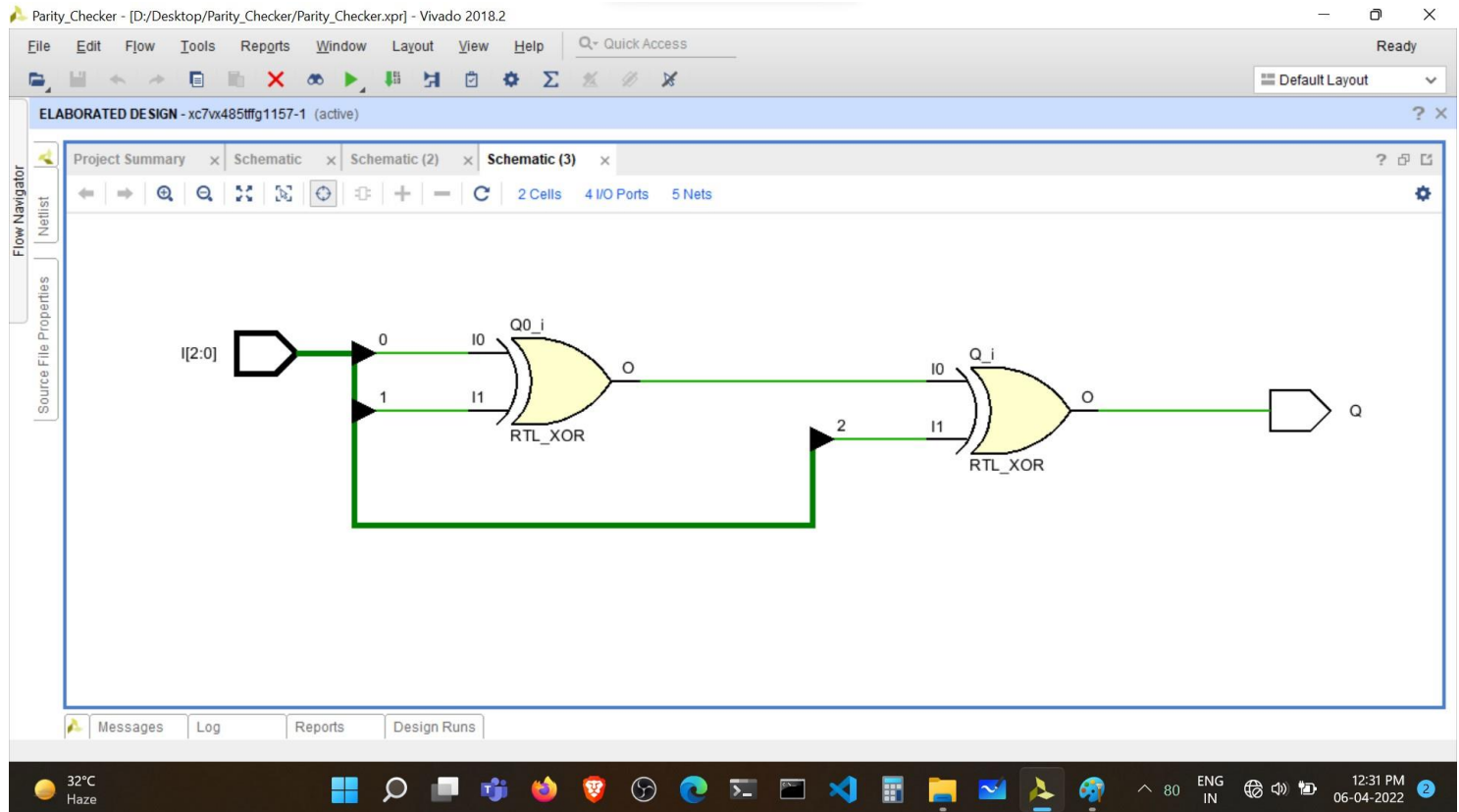
Verilog Code and Test bench for 2-bit magnitude Comparator



