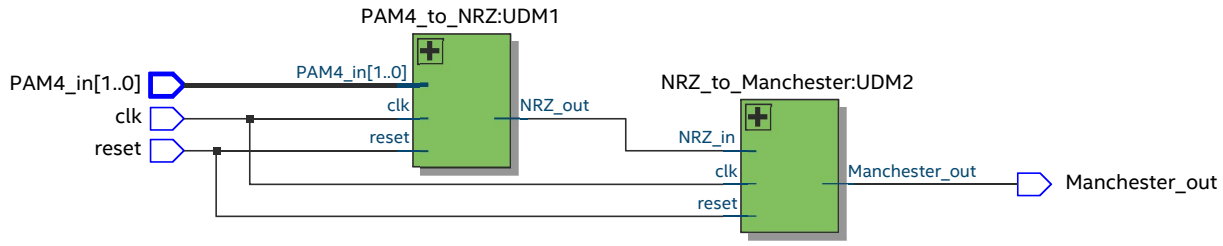
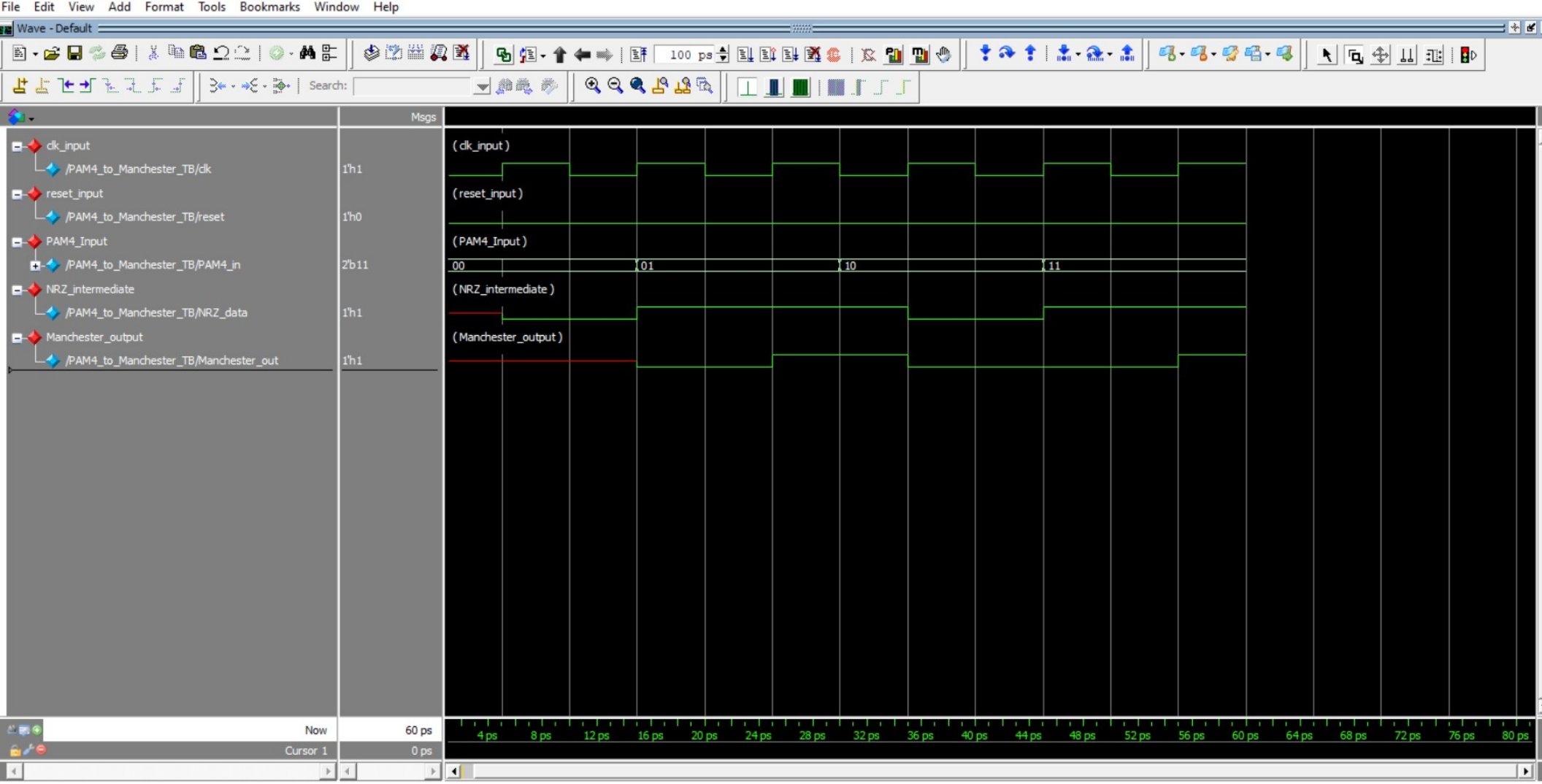
```
1   module PAM4_to_Manchester (
2                              Manchester_out,
3                              PAM4_in,
4                              clk,
5                              reset
6                              );
7
8   //assigen the inputs and outputs as registers and wires
9   input  wire       clk, reset;           //clk and reset signal
10  input  wire [1:0] PAM4_in;             //2 bits PAM4_in input signal
11  output wire       Manchester_out;      //declare as wire since it will be connected to module
12
13  //internal probe
14  wire NRZ_data;             //output signal from the PAM4_to NRZ module
15
16  //instantiate the PAM4-NRZ and NRZ-Manchester modules as user define modules (UDM1 and UDM2)
17  PAM4_to_NRZ UDM1 (
18                    .NRZ_out(NRZ_data),                          //note NRZ_out becomes
    NRZ_data
19                    .PAM4_in(PAM4_in),
20                    .clk(clk),
21                    .reset(reset)
22                   );
23
24  NRZ_to_Manchester UDM2 (
25                          .Manchester_out(Manchester_out),
26                          .NRZ_in(NRZ_data),                     //note NRZ_in becomes NRZ_data
27                          .clk(clk),
28                          .reset(reset)
29                         );
30
31  endmodule
```


