module mac(a,b,c,clk,asign,bsign,csign,m,sign);

input [15:0] a;

input [15:0] b;

input [31:0] c;

input clk;

input asign;

input bsign;

input csign;

output [32:0] m;

output sign;

wire [31:0]p;

wire isign;

wire y1,y2,y3,y4,y5,y6,y7,y8,y9,y10,y11,y12,y13,y14,y15,y16,y17,y18,y19,y20,y21,y22,y23;

wire y24,y25,y26,y27,y28,y29,y30,y31,y32;

sixteen\_sign ss(a,b,clk,asign,bsign,p,isign);

hal h1(p[0],c[0],m[0],y1);

fa f1(c[1],p[1],y1,m[1],y2);

fa f2(c[2],p[2],y2,m[2],y3);

fa f3(c[3],p[3],y3,m[3],y4);

fa f4(c[4],p[4],y4,m[4],y5);

fa f5(c[5],p[5],y5,m[5],y6);

fa f6(c[6],p[6],y6,m[6],y7);

fa f7(c[7],p[7],y7,m[7],y8);

fa f8(c[8],p[8],y8,m[8],y9);

fa f9(c[9],p[9],y9,m[10],y10);

fa f10(c[10],p[10],y10,m[11],y11);

fa f11(c[11],p[11],y11,m[12],y12);

fa f12(c[12],p[12],y12,m[13],y13);

fa f13(c[13],p[13],y13,m[14],y14);

fa f14(c[14],p[14],y14,m[15],y15);

fa f15(c[15],p[15],y15,m[16],y16);

fa f16(c[16],p[16],y16,m[17],y17);

fa f17(c[17],p[17],y17,m[18],y18);

fa f18(c[18],p[18],y18,m[19],y19);

fa f19(c[19],p[19],y19,m[20],y20);

fa f20(c[20],p[20],y20,m[21],y21);

fa f21(c[21],p[21],y21,m[22],y22);

fa f22(c[22],p[22],y22,m[23],y23);

fa f23(c[23],p[23],y23,m[24],y24);

fa f24(c[24],p[24],y24,m[25],y25);

fa f25(c[25],p[25],y25,m[26],y26);

fa f26(c[26],p[26],y26,m[27],y27);

fa f27(c[27],p[27],y27,m[28],y28);

fa f28(c[28],p[28],y28,m[29],y29);

fa f29(c[29],p[29],y29,m[30],y30);

fa f30(c[30],p[30],y30,m[31],y31);

fa f31(c[31],p[31],y31,m[32],y32);

xor (sign,csign,isign);

endmodule