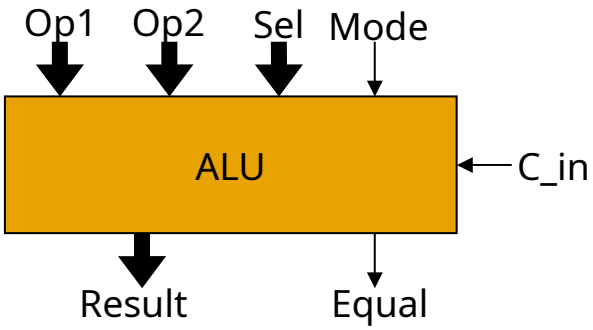
Exercise 1: SystemVerilog

1. ALU

a) DUT implementation

Implement the ALU as shown here:




```
package Types;
      typedef enum logic [1:0] {
                  ADD      = 2'b00,
                  SUBTRACT = 2'b01,
                  MULTIPLY = 2'b10
            } sel_t;

      typedef enum logic {
                  WITH_CARRY = 1'b1,
                  NO_CARRY = 1'b0
            } mode_t;
endpackage : Types

module ALU(
      output logic [7:0] Result,
      output logic Equal,
      input logic [3:0] Op1,
      input logic [3:0] Op2,
      input sel_t Sel,
      input logic C_In,
      input mode_t Mode);
⋮
// your code goes here
⋮
endmodule : ALU
```

The inputs and outputs are defined as follows:

- Result: The result of the operation performed.
- Equal: High, if both operands are equal.
- C_in: Carry-in
- Op1: The first operand.
- Op2: The second operand.
- Sel: The operation to perform.
  - ADD: Op1 + Op2
  - SUBTRACT: Op1 – Op2
  - MULTIPLY: Op1 × Op2

    - others: *undefined*
- Mode: Enable or disable the carry input.

b) Testing

- Write a testbench to verify that the ALU performs as expected.

c) Clocking

- Add a Clock input to the ALU and place the result in a flip-flop.
- Adjust the testbench accordingly. Observe the delay between input and output changes.
  Hint: Use @posedge (Clock); in your testbench code to wait for the next clock edge:
  ```
  initial begin
   @posedge (Clock); // can go where #delay used to be
   // do something
  end
  ```

2. SystemVerilog: Up-/Down-Counter

Implement a clocked counter that can count either up or down, and has

1. a sync. RESET
2. a sync. LOAD function (loading from DATA into the COUNT register).

Note: RESET shall take precedence over LOAD.

```
module UpDownCounter(
      input logic CLK,
      input logic RESET,
      input logic LOAD,
      input logic [5:0] DATA,
      input logic COUNT_UP, // == 1: count up, == 0: count down
      output logic [5:0] COUNT
);
⋮
// your code goes here
⋮
endmodule : UpDownCounter
```

b) Testing

- Write a testbench exercising all modes of the counter.