Week 9

1. Write a SV program for Memory Access & Writeback Stage

Memory Access

a)

module memstage #(parameter N=32, M=1024) (

input logic [N-1:0] alu\_out, data\_in,

input logic mem\_read, mem\_write,

output logic [N-1:0] data\_out

);

logic [N-1:0] mem [0:M-1];

always\_comb

begin

if( mem\_read==1 && mem\_write==0)

data\_out= mem[alu\_out];

else if (mem\_read==0 && mem\_write==1) begin

mem[alu\_out]= data\_in;

data\_out= 32'b0;

end

else

data\_out= 32'b0;

end

endmodule

b)

module data\_mem #(parameter N=32) (

input logic [N-1:0] alu\_out, data\_in,

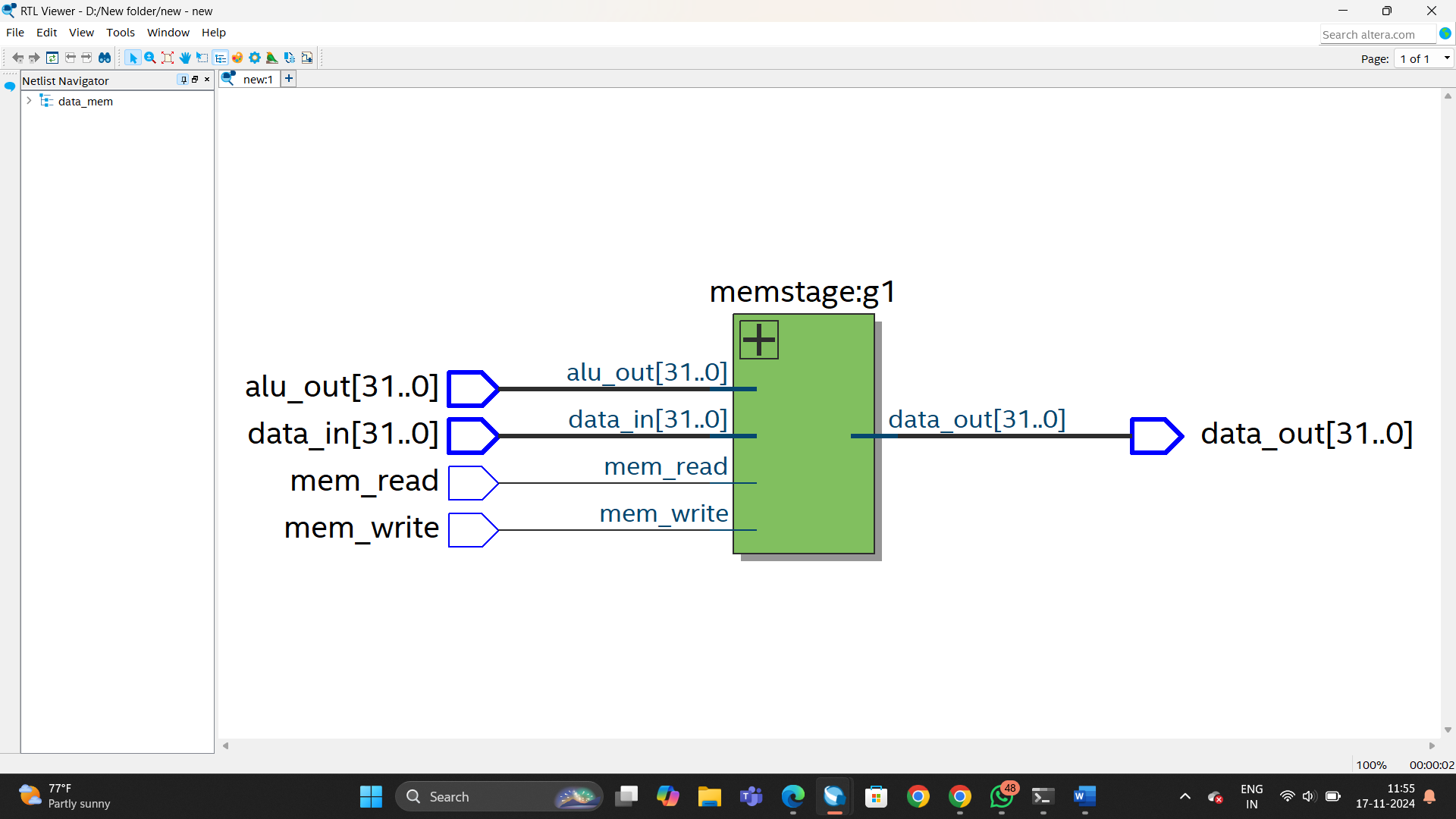
input logic mem\_read, mem\_write,

output logic [N-1:0] data\_out

);

memstage g1(alu\_out, data\_in,mem\_read, mem\_write,data\_out);

endmodule



Write Back

module write\_back1 #(parameter N=32)(

input logic [N-1:0] alu\_result,

input logic [N-1:0] data\_out,

input logic mem\_reg,

output logic [N-1:0] wr\_data

);

mux\_1 wb\_mux (alu\_result,data\_out,mem\_reg,wr\_data);

endmodule

