Maneesh Wijewardhana (1125828) 2021/09/11

TA: Muhammad Rehman CIS* 2030

Assignment #1

1.

Binary	Decimal	Octal	Hexadecimal
1011011.101	91.625	133.5	5B.A
1100010	98	142	62
11101110	238	356	EE
10101111.11	175.75	257.6	AF.C

$1011011.101_2 \rightarrow 133.5_8 \rightarrow 5B.A_{16}$

$2^6\,2^5\,2^4\,2^3\,2^2\,2^1\,2^0 .\,2^{-1}\,2^{-2}\,2^{-3}$

$64+16+8+2+1.0.5+0.125$

$= 91.625_{10}$

$356_8 \rightarrow 2^2 2^1 2^0$

$011 + 101 + 110$

$= 11101110_2 \rightarrow 128+64+32+8+4+2 = 238_{10} \rightarrow EE_{16}$

$98_{10} \rightarrow 2^6\,2^5\,2^4\,2^3\,2^2\,2^1\,2^0$

$1100010_2 \rightarrow 142_8 \rightarrow 62_{16}$

$98 - 64 = 34$

$34 - 32 = 2$

10 15 12

$AF.C_{16} \rightarrow 2^3 2^2 2^1 2^0$

$1010 + 1111 + .11$

$= 10101111.11_{10} \rightarrow 128+32+8+4+2+1+.0.5+0.25 = 175.75_{10} \rightarrow 257.6_8$

2.
$$\begin{array}{r} 101101 \\ +110101 \\ \hline 1100010 \end{array} \rightarrow \begin{array}{r} 45 \\ +53 \\ \hline 98 \end{array}$$

Since these numbers are unsigned, their range is 0 to $2^k - 1$. In a 6 bit location, our range is 0–63. Since the decimal answer of 98 is out of that range, overflow occurs.

3.
$$\begin{array}{r} 1001101 \\ -10111 \\ \hline 110110 \end{array} \rightarrow \begin{array}{r} 77 \\ -23 \\ \hline 54 \end{array}$$

32 16 8 4 2 1

4.

Decimal	Sign-Magnitude	2's Complement
+14	001110	001110
-20	110100	101100
37	0100101 (overflow)	0100101 (overflow)

+14: $001110_{\text{sign-mag}}$ (+) → Since positive, 2's-comp is same as sign-mag

−20: 110100 sign-mag (−)

−20: $N = 20$, $n = 6$ → $2^n - N = 2^6 - 20 = 44_{10} \rightarrow 101100_{\text{2's-comp}}$

37: $0\,100101_{\text{sign-mag}}$ (+) → since positive, 2's-comp is same as sign-mag

↳ One difficulty that arises is the fact that in order to represent 37_{10} in binary, we have to use more than 6 bits which causes overflow. Also, the range for signed numbers with 6 bits is $-(2^{k-1}-1)$ to $+(2^{k-1}-1)$ which is −31 to 31. 37 is out of that range. Same thing with 2's-comp where the range is $-(2^{k-1})$ to $+(2^{k-1}-1)$ which is −32 to 31. 37 is also not in that range so overflow occurs.

5. 21 bits → 2's-comp: $-(2^{k-1})$ to $+(2^{k-1}-1)$

$= -(2^{20})$ to $+(2^{20}-1)$

$= -1048576$ to $+1048575$

↳ unsigned: 0 to $2^k - 1$

$= 0$ to $2^{21} - 1$

$= 0$ to 2097151

6. $A = 9_{10}$ $\quad B = 19_{10}$

assume 6-bits

A+B

$+9: 001001_2$ (+)

$+19: 010011_2$ (+)

2's-comp is same as unsigned so...

$$\begin{array}{r} 001001 \\ +\ 010011 \\ \hline 011100 \end{array}$$

A−B

$+9: 001001_2$ (+) 2's-comp is same

$-19: 010011_2 \xrightarrow{\text{flip}} 101100 \xrightarrow{\text{add 1}}$

$$\begin{array}{r} 101100 \\ +\ 1 \\ \hline 101101 \end{array} \text{ 2's-comp}$$

$$\begin{array}{r} 001001 \\ +\ 101101 \\ \hline 110110 \end{array}$$

(−)

B−A

$-9: 001001_2 \xrightarrow{\text{flip}} 110110 \xrightarrow{\text{add 1}}$

$$\begin{array}{r} 110110 \\ +\ 1 \\ \hline 110111 \end{array} \text{ 2's-comp}$$

$+19: 010011_2$ (+) 2's-comp is same

$$\begin{array}{r} 010011 \\ +\ 110111 \\ \hline 001010 \end{array}$$

(+)

We discard the final carry here

−A−B

$-9: 001001_2 \xrightarrow{\text{flip}} 110110 \xrightarrow{\text{add 1}}$

$$\begin{array}{r} 110110 \\ +\ 1 \\ \hline 110111 \end{array} \text{ 2's-comp}$$

$-19: 010011_2 \xrightarrow{\text{flip}} 101100 \xrightarrow{\text{add 1}}$

$$\begin{array}{r} 101100 \\ +\ 1 \\ \hline 101101 \end{array} \text{ 2's-comp}$$

$$\begin{array}{r} 110111 \\ +\ 101101 \\ \hline 100100 \end{array}$$

We discard the final carry here

7. $A = 32_{10}$, $B = 31_{10}$

For 6-bit 2's-comp, the range is: $-(2^{k-1})$ to $+(2^{k-1}-1)$
$= -32$ to $+31$

In this case, A does not fall within this range and so overflow would occur when trying to do all four calculations A+B, A−B, B−A, and −A−B.

8. Go Jays!

71 111 32 74 97 121 115 33

Ascii encoding = 71 111 32 74 97 121 115 33

Hexadecimal encoding = 47 6F 20 4A 61 79 73 21

$71_{10} = 1000111_2 = 47_{16}$

$111_{10} = 1101111_2 = 6F_{16}$

$32_{10} = 100000_2 = 20_{16}$

$74_{10} = 1001010_2 = 4A_{16}$

$97_{10} = 1100001_2 = 61_{16}$

$121_{10} = 1111001_2 = 79_{16}$

$115_{10} = 1110011_2 = 73_{16}$

$33_{10} = 100001_2 = 21_{16}$

9. Character "8" = 00111000_2 (odd parity)

Sender Transmits:

00111000

After single bit error:

10111000 = 8

01111000 = X

00011000 = "CAN"

00101000 = (

00110000 = 0

00111100 = <

00111010 = :

00111001 = 9