Takahiro Shimokobe

CSC 225

Professor Planck

**Assignment 1**

**Unsigned 8-bit arithmetic:**

00000000: 0 00101100: 44

11111111: 256 00010000: 16

10000000: 128 00010001: 17

Min: 00000000 0

Max: 11111111 256

**Signed 2’s Compliment 8 bit:**

00000000: 0 00011100: 28

11111111: -1 00010101: 21

10000000: -128 10111001: -1

Min: 10000000: -128

Max: 01111111: 127

**Carry and Overflow:**

8 bit unsigned number that results in a carry:

Binary: decimal:

10000000 128

+ 10000000 128

100000000 256

Bit signed number that produces an overflow but not a carry

Binary: decimal:

01000000 64

+ 01000000 64

10000000 -128

**Hexadecimal:**

00000000: x00 00001011: xB

11111111: xFF 00010101: x15

10001000: x88 11010110: xD6

x24A3 + x5678 : x7B1B xABCD + x1EEE : xCABB

x94AA – x5678: x3E32 xFBCD – xEEEE: xCDF

**Bitwise Logic Operations:**

x24A3 & x5087 = x83 xABCD | x1EEE = xBFEF

Determine the operation and value needed to set bit 5 of the value to zero in a single operation.

AND x945A

Number two:

ADD x10

**Using the LC-3 Simulator:**

1. The Program Counter Register is set to x3000 by default.
2. The content of memory location x0023 in binary is 0000010010100000
3. The content of memory location xFFFE in hex is x7FFF
4. The opcode for the instruction at memory location x0590 is LD. # CHECK THIS WITH PROFESSOR

**Making the LC-3 do something:**

1. The hexadecimal value of the data in R0 is x0064
2. The decimal value of the data in R1 is 35.
3. The hexadecimal value of the data in R2 is x0087.

**Let’s do more complicated things with ADD, AND, and NOT (1pt):**

**Copy R0 to R1 (1 instruction):**

**x1220**

**Clear R0 to zero:**

**x5020**

**Clear bit 2 of R0 but leave the rest of the register unchanged:**

**x503B**