Electronics Club

Task-3

Chirag Hegde

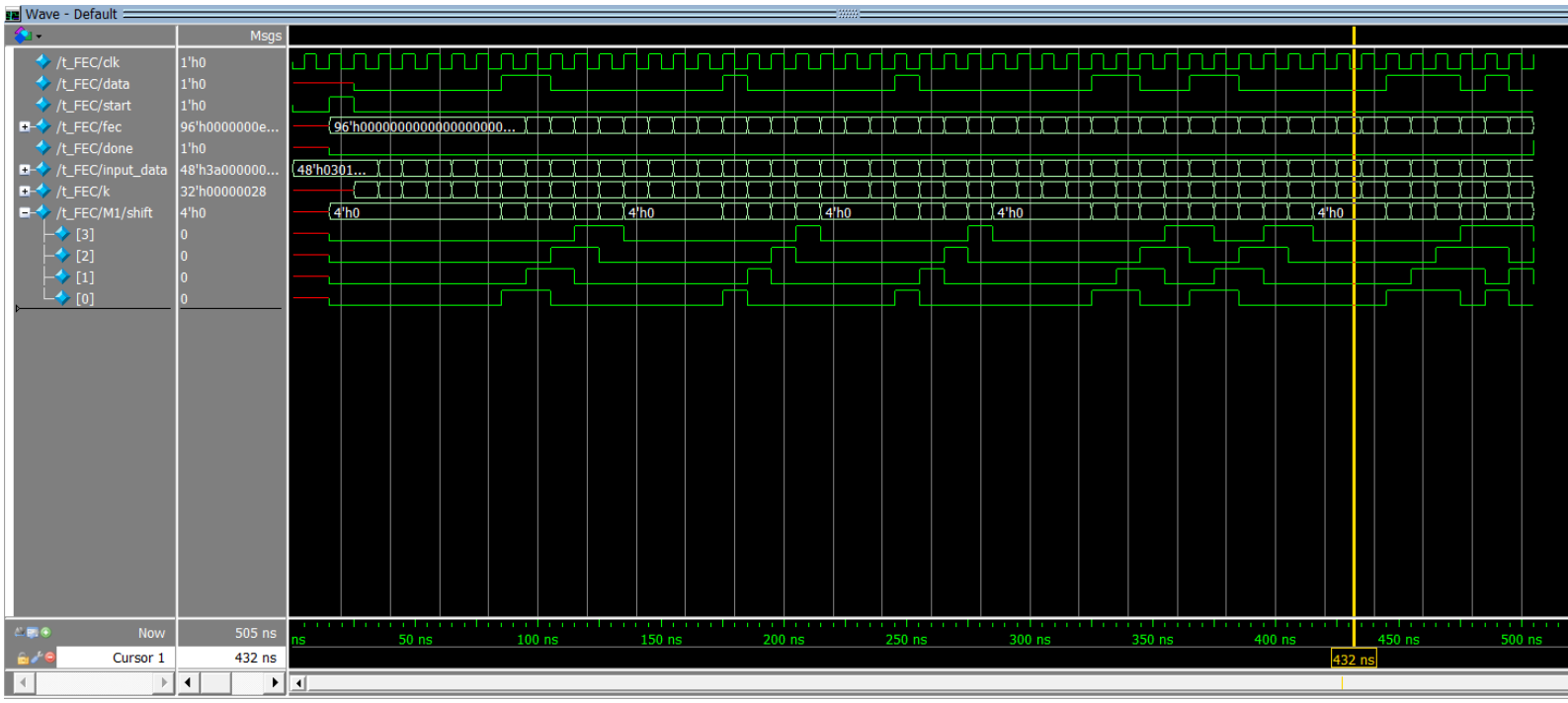
Design.v:

```
module FEC(
input clk,
input data, start,
output reg [95:0] fec,
output done
);
  wire temp=data;

  reg [6:0] count;
  reg [3:0] shift;


  always @(posedge clk, posedge start) begin
    if(start) begin
      fec<=96'd0;
      count<=49;
      shift<=4'b0000;
    end

   else if(count!=0) begin
     shift[3:1] <= shift[2:0]; //left shift by 1 bit
     shift[0]<=data;
     //$display("%b %b",shift,data); Debugging
     fec[0] <= shift[0]+shift[1]+shift[2]+shift[3]; //p0=x[n]+x[n-1]+x[n-2]+x[n-3]
     fec[1] <= shift[0]+shift[2]+shift[3]; //p1=x[n]+x[n-2]+x[n-3]
     fec[95:2]<=fec[93:0]; //left shift fec by 2 bits
     count<=count-1;
   end

  end
  assign done=(count==0);

endmodule
```

Testbench.v:

```
module t_FEC;
  reg clk;
  reg data;
  reg start;
  wire [95:0] fec;
  wire done;

  reg [47:0] input_data = 48'h03_01_02_03_30_3a; //Input appended with shift
register

  integer k;
  initial  forever #5 clk = ~clk ;

  FEC M1 (.clk(clk),.data(data),.start(start),.fec(fec),.done(done));

  initial begin
    clk=0;
    start=0;
    #15;
    start=1;
    #10 start=0;
    //#10;
   data=input_data[47];
    for(k=0;k<48;k=k+1)
      begin
        #10;
        input_data = input_data<<1;
        data=input_data[47];
      end
  end

  initial begin
    @(posedge done) begin
      $display("%h",fec);
      $finish;
    end
  end

  /*Debugging
  initial begin
    $monitor("%h %b",fec,done);
  end  */

endmodule
```

Waveforms:




```
VSIM 9> restart -f
# Loading work.t_FEC
# Loading work.FEC
VSIM 10> run -all
# 000e8c037c0df00e828c0e5e
# ** Note: $finish    : C:/Modeltech_pe_edu_10.4a/examples/t_FEC.v(43)
#    Time: 505 ns  Iteration: 2  Instance: /t_FEC
```