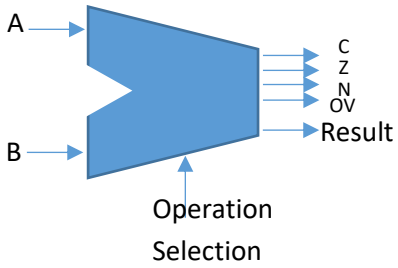
Marmara University Electrical-Electronics Eng. Dept.

CSE4117 Microprocessors Laboratory Assignment #1

Each group must submit their own study. The group members have to work together and each member must know each step of the study. The studies will be submitted until 03.08.2022. A report (pdf file) should explain the digital circuits, operational blocks and all necessary technical information (You may use screenshots in your report). A short video file (<2 mins and <10 MByte)must be prepared to present the study. Each group must upload these files to a cloud-drive. The drive link must be sent to cse4117@yaani.com email address.

For this laboratory assignment, you will use Altera Quartus II software and the simulation tool to design and demonstrate an ALU that meets the following requirements.



You are required to design an 8-bit ALU with two input values A&B, and a selection input. The result and several flags(Carry, Zero, Negative and Overflow) are the output of the ALU block.

The ALU should have the following operations with the given definitions:

1. ADDA: adds <input A> to <input B>, affected flags: Z,C,N,OV
   Result = A+B
2. SUBZ: substracts <input B> from <input A>, affected flags: Z,C,N,OV
   Result = A-B
3. ANDA: Bitwise AND operation, ANDs <input A> with <input B> bit by bit, affected flags: Z,N
4. Result = A&B
5. ORA: Bitwise OR operation, ORs <input A> with <input B> bit by bit, affected flags: Z,N
   Result = A|B
6. MOVA: moves <input A> to the result, affected flags: Z,N
   Result = A
7. SHFTLA: Shifts <input A> to the left and moves it to the result , affected flags: Z,N
   Result = A<<1
8. SHFTRA: Shifts <input A> to the right and moves it to the result , affected flags: Z,N
   Result = A>>1
9. INCR: Increases <input B> by one and moves it to the result. affected flags: Z,C,N,OV
   Result = B+1
10. DECR: Decreases <input B> by one and moves it to the result. affected flags: Z,C,N,OV
    Result = B-1

The Flag definitions:

Z: Zero Flag is set when the result bits are all zero.

C: Carry flag is set when the add or sub operation generates a carry out.

N: Negative flag is set when the result is a negative number.

OV: Overflow flag indicates an overflow

You are required to demonstrate that your design is fully functional.

YOUR DESIGN MUST HAVE THE VERILOG-HDL CODE THAT HAS GATE LEVEL PROGRAMMING.

Marmara University Electrical-Electronics Eng. Dept.

CSE4117 Microprocessors Laboratory Assignment #1

Each group must submit their own study. The group members have to work together and each member must know each step of the study. The studies will be submitted until 03.08.2022. A report (pdf file) should explain the digital circuits, operational blocks and all necessary technical information (You may use screenshots in your report). A short video file (<2 mins and <10 MByte)must be prepared to present the study. Each group must upload these files to a cloud-drive. The drive link must be sent to cse4117@yaani.com email address.

Some examples:

```
module full_adder(
   input x,
   input y,
   input cin,
   output o,
   output cout
);
   wire t1, t2, t3;
   xor(t1, x, y);
   xor(o, t1, cin);
   and(t2, x, y);
   and(t3, t1, cin);
   or(cout, t2, t3);
endmodule

module full_adder_operators(
   input x,
   input y,
   input cin,
   output o,
   output cout
);
   assign o = (~x & ~y & cin) | (~x & y & ~cin) | (x & y & cin) | (x & ~y & ~cin);
   assign cout = (x & y) | (y & cin) | (x & cin);
endmodule


module eight_bit_adder(
   input [7:0] x,
   input [7:0] y,
   output [7:0] o,
   output carryout
);
   wire cout1, cout2, cout3, cout4, cout5, cout6, cout7;
   one_bit_adder(x[0], y[0], 1'b0, o[0], cout1);
   one_bit_adder(x[1], y[1], cout1, o[1], cout2);
   one_bit_adder(x[2], y[2], cout2, o[2], cout3);
   one_bit_adder(x[3], y[3], cout3, o[3], cout4);
   one_bit_adder(x[4], y[4], cout4, o[4], cout5);
   one_bit_adder(x[5], y[5], cout5, o[5], cout6);
   one_bit_adder(x[6], y[6], cout6, o[6], cout7);
   one_bit_adder(x[7], y[7], cout7, o[7], carryout);
endmodule
```

Marmara University Electrical-Electronics Eng. Dept.

CSE4117 Microprocessors Laboratory Assignment #1

Each group must submit their own study. The group members have to work together and each member must know each step of the study. The studies will be submitted until 03.08.2022. A report (pdf file) should explain the digital circuits, operational blocks and all necessary technical information (You may use screenshots in your report). A short video file (<2 mins and <10 MByte)must be prepared to present the study. Each group must upload these files to a cloud-drive. The drive link must be sent to cse4117@yaani.com email address.

```
module two_to_one_multiplexer(
   input s,
   input x,
   input y,
   output o
);
   wire t1, t2;
   and(t1, s, y);
   and(t2, ~s, x);
   or(o, t1, t2);
endmodule

module multiplexer( //Four to one
   input [1:0] s,
   input [3:0] x,
   output o
);
   wire t1, t2;
   two_to_one_multiplexer(s[0], x[0], x[1], t1);
   two_to_one_multiplexer(s[0], x[2], x[3], t2);
two_to_one_multiplexer(s[1], t1, t2, o);
endmodule
```