**Chapter -1: Verilog Refresh**

Q1.Design and verify configurable single port memory using Verilog?

Ans. module memory(addr,rd\_en,wr\_en,data,clk,cs,data\_o);

input [0:9] addr;

input wr\_en,rd\_en,clk,cs;

input [31:0] data;

reg [31:0] mem [0:1023];

output reg [31:0] data\_o;

always@(posedge clk)

begin

if(wr\_en && cs && ! rd\_en)

mem[addr] = data;

else if(cs && ! wr\_en && rd\_en)

data\_o = mem[addr];

else data\_o =32'hz;

end

endmodule

================testbench===================

// Code your testbench here

// or browse Examples

module tb();

reg wr\_en,rd\_en,clk,cs;

reg [0:9] addr;

reg [31:0] data;

wire[31:0] data\_o;

memory mem (addr,rd\_en,wr\_en,data,clk,cs,data\_o);

initial begin

clk=0;

forever #5 clk=~clk;

end

initial begin

#5;

cs=1;

addr = 2;

rd\_en=0;

wr\_en=1;

data = $random;

#5;

wr\_en=0;

rd\_en=1;

end

initial begin

$monitor("data\_o=%0h,addr=%0h,rd\_en=%0h,wr\_en=%0h,%0h",data\_o,addr,rd\_en,wr\_en,data);

#1000$finish;

end

endmodule

==========================output ============================

xcelium> run  
data\_o=x,addr=x,rd\_en=x,wr\_en=x,x  
data\_o=x,addr=2,rd\_en=0,wr\_en=1,12153524  
data\_o=x,addr=2,rd\_en=1,wr\_en=0,12153524  
data\_o=12153524,addr=2,rd\_en=1,wr\_en=0,12153524  
Simulation complete via $finish(1) at time 1 US + 0