Tyler Kim

DLD

***ALU Verification Report***

**Component**

**Diagram, schematic

Description automatically generated**

**Figure 1: ALU Schematic**

The purpose of this learning activity is to create an arithmetic and logic unit (alu) for the processor. The alu takes in four inputs: *p*, *q*, *f0*, and *f1* and returns two outputs: *alu\_out* and *Z* by performing arithmetic operations on *p* and *q*. The alu can do addition, subtraction, increment on *p*, or decrementon *p* on two input bytes. Figure 1 highlights the schematic of the alu. The alu is made with two sub components, the full adder that adds two bits and an adder\_subtractor that performs the operations.

Table

Description automatically generated

**Figure 2: Input Operations**

The alu determines which operation to perform using the two inputs: *f0* and *f1*. The input, *f1* determines if the operation is addition/subtraction or increment/decrement and the input, *f0* determines if the opposite of the operation is desired.

**Verification Tests**

By the nature of this component, testing was done by doing one example of each operation and seeing if the alu works correctly. In addition, “edge cases” were tested as well to ensure that alu works optimally.

Graphical user interface

Description automatically generated

**Figure 3: Verification Tests**

Text

Description automatically generated

**Figure 4: Verification Tests with Assertion Statements**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **p** | **q** | **Operation** | **English** | **output** | **Expected output** |
| 00000010 | 00000111 | Addition | 2 + 5 = 7 | 00000111  (Z = 0) | 00000111  (Z = 0) |
| 00000111 | 00000100 | Subtraction | 7 – 4 = 3 | 00000011  (Z = 0) | 00000011  (Z = 0) |
| 11111111 | 00000000 | Increment | 255 + 1 | 00000001  (Z = 0) | 00000001  (Z = 0) |
| 00000000 | 00000000 | Decrement | 0 – 1 | 11111111  (Z = 0) | 11111111  (Z = 0) |
| 00000001 | 00000001 | Subtraction | 1 – 1 = 0 | 00000000  (Z = 1) | 00000000  (Z = 1) |

**Figure 5: Truth Table for Given Examples**

Figure 3 – 5 illustrate the verification process for testing the alu. Notice in Figure 5 that when the operation has to pass the max or min value, the alu “wraps around” to the other extreme value. This is expected from the alu. The test results indicate that the schematic passed all tests according tot eh verification tests and assertion statements. Therefore, the alu schematic is correct and successfully passed all the tests.