```
module ALU #(parameter size = 16)  (A,B,ctrl,R,O,N,Z);   //Declaration of
default parameter size and port names
//Delclaring input ports
input [size-1 : 0]A, B;
input [1:0]ctrl;
//Delclaring output ports
output [size-1 : 0] R;
output O, N, Z;
//Delcaring registers
reg signed [size-1 : 0] R;
reg O,N,Z;
always @(*)    // * denotes a change in any variables
begin
case(ctrl)      //ctrl bit denotes the operation to be performed
      2'b00:
      begin
            R = A + B;       //Result
            N = R[size-1];         //Sign Flag
            O = (((~A[size-1]) & (~B[size-1]) & (R[size-1])) | ((A[size-
1]) & (B[size-1]) & (~R[size-1])));          // Overflow Flag
      end
      2'b01:
      begin
            R = A - B;
            N = R[size-1];
            O = (((~A[size-1]) & (B[size-1]) & (R[size-1])) | ((A[size-
1]) & (~B[size-1]) & (~R[size-1])));
      end
      2'b10:
      begin
            R = A & B;
      end
      2'b11:
      begin
            R = A | B;
      end
endcase;
if(R==0)         //Conditions for Zero Flag
begin
      O=1'b0;
      Z=1'b1;
end
else
      Z=1'b0;
end
endmodule
```