Sunil Kublalsingh
Student ID: 40212432
Course: COEN 311 W – Computer Organization and Software
Professor: Anjali Agarwal

Question 1

Question 1 (Data Representation) *15 points*

(a) Convert the following pairs of decimal numbers to 5-bit, signed, 2's complement numbers and add them. State, whether or not overflow occurs in each case.
1) 5 and 10
2) 7 and 13
3) -14 and 11
4) -10 and -13

(b) Repeat Part (a) for the subtract operation, where the second number of each pair is to be subtracted from the first number. State whether or not overflow occurs in each case.

<u>How to detect overflow</u>

Overflow occurs only if we add two positive numbers and we get a negative number, or if we add two negative numbers and get a positive number. We could also think of it as the final carry in bit is different from the final carry out bit.

<u>How to convert negative numbers to 2's complement</u>

We convert to 2's complement by flipping all bits that appear on the left of the least significant 1.

a)

1)

$5 = (00101)_2$
$10 = (01010)_2$

5+10=15

```
 0 0 0 0 0
   0 0 1 0 1
 + 0 1 0 1 0
 -----------
   0 1 1 1 1
```

final carry in = final carry out

$\mathbf{15 = (01111)_2}$
No overflow has occurred.

2)

$7 = (00111)_2$
$13 = (01101)_2$

7+13=20

```
 0 1 1 1 1
   0 0 1 1 1
 + 0 1 1 0 1
 -----------
   1 0 1 0 0
```

final carry in = 1
final carry out = 0

$0 \neq 1$

Sunil Kublalsingh
Student ID: 40212432
Course: COEN 311 W – Computer Organization and Software
Professor: Anjali Agarwal

$result = (\mathbf{10100})_2$

Overflow has occurred because the addition of two positive numbers gave a negative number. The final carry in bit was not equal to the final carry out bit.

3)

$14 = (01110)_2$

$-14 = 2's\ complement\ of\ 14 = (10010)_2$

$11 = (01011)_2$

11-14=-3

$$\begin{array}{r} {\scriptstyle 0\,0\,0\,1\,0} \\ 10010 \\ +\ 01011 \\ \hline 11101 \end{array}$$

$0 = 0$

$3 = (00011)_2$
$-3 = 2's\ complement\ of\ 3 = (\mathbf{11101})_2$

No overflow has occurred.

4)

$10 = (01010)_2$
$-10 = 2's\ complement\ of\ 10 = (10110)_2$

$13 = (01101)_2$
$-13 = 2's\ complement\ of\ 13 = (10011)_2$

-10+(-13) = -23

$$\begin{array}{r} {\scriptstyle 1\,0\,1\,1\,0} \\ 10110 \\ +\ 10011 \\ \hline 01001 \end{array}$$

$1 \neq 0$

$result = (101001)_2$

But we only have 5 bits, so the actual result is
$\boldsymbol{result} = (\mathbf{01001})_2$
Overflow has occurred because the addition of two negative numbers gave a positive number. The final carry in and the final carry out bit were not equal.

Sunil Kublalsingh
Student ID: 40212432
Course: COEN 311 W – Computer Organization and Software
Professor: Anjali Agarwal

b)

1)

$5 = (00101)_2$
$-10 = (10110)_2$

5-10=-5

$$\begin{array}{r} 00100 \\ 00101 \\ +\ 10110 \\ \hline 11011 \end{array}$$

$0 = 0$

$-5 = (\mathbf{11011})_2$
No overflow has occurred.

2)

$7 = (00111)_2$
$-13 = (10011)_2$

7-13=-6

$$\begin{array}{r} 00111 \\ 00111 \\ +\ 10011 \\ \hline 11010 \end{array}$$

$0 = 0$

$-6 = (\mathbf{11010})_2$
No overflow has occurred.

3)

$-14 = (10010)_2$

$11 = (01011)_2$
$-11 = 2's\ complement\ of\ 11 = (10101)_2$

-14-11=-25

$$\begin{array}{r} 1\ 0000 \\ 10010 \\ +\ 10101 \\ \hline 00111 \end{array}$$

$1 \neq 0$

Actual result = $(\mathbf{00111})_2$
Overflow has occurred because subtracting a positive number from a negative number resulted in a positive number. The final carry in bit was not equal to the final carry out bit.

Sunil Kublalsingh
Student ID: 40212432
Course: COEN 311 W – Computer Organization and Software
Professor: Anjali Agarwal

4)

$-10 = (10110)_2$
$13 = (01101)_2$

-10-(-13) = 3

$$\begin{array}{r} 11\ 100 \\ 10110 \\ +\ 01101 \\ \hline 00011 \end{array}$$

$result = (\mathbf{00011})_2$

No overflow has occurred. The final carry in bit and final carry out bit are both 1.

Question 2

Question 2 (Data Representation) *15 points*
Give an interpretation to the following string of bits assuming it is:

1100 0110 1001 0111

- Unsigned Integer
- Signed Integer
- BCD number
- String of ASCII characters
- IEEE 754 Floating Point number

Unsigned Integer

Convert to Hex:

$(1100)_2 = 2^3 + 2^2 = (12)_{10}$ = $C

$C697

$12 * 16^3 + 6 * 16^2 + 9 * 16 + 7 = \mathbf{50839\ \textit{in decimal}}$

Signed Integer

First bit is the sign, which is negative since it's a 1. So, the first number in hex becomes $(0100)_2$ which is $4

-$4697

$-4 * 16^3 + 6 * 16^2 + 9 * 16 + 7 = \mathbf{-14697\ \textit{in decimal}}$

Sunil Kublalsingh
Student ID: 40212432
Course: COEN 311 W – Computer Organization and Software
Professor: Anjali Agarwal

BCD Number

BCD only goes up to decimal 9. So, any binary number after 1001 is invalid. Therefore, the first four bits (C) are invalid.

NA 6 9 7

Not a valid number

String of ASCII Characters

An ASCII character is 8 bits. So, we take $C6 and $97 as the two characters.

Maximum range for ASCII is $7F. Therefore, the two characters are not available as ASCII.