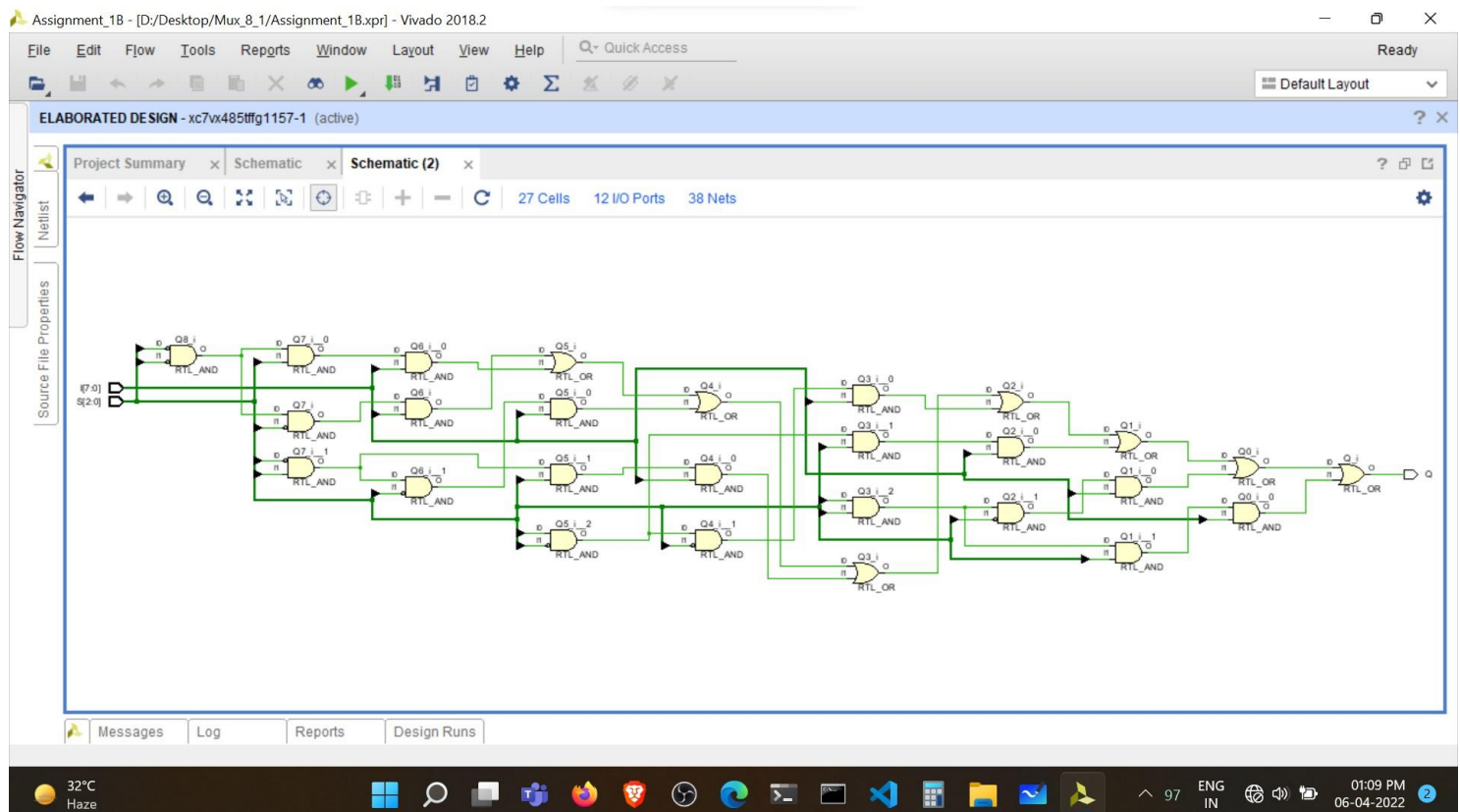
Simulation output of 2-bit magnitude Comparator