PAM4_to_NRZ:UDM1
PAM4_in[1..0]
clk
reset
PAM4_in[1..0]
clk
reset
NRZ_out
NRZ_to_Manchester:UDM2
NRZ_in
clk
reset
Manchester_out
Manchester_out

```
1   module PAM4_to_Manchester_TB ();
2
3   //define the registers=wire and wires=registers from desing-to-testbench
4   reg       clk, reset;              //clk and reset signal
5   reg [1:0] PAM4_in;                //2 bits PAM4_in input signal
6   wire       Manchester_out;
7
8   //define the unit under test UUT
9   PAM4_to_Manchester UUT (
10                          .Manchester_out(Manchester_out),
11                          .PAM4_in(PAM4_in),
12                          .clk(clk),
13                          .reset(reset)
14                          );
15
16  //monitor internal probe
17  assign NRZ_data = UUT.UDM1.NRZ_out;
18
19  //instantiate the clk signal
20  initial
21     begin
22        clk = 1'b0;
23        forever #5 clk = ~clk;     //10ns clk period
24     end
25
26  //instantiate the reset signal
27  initial
28     begin
29              reset = 1'b0;       //togel the reset signal on
30              PAM4_in  = 2'b00;
31        #100 reset = 1'b1;      //toggle the reset signal off
32     end
33
34  //instantiate all the posibble states for PAM4_in with time intervals
35  initial
36     begin
37        PAM4_in = 2'b00; #15;
38        PAM4_in = 2'b01; #15;
39        PAM4_in = 2'b10; #15;
40        PAM4_in = 2'b11; #15;
41
42        $stop;
43     end
44
45  //display the results
46  initial begin
47    $display("PAM4_in--------Binary/NRZ_out-------Manchester_out");
48    $monitor("%b          %b           %b ",PAM4_in, NRZ_data,
    Manchester_out);
49   end
50  endmodule
```


File Edit View Add Format Tools Bookmarks Window Help
Wave - Default
Search:
Msgs
clk_input
/PAM4_to_Manchester_TB/clk
1'h1
reset_input
/PAM4_to_Manchester_TB/reset
1'h0
PAM4_Input
/PAM4_to_Manchester_TB/PAM4_in
2'b11
NRZ_intermediate
/PAM4_to_Manchester_TB/NRZ_data
1'h1
Manchester_output
/PAM4_to_Manchester_TB/Manchester_out
1'h1
(clk_input)
(reset_input)
(PAM4_Input)
00
01
10
11
(NRZ_intermediate)
(Manchester_output)
Now
60 ps
Cursor 1
0 ps
4 ps
8 ps
12 ps
16 ps
20 ps
24 ps
28 ps
32 ps
36 ps
40 ps
44 ps
48 ps
52 ps
56 ps
60 ps
64 ps
68 ps
72 ps
76 ps
80 ps

Output table

```
# PAM4_in--------Binary/NRZ_out-------Manchester_out
# 00                    x                   x
# 00                    0                   x
# 01                    1                   0
# 01                    1                   1
# 10                    1                   1
# 10                    0                   0
# 11                    1                   0
# 11                    1                   1
```

PAM4 to Manchester out

- This module instantiates 2 different modules
- The final modules take PAM4 data converts the data to NRZ
- The NRZ data is converter to Manchester code as final output