

include "processor\_defines.sv"

module alu\_core(

input logic [31:0] rs1\_val,

input logic [31:0] rs2\_val,

input logic [4:0] alu\_control,

output logic [31:0] rd\_write\_val

);

// Edit the code here begin ---------------------------------------------------

always @(\*) begin

case (alu\_control)

5'd1: rd\_write\_val = rs1\_val + rs2\_val;

5'd2: rd\_write\_val = rs1\_val - rs2\_val;

5'd3: rd\_write\_val = rs1\_val ^ rs2\_val;

5'd4: rd\_write\_val = rs1\_val | rs2\_val;

5'd5: rd\_write\_val = rs1\_val & rs2\_val;

5'd6: rd\_write\_val = rs1\_val << rs2\_val;

5'd7: rd\_write\_val = rs1\_val >> rs2\_val;

default rd\_write\_val = 32'b0;

endcase

end

// Edit the code here end -----------------------------------------------------

/\*

Following section is necessary for dumping waveforms. This is needed for debug and simulations

\*/

`ifndef SUBMODULE\_DISABLE\_WAVES\_ALU\_CORE

initial begin

$dumpfile("./sim\_build/alu\_core.vcd");

$dumpvars(0, alu\_core);

end

`endif

Endmodule