# Observations:

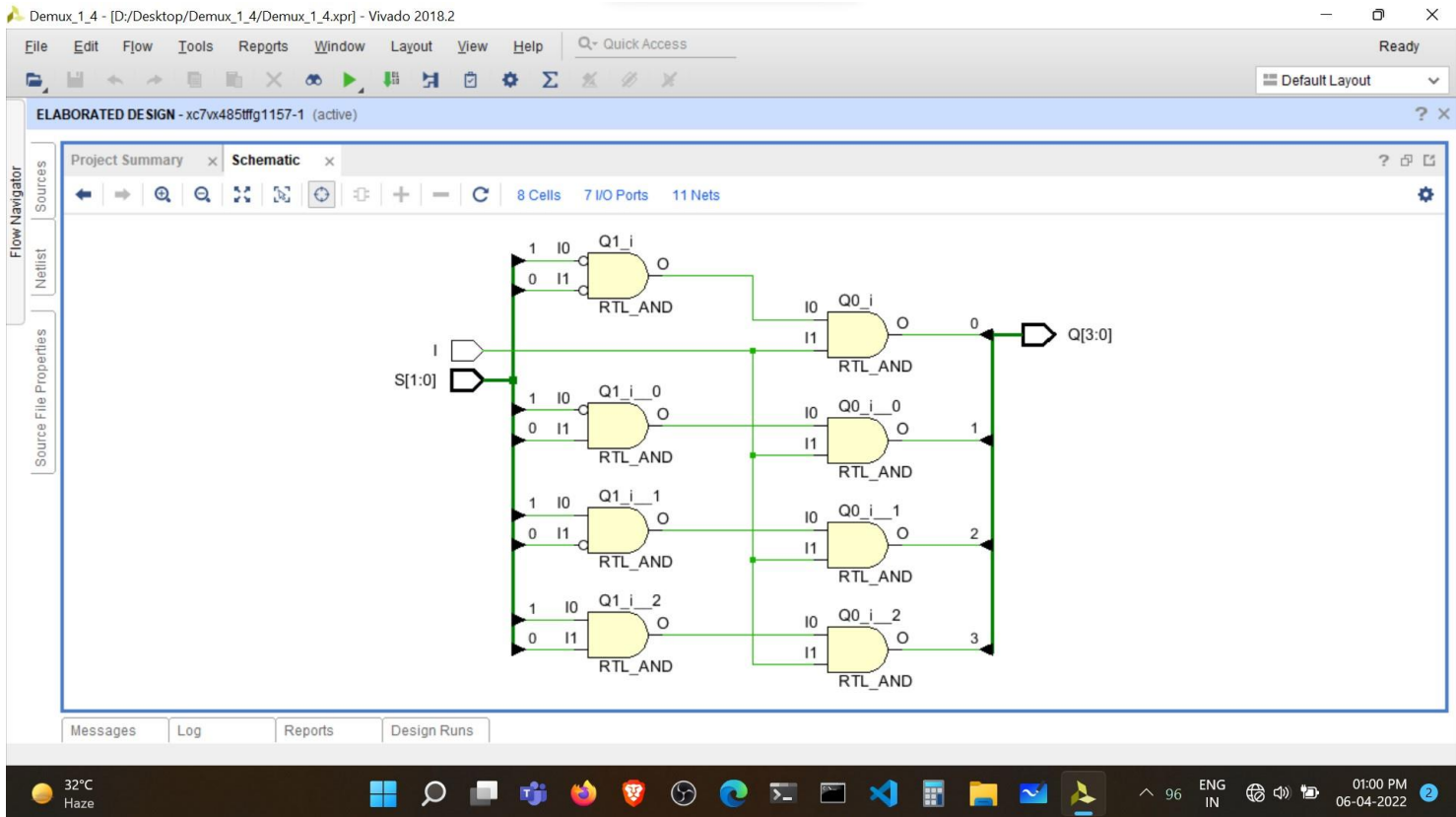
- Parity Checker



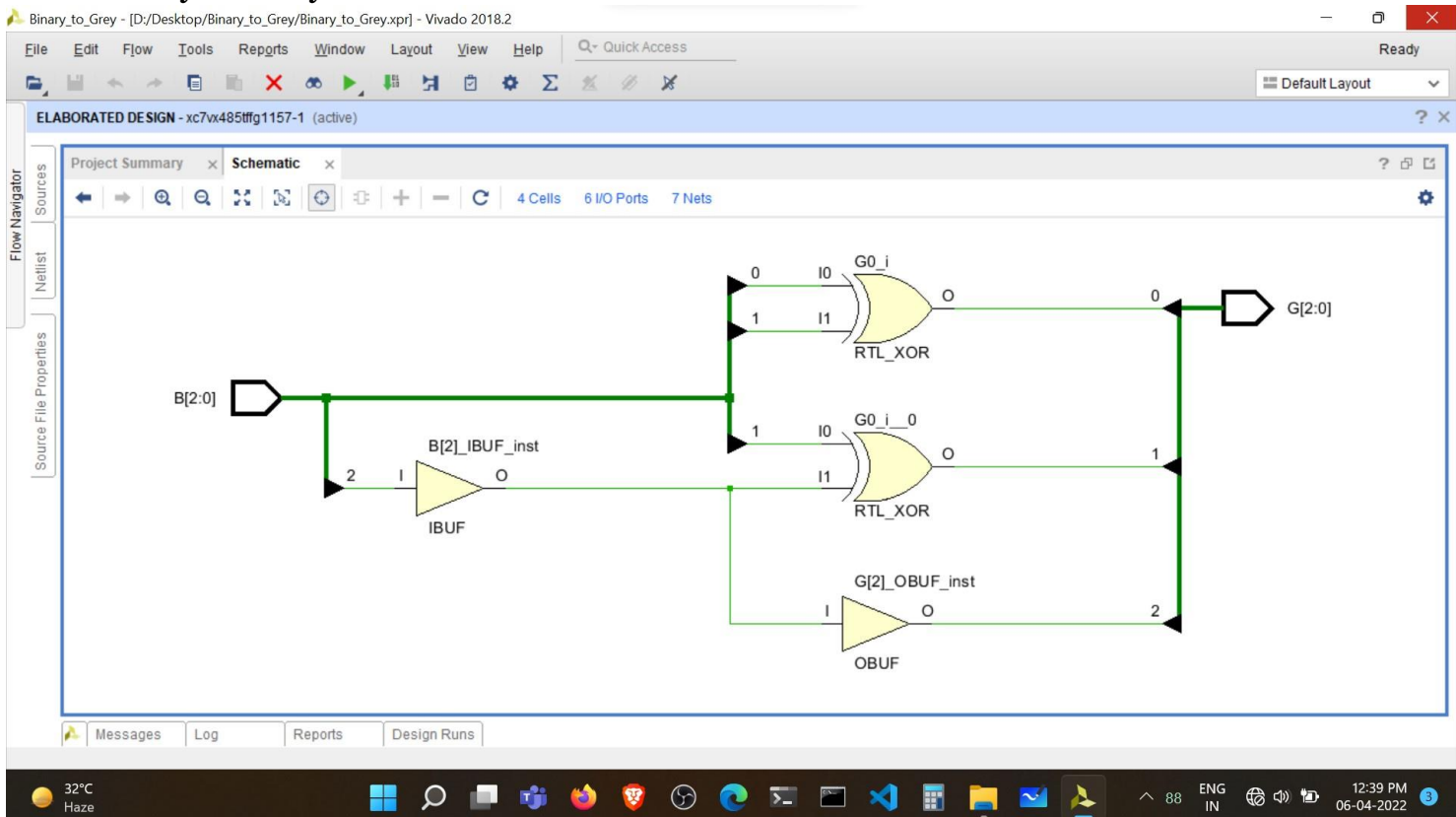
