Lab workshop on FPGA Architecture and programming using Verilog

fpga0622

Prepared by

PRINCE

Verilog code:

```
`timescale 1ns / 1ps
module miniproject(
    input clk,
    //input i0,input i1,input i2,input i3,
    //output reg[3:0] y,
    output reg[3:0] Y
    //output [1:0] count
    );
```

MUX4to1:

```
    wire [1:0] count;
    //wire dat;
    wire [3:0] y;
   c_counter_binary_2 your_instance_name (
  .CLK(clk),  // input wire CLK
  .Q(count)      // output wire [1 : 0] Q
);
```

```
assign dat=count[1] ? (count[0] ? i3 : i2) : (count[0] ? i1 : i0);
```

D Latch:

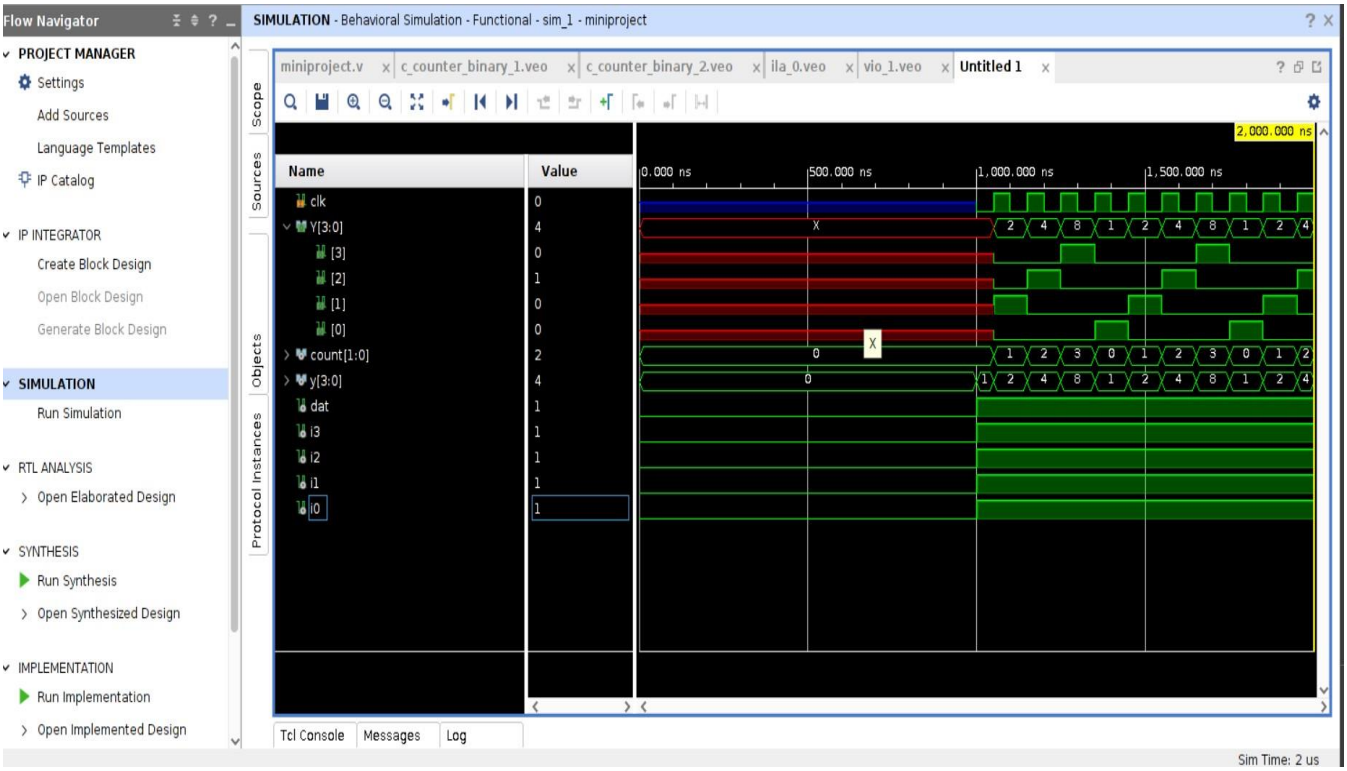
```
decoder_2x4 decoder (count, dat, y);
always@(*)
if(clk)
Y<=y;
ila_0 your_result (
        .clk(clk), // input wire clk



        .probe0(Y) // input wire [3:0] probe0
);
vio_1 your_input_name (
  .clk(clk),                // input wire clk
  .probe_out0({i3,i2,i1,i0})  // output wire [3 : 0] probe_out0
);

Endmodule
```

2x4Decoder:

