```
module ALU (
    input [3:0] A,          // 4-bit Input A
    input [3:0] B,          // 4-bit Input B
    input [2:0] Opcode,     // 3-bit Opcode to
select operation
    output reg [3:0] Result, // 4-bit Result of
operation
    output reg CarryOut     // Carry Out for
addition
);
    always @(*) begin
        CarryOut = 0; // Default CarryOut to 0
        case (Opcode)
            3'b000: begin
                {CarryOut, Result} = A + B; //
Addition with CarryOut
            end
            3'b001: begin
                Result = A - B; // Subtraction
            end
            3'b010: begin
```

```
                Result = A & B; // AND
            end
            3'b011: begin
                Result = A | B; // OR
            end
            3'b100: begin
                Result = ~A;    // NOT
            end
            default: begin
                Result = 4'b0000; // Default to 0
            end
        endcase
    end
endmodule
```