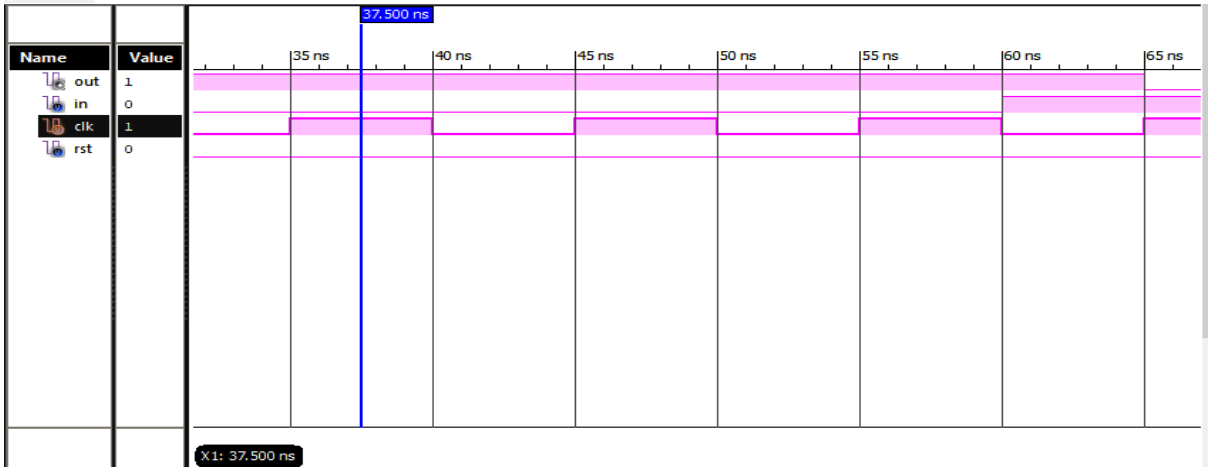
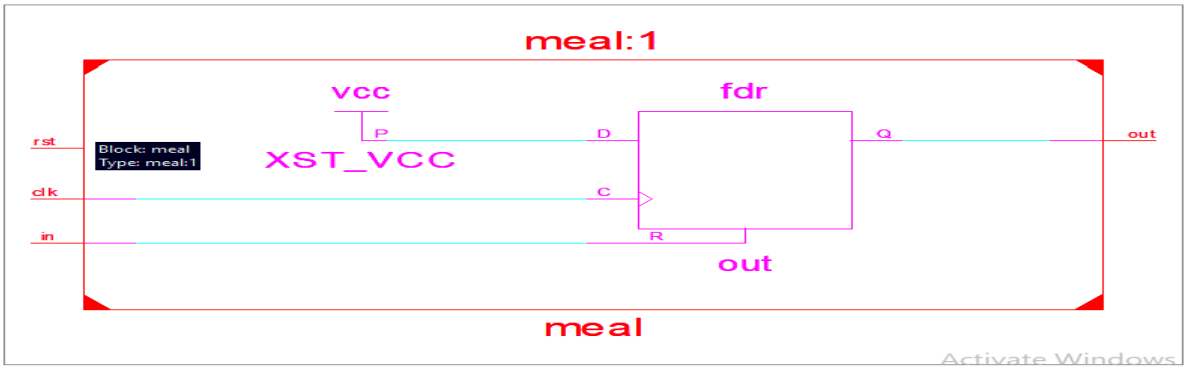
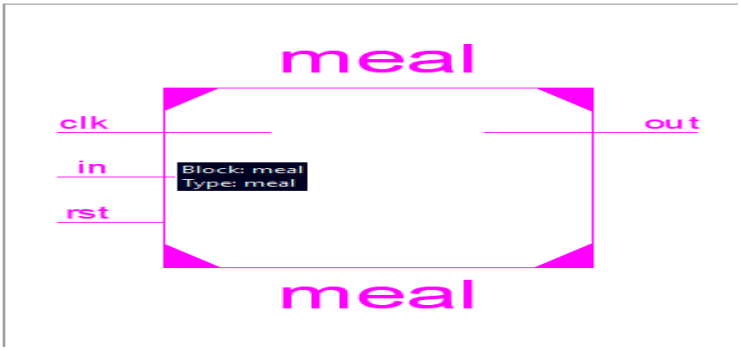
```
21  module meal(in,clk,rst,out);
22  input in,clk,rst;
23  output reg out;
24  reg [1:0] cst;
25  reg [1:0] nst;
26  parameter [1:0] s0=2'b00;
27  parameter [1:0] s1=2'b01;
28  parameter [1:0] s2=2'b10;
29  always@ (posedge clk)
30  begin
31  if(rst)
32  out=1'b0;
33  cst=s0;
34  nst=s0;
35  //else
36  begin
37  cst=nst;
38  case(cst)
39  s0:if(in) begin
40  out=1'b0;
41  nst=s1;
42  end
43  else begin
44  out=1'b1;
45  nst=s0;
46  end
47  s1:if(in) begin
48  out=1'b1;
49  nst=s2;
50  end
51  else begin
52  out=1'b0;
53  nst=s1;
54  end
55  s2:if(in) begin
56  out=1'b0;
57  nst=s1;
58  end
59  else begin
60  out=1'b1;
61  nst=s0;
62  end
63  endcase
64  end
65  end
66  endmodule
```




meal:1
vcc
fdr
rst
Block: meal
Type: meal:1
P
D
Q
out
XST_VCC
clk
C
in
R
out
meal
Activate Windows


meal
clk
out
in
Block: meal
Type: meal
rst
meal