Austin Ngo

Professor Atanasio

CSCI 150

03/03/2022

CSCI 150 – Lab 1

1. What is the range of values of an unsigned integer that is stored in 16 bits? Please use range notation as was used in class and use decimal values for your answer.

For unsigned integers we use [ 0, 2^n-1], where ‘n’ would be the number of bits. Therefore, the range of values for an unsigned integer stored in 16 bits would be

[ 0, 2^(16-1) ] , which is **[ 0, 32768].**  
   
2) What is the range of values of a signed integer that is stored in 16 bits? Please use range notation as was used in class and use decimal values for your answer.

For signed integer values we use [ -2^(n-1), 2^(n-1) – 1 ] , therefore the range of a signed integer that is stored in 16 bits would be **[ -32768, -32767 ].**

3) Convert the following unsigned values to binary so that it can be stored in a 16-bit memory location. Be sure to include all 16 bits in your answers.   
a. 568 – to binary: **0 0 0 0 | 0 0 1 0 | 0 0 1 1 | 1 0 0 0**

568 – 512 = 56

56 – 32 = 24

24 – 16 = 8

b. 248 – to binary: **0 0 0 0 | 0 0 0 0 | 1 1 1 1 | 1 0 0 0**

248 – 128 = 120

120 – 64 = 56

56 – 32 = 24

24 – 16 = 8

c. 1306 – to binary: **0 0 0 0 | 0 1 0 1 | 0 0 0 1 | 1 0 1 0**

1306 – 1024 = 282

282 – 256 = 26

26 – 16 = 10

10 – 8 = 2

d. 2005 – to binary: **0 0 0 0 | 0 1 1 1 | 1 1 0 1 | 0 1 0 1**

2005 – 1024 = 981

981 – 512 = 469

469 – 256 = 213

213 – 128 = 85

85 – 64 = 21

21 – 16 = 5

5 – 4 = 1

4) Convert the following signed values to binary using 2’s complement so that it can be stored in a 16-bit memory location. Be sure to include all 16 bits in your answers.   
a. -1099 – to binary: **1 1 1 1 | 1 0 1 1 | 1 0 1 1 | 0 1 0 1**

1099 – 1024 = 75

75 – 64 = 11

11 – 8 = 3

3 – 2 = 1

0 0 0 0 | 0 1 0 0 | 0 1 0 0 | 1 0 1 1 – positive binary value

1 1 1 1 | 1 0 1 1 | 1 0 1 1 | 0 1 0 0 - one’s compliment

1 1 1 1 | 1 0 1 1 | 1 0 1 1 | 0 1 0 1 – add one for two’s compliment

b. -58 – to binary: **1 1 1 1 | 1 1 1 1 | 1 1 0 0 | 0 1 1 0**

58 – 32 = 26

26 – 16 = 10

10 – 8 = 2

0 0 0 0 | 0 0 0 0 | 0 0 1 1 | 1 0 1 0 – positive binary value

1 1 1 1 | 1 1 1 1 | 1 1 0 0 | 0 1 0 1 – one’s compliment

1 1 1 1 | 1 1 1 1 | 1 1 0 0 | 0 1 1 0 – add one for two’s compliment

c. -4558 to binary: **1 1 1 0 | 1 1 1 0 | 0 0 1 1 | 0 0 1 0**

4558 – 4096 = 462

462 – 256 = 206

206 – 128 = 78

78 – 64 = 14

14 – 8 = 6

6 – 4 = 2

0 0 0 1 | 0 0 0 1 | 1 1 0 0 | 1 1 1 0 – positive binary value

1 1 1 0 | 1 1 1 0 | 0 0 1 1 | 0 0 0 1 – one’s compliment

1 1 1 0 | 1 1 1 0 | 0 0 1 1 | 0 0 1 0 – add one for two’s compliment

5) Perform binary addition operations on the following 8-bit signed integer values. If there is an overflow, then state it and explain. You must show your work.   
**a. 128 + (-58)**

1 0 0 0 | 0 0 0 0 (128)

1 1 0 0 | 0 1 1 0 (-58)

**0 1 0 0 | 0 1 1 0 (70) : There is no overflow, because we added a negative number to a positive number.**

58 – 32 = 26

26 – 16 = 10

10 – 8 = 2

0 0 1 1 | 1 0 1 0 (58)

1 1 0 0 | 0 1 0 1 (58 flipped to one’s compliment)

1 1 0 0 | 0 1 1 0 (-58, added one for two’s compliment)

**b. 120 + 8**

0 1 1 1 | 1 0 0 0 (120)

0 0 0 0 | 1 0 0 0 (8)

**1 0 0 0 | 0 0 0 0 (128)**: **There is an overflow, because we added 120 + 8 = 128, which cannot be represented only 8-bit signed integer values. The maximum number which can be represented by an 8-bit signed integer value is 127.**

**c. 115 + 64**

0 1 1 1 | 0 0 1 1 (115)

0 1 0 0 | 0 0 0 0 (64)

**1 0 1 1 | 0 0 1 1 (-51)**: **There is an overflow because we added 115 + 64 = 179, which is higher than the maximum number which can be represented only using 8-bit signed integer values. The maximum number which can be represented by an 8-bit signed integer value is 127.**

6) Using truth tables, prove the following equivalence rules   
**a. ¬ (¬A) ≡ A**

**A | ¬ A | ¬ (¬ A)**

**F | T | F |**

**T | F | T |**

b. ¬ (A ^ B) ≡ ¬A v ¬B

**A | B | A ^ B | ¬ (A ^ B)**

**F | F | F | T**

**F | T | F | F**

**T | F | F | F**

**T | T | T | F**

c. A ^ (B v Z) ≡ (A ^ B) v (A ^ Z)

**A | B | Z | (A ^ B) v (A ^ Z)**

**F | F | F | F**

**F | F | T | F**

**F | T | F | F**

**F | T | T | F**

**T | F | F | F**

**T | F | T | T**

**T | T | F | T**

**T | T | T | T**