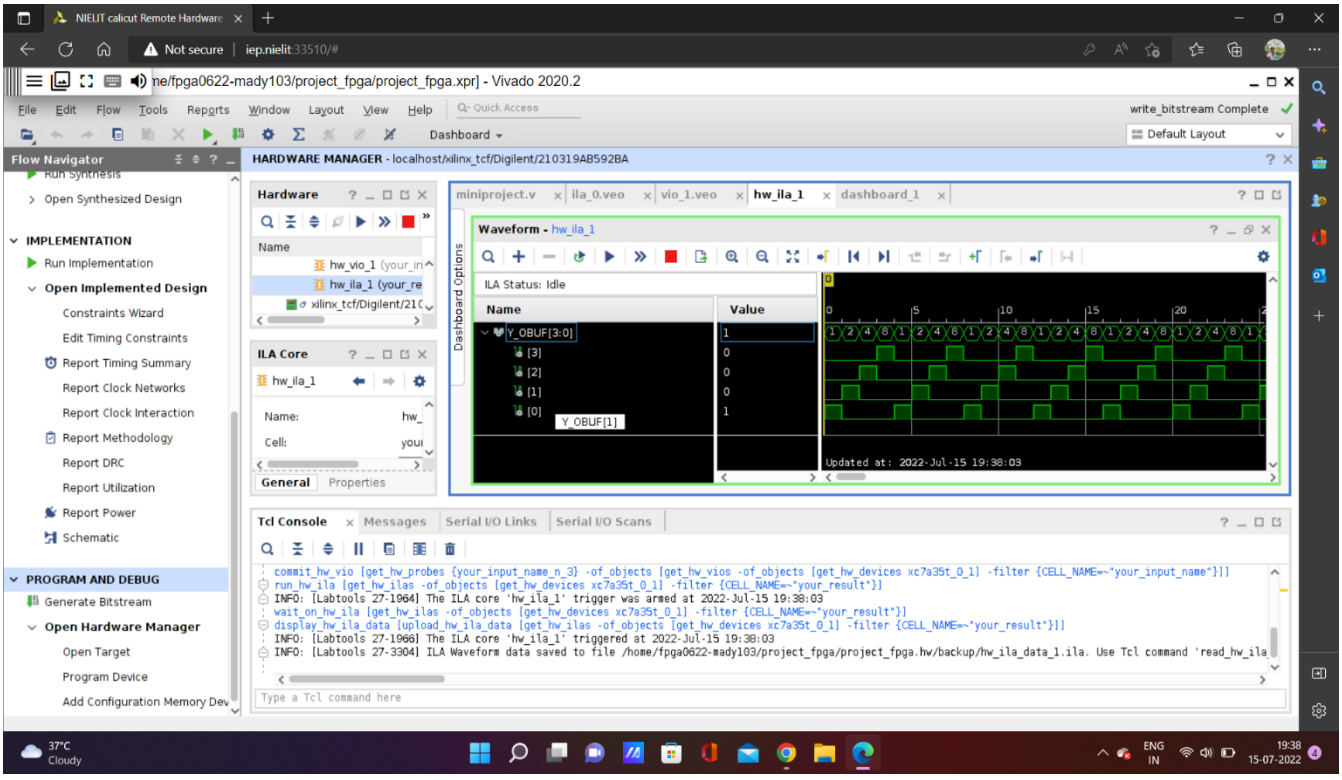
```
module decoder_2x4(
 input [1:0] in,
```

```
 input en,
 output reg [3:0] out
 );
always@(*)
if(en)
case(in)
0:out = 1;
1:out = 2;
2:out = 4;
3:out = 8;
default: out = 0;
endcase
else
out = 0;
endmodule
```

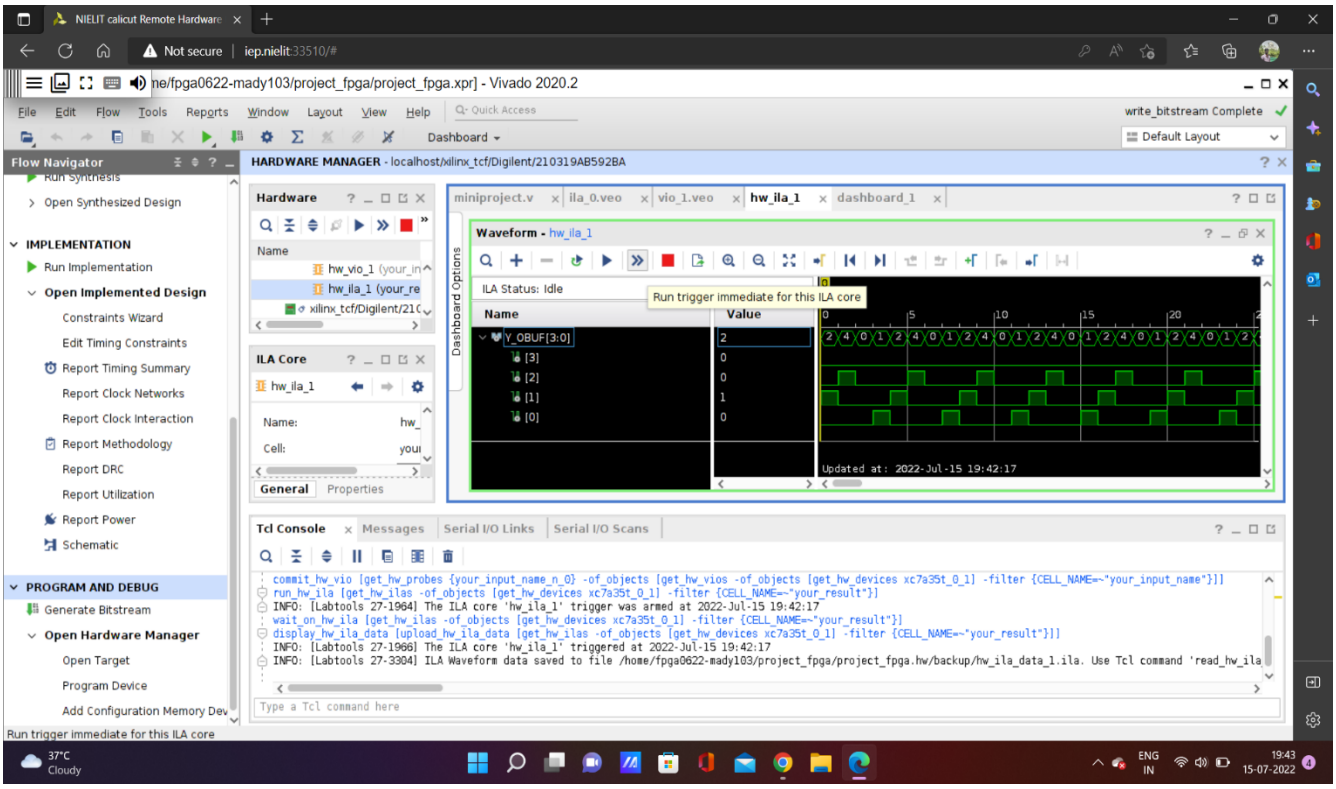
Waveforms:



