COMP 3350 – 001 Project #1

Haegang Yang

1. (9 points) Convert the following unsigned base 2 numbers (binary) to base 16 numbers (hexadecimal):

A. 0110 0001 1111

=> 0110/0001/1111

=> 0110 = 2^2 + 2 = 6

=> 0001 = 2^0 = 1

=> 1111 = 2^3 + 2^2 + 2 + 2^0 = 15 = F

=> 61F

B. 1000 1111 1100

=> 1000/1111/1100

=> 1000 = 2^3 = 8

=> 1111 = 2^3 + 2^2 + 2 + 2^0 = 15 = F

=> 1100 = 2^3 + 2^2 = 12 = C

=> 8FC

C. 0001 0110 0100 0101

=> 0001/0110/0100/0101

=> 0001 = 2^0 = 1

=> 0110 = 2^2 + 2 = 6

=> 0100 = 2^2 = 4

=> 0101 = 2^2 + 2^0 = 5

=> 1645

2. (27 points) Convert the following binary numbers to base 10 numbers (decimal):

Each time if binary numbers are represented in:

a) Signed magnitude representation.

b) One's complement representation.

c) Two's complement representation.

A. 1100 1010

   a) $-(2^6 + 2^3 + 2) = -74$

   b) Except MSB (which is 1 here) change 1 to 0 and change 0 to 1

      10110101 = signed magnitude value

      $-(2^5 + 2^4 + 2^2 + 2^0) = -53$

   c) Make it 1's complement first

      1100 1001

      Make it signed magnitude value

      1011 0110 => $-(2^5 + 2^4 + 2^2 + 2^1) = -54$

B. 1111 0010

   a) $-(2^6 + 2^5 + 2^4 + 2)$

      $= -114$

   b) change 1 to 0 and change 0 to 1

      10001101 = signed magnitude value

      $-(2^3 + 2^2 + 2^0) = -13$

   c) Make it 1's complement first

      1111 0001

Make it signed magnitude value

1000 1110=> $-(2^3 + 2^2 + 2^1) = -14$

C. 1000 0111

a) $-(2^2 + 2 + 2^0)$

$= -(4 + 2 + 1)$

$= -7$

b) **change 1 to 0 and change 0 to 1**

1111 1000 = signed magnitude value

$= -(2^6 + 2^5 + 2^4 + 2^3)$

$= -(64 + 32 + 16 + 8)$

$= -120$

c) **Make it 1's complement first**

1000 0110

Make it signed magnitude value

1111 1001=> $-(2^6 + 2^5 + 2^4 + 2^3 + 2^0) = -121$

3. (36 points) Convert the following base 10 (decimal) values to binary numbers (8-bits):

Each binary result represented in:

a) Signed magnitude representation.

b) One's complement representation.

c) Two's complement representation.

A. -100_d

a) 1110 0100

$100 = 2^6 + 2^5 + 2^2$

0110 0100 (in binary)

8bit and negative sign => MSB must be 1

1110 0100

b) 1001 1011

Except MSB (which is 1 here) change 1 to 0 and change 0 to 1 from Signed magnitude representation

c) 1001 1100

add 1 to LSB of 1's complement

B. -16_d

a) 1001 0000

$16 = 2^4$

00010000 (in binary)

8bit and negative sign => MSB must be 1

1001 0000

b) 11101111

Except MSB (which is 1 here) change 1 to 0 and change 0 to 1 from Signed magnitude representation

c) 11110000

add 1 to LSB of 1's complement

C. -21_d

a) 10010101

$21 = 2^4 + 2^2 + 2^0$

00010101 (in binary)

8bit and negative sign => MSB must be 1

10010101

b)11101010

Except MSB (which is 1 here) change 1 to 0 and change 0 to 1 from Signed magnitude representation

c)11101011

add 1 to LSB of 1's complement

D. -0_d

a)10000000

Even though it is 0, it still has the sign so the MSB must be 1 which turns out 1000 0000

b)11111111

Except MSB (which is 1 here) change 1 to 0 and change 0 to 1 from Signed magnitude representation

c) does not exist

add 1 to LSB of 1's complement but it makes 100000000 which is more than 8 bits.

There is no representation of two's complement for negative zero.

4. (4 points) What is the range of:

A. An unsigned 7-bit number?

An unsigned n bit shows number from $0 \sim 2^n - 1$.

Because there is no bit for showing the sign.

Therefore, an unsigned 7 bit number's range is $0 \sim 2^7 - 1$

B. A signed 7-bit number?

In a signed bit, one bit has to show the sign.

Therefore, it shows $-2^{(n-1)} \sim (2^{(n-1)} - 1)$.

In 7 bit, its range is $-2^6 \sim (2^6 - 1)$.

5. (12 points) Provide the answer to the following problems (∧ = AND, ∨ = OR)

1. 1000 ∧ 1110

   = 1000 * 1110

   = 1110000

2. 1000∨1110

   = 1000 + 1110

   = 10110

3. (1000∧1110) ∨ (1001∧1110)

   = (1000 * 1110) + (1001 * 1110)

   = 1110000 + 1111110

   = 11101110

6. (9 points) Please demonstrate each step in the calculation of the arithmetic operation 25 - 65. (both 25 and 65 are signed decimal numbers)

25 = $2^4 + 2^3 + 2^0$ = 11001

65 = $2^6 + 2^0$ = 1000001

Let's make them 8 bit signed magnitude number.

25 is a positive number so the MSB is 0 => 00011001

-65 is a negative number so the MSB is 1 => 11000001

Now get the 1's complements.

25 is a positive number so the 1's complements remain same.

- ⇨ **00011001**

-65 => 11000001 => 10111110

Now get the 2's complement for each number

25 is a positive number so the 2's complements remain same.

=> 00011001

-65 => 11000001 => 10111110 => 10111111

Therefore 25 – 65 is 11001 + 10111111 in binary.

11001 + 10111111 = 11011000

It is not overflowed so make 11011000 2’s complement again.

11011000 => 101000.

The answer is 101000

7. (3 points) Mathematically the answer in Q6 is -40_d. Please verify your answer in Q6 using a conversion of 2’s and decimal numbers.

Change the -40 into the signed magnitude number

$40 = 2^5 + 2^3$ = 101000 => 1101000

Change the signed magnitude number into 1’s complement

1101000 => 1010111

Change the 1’s complement into 2’s complement

1010111 => 1011000

As we did 2’s complement again, make the 1011000 2’s complement again

101000 which is the same answer with 101000

The answer is correct.