**Lab 2 Prelab**

1. SELF-STUDY QUESTIONS

*Q1. What unsigned integer value does the binary value 1111 10012 represent?*

A1: 24910

*Q2. What is the largest unsigned integer value that can be represented by an 8-bit binary value?*

A2:

*Q3. What range of signed integer values can be represented by 8-bit binary values under 2’s complement interpretation?*

A3:

*Q4. What is the 2’s complement of 1111 10012 ?*

A4: 0000 01112

*Q5. What signed integer value is represented by 1111 10012 ?*

A5: -7

*Q6. Why doesn’t taking the 2’s complement of X = 1000 00002 result in the value representing -X?*

A6: There can only be a total of integers that can be represented by a set of n bits in binary. Two’s complement solves the problem of having two “zeroes”.

*Q7. What is the 8-bit signed integer result of adding X = 1111 11112 + Y = 0000 00012 ?*

A7: 0000 00002 with a carry out, 010

*Q8. Show an addition of two unsigned integers represented using 8-bit values that results in a carry out of the MSB. Explain what value the 9-bit answer represents.*

A8: (ANSWER UNKNOWN)

*Q9. Use your answer in Q7 to argue that a carry out of the MSB when adding 8-bit values representing signed integers does not necessarily mean that the answer cannot be represented in 8-bit answer.*

A9: (ANSWER UNKNOWN)

*Q10. What is the 8-bit binary result of adding X = 0111 11112 + Y = 0000 00012 ? What are the signed integer values represented by X, Y, and the sum?*

A10: 1000 00002 = -25610. X represents 25510 and Y represents 110. The actual result is 25610, however 256 is out of the 8-bit signed integer range. Therefore, an overflow occurs.

*Q11. Suppose that Z = X + Y and that X, Y, and Z are signed integers. Describe a simple test that can be performed on binary values representing X, Y, and Z to decide if overflow occurs when doing the addition.*

A11: Checking if Z is out of the range in decimal allows one to determine if an overflow will occur. It is also possible to pass the carry out of the 8th and 7th bits through an XOR gate, where the result will represent whether an overflow occurred.

1. SELF STUDY QUESTIONS

*Q. Why can a single binary value represent both a signed integer value and an unsigned integer value? Use a representative example to help your explanation.*

A. In signed integers, the MSB is used to denote whether the number is positive or negative and the rest of the bits are used to represent the magnitude of the integer. In unsigned integers, all of the bits are used to represent the magnitude of the integer.

*Q. For each component supplied in Logisim’s Arithmetic Library, give a simple one-sentence summary (in your own words) of the function performed by the component.*

A.

Adder: works like n cascaded full adders, where n is the bit width of the inputs.

Subtractor: works like an adder (see adder), except the subtrahend is two’s complemented

Multiplier: multiplies numbers

Divider: divides numbers

Negator: performs 2s complement

Comparator: compares the magnitude of inputs

Shifter: shifts bits

Bit Adder: adds bits

Bit Finder: finds a bit

*Q. Why are separate UMIN and SMIN operations included? Give an example in which a pair of input values (X and Y) would result in different output values (R) for the functions.*

A.

Example values: 10012 and 00002

UMIN:

Compares 910  and 010

1001 is larger than 0000

SMIN:

Compares -7 and 0

0000 is larger than 1001

*Q. For each operation, propose test cases that can be used to test the ALU to demonstrate that it complies with the requirements of the operation given above. Each test case should expose different important aspects of the required functionality. There should be enough test cases to expose all important aspects of the required functionality. Describe each test in terms of any initial conditions, the inputs, the expected outputs, the particular aspects of the requirements that are tested, and why the test validates the aspects of the requirements being tested.*

A.

Cases:

* Result overflows on the positive end (+/+)
* Result overflows on the negative end (-/-)