- 8:1 Multiplexer



- 1:4 De-Multiplexer

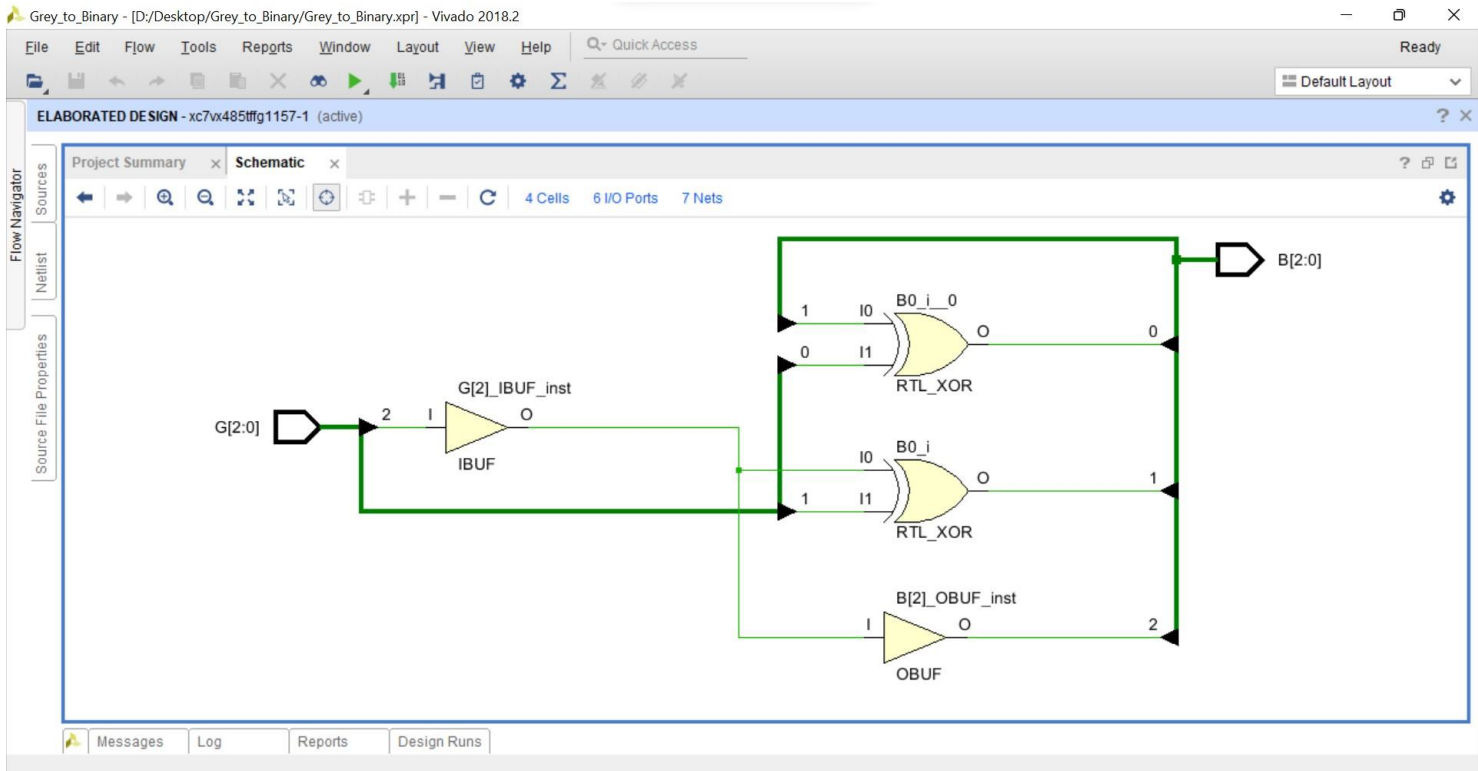


- Binary to Gray Converter





- Gray to Binary Converter



- 2-bit Magnitude Comparator

