module Mat\_mult(A,B,Res);

input [31:0] A;

input [31:0] B;

output [31:0] Res;

reg [31:0] Res;

reg [7:0] A1 [0:1][0:1];

reg [7:0] B1 [0:1][0:1];

reg [7:0] Res1 [0:1][0:1];

integer i,j,k;

always@ (A or B)

begin

{A1[0][0],A1[0][1],A1[1][0],A1[1][1]} = A;

{B1[0][0],B1[0][1],B1[1][0],B1[1][1]} = B;

i = 0; j = 0; k = 0;

$display("%d %d \n%d %d\n\n",A[31:24],A[23:16],A[15:8],A[7:0]);

$display("%d %d \n%d %d\n\n",B[31:24],B[23:16],B[15:8],B[7:0]);

{Res1[0][0],Res1[0][1],Res1[1][0],Res1[1][1]} = 32'd0;

for(i=0;i < 2;i=i+1)

for(j=0;j < 2;j=j+1)

for(k=0;k < 2;k=k+1)

Res1[i][j] = Res1[i][j] + (A1[i][k] \* B1[k][j]);

Res = {Res1[0][0],Res1[0][1],Res1[1][0],Res1[1][1]};

$display("%d %d \n%d %d\n\n",Res[31:24],Res[23:16],Res[15:8],Res[7:0]);

end

endmodule

module tb;

reg [31:0] A;

reg [31:0] B;

wire [31:0] Res;

Mat\_mult uut (.A(A),

.B(B),

.Res(Res)

);

initial

begin

A = 0; B = 0; #100;

A = {8'd1,8'd2,8'd3,8'd4};

B = {8'd5,8'd6,8'd7,8'd8};

end

endmodule