MEMORY ACCESS :-

MAIN PROGRAM :-

module memory (

input clk,

input mem\_read, mem\_write,

input [18:0] address,

inout [18:0] data);

reg [18:0] mem [0:1023];

always @(posedge clk) begin

if (mem\_write) begin

mem[address] <= data;

end

end assign data = (mem\_read) ? mem[address] : 19'bz;

endmodule

TESTBENCH :-

`timescale 1ns / 1ps

module memory\_tb();

reg clk;

reg mem\_read, mem\_write;

reg [18:0] address, write\_data;

wire [18:0] data;

memory uut (

.clk(clk),

.mem\_read(mem\_read),

.mem\_write(mem\_write),

.address(address),

.data(data));

assign data = (mem\_write) ? write\_data : 19'bz;

initial clk = 0;

always #5 clk = ~clk;

initial begin

mem\_write = 1; address = 19'd0; write\_data = 19'd123;

#10 mem\_write = 0;

mem\_read = 1; address = 19'd0;

#10;

$display("Memory Read: address=%d, data=%d", address, data);

#10 $finish;

end

endmodule

OUTPUT :-

