```
module alu (input [31:0] a, input [31:0] b, input [2:0] sel, output [31:0] out_data);
    reg [31:0] out;
    assign out_data = out;
    reg [32:0] carry;
    always @(*) begin
        case (sel)
            3'b000: out <= $signed(a) + $signed(b);
            3'b001: {carry, out} <= $unsigned(a) + $unsigned(b);
            3'b010: out <= $signed(a) - $signed(b);
            3'b011: out <= a - b;
            3'b100: out <= a & b;
            3'b101: out <= a | b;
            3'b110: out <= a >>> b;
            3'b111: out <= a >> b;
            default: out <= a + b;
        endcase
    end
endmodule
```