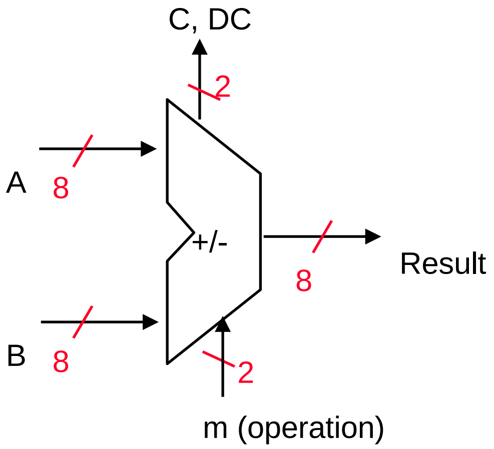
University of Jordan
Computer Engineering Department
CPE439: Computer Design Lab

Experiment 2: 8-Bit Adder/Subtractor

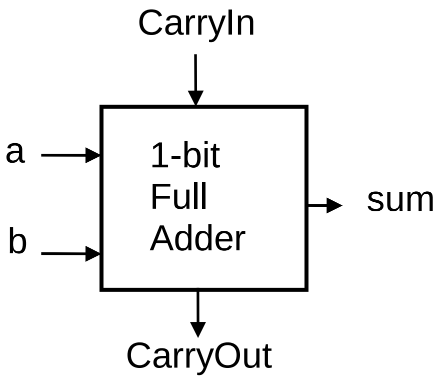
It is required to construct and test a Verilog module for an 8-bit adder/subtractor suitable for incorporation in your PIC16F84A design. You should use modular design where you start by building and testing low-level modules using the library modules defined in **Lib439.v**, then use the low-level modules in larger modules. The symbol and function of this adder/subtractor are shown below. Where “C” is the Carry/Borrow’ bit and “DC” is the Digit Carry/Borrow’ bit.



m (operation)	Function
00	B + A Add
01	B + 1 Increment
10	B – A Subtract
11	B – 1 Decrement

Full Adder

We suggest that you start by building a Verilog module for a full adder that has the following symbol and truth table:



a	b	CarryIn	CarryOut	sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

The Verilog module for this adder should be similar to the following code:

```
module FullAdder(sum, CarryOut, CarryIn, a, b);
  output sum, CarryOut;
  input CarryIn, a, b;
  // implementation details are left to the student
  …
endmodule
```

Multiplexer

Then build and test a 2-to-1 multiplexer module that has Verilog code similar to the following code:

```
module Mux_2_to_1(Result, s, i0, i1);
  output Result;
  input s;
  input i0, i1;
  // implementation details are left to the student
  …
endmodule
```

8-Bit Adder/Subtractor

Finally use eight full-bit adder modules and eight 2-to-1 multiplexer modules to construct your 8-bit adder/subtractor. Note that you need also to add xor gates to implement the subtraction. Your module should have Verilog code similar to the following code:

```
module AddSub_8b(Result, C, DC, A, B, m);
  output [7:0] Result;
  output C, DC;
  input [7:0] A, B;
  input [1:0] m;
  // implementation details are left to the student
  …
endmodule
```

Report

Your report should include detailed design, Verilog code for all modules including your test modules, and timing diagram that demonstrates the correct operation of your design given the input shown in the following table.

A	B	m
0000 1111	1111 0000	00
0101 1001	0011 1001	00
0101 1001	0111 1111	01
0000 1111	1111 0000	10
0101 1001	0011 1001	10
0101 1001	1000 0000	11

Bonus marks will be given to the design that uses carry-look ahead scheme. Also estimate the maximum delay expected for you design.