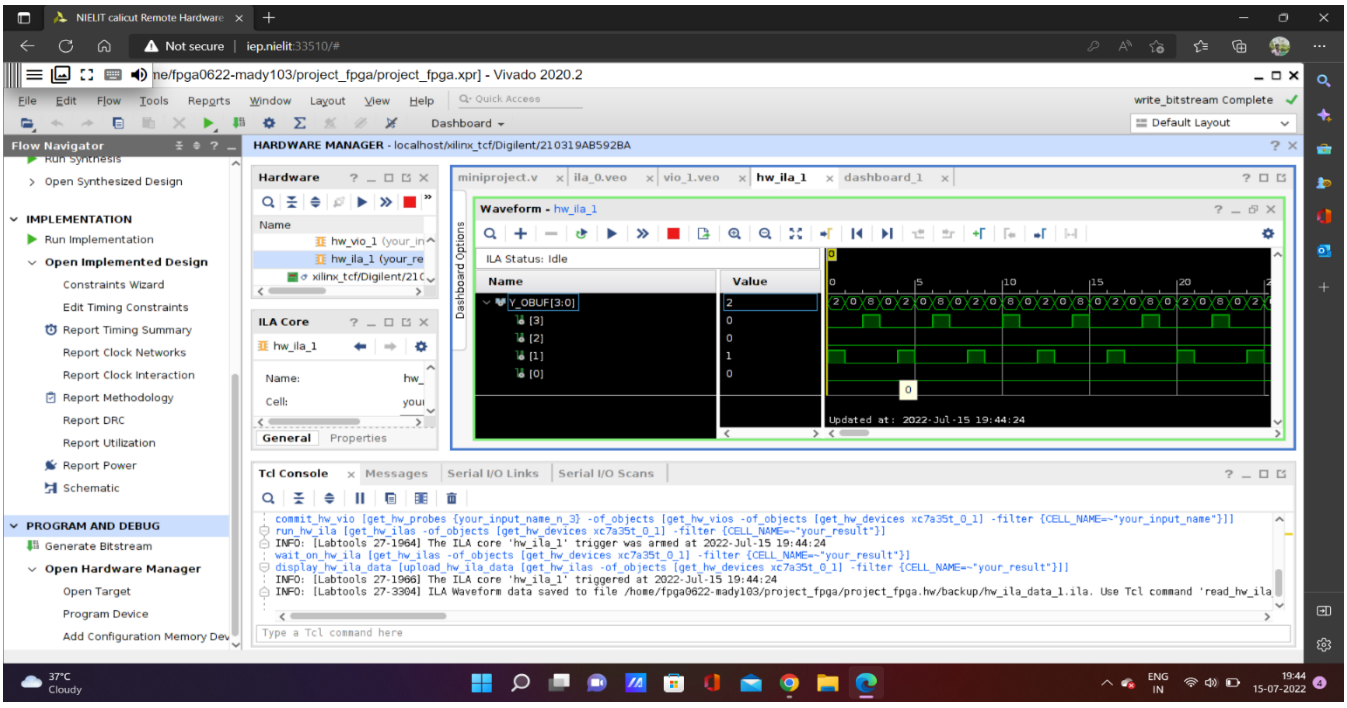
When the input is 4’b1111:



When the input is 4’b0110:



When the input is 4’b1010:



When the input is 4’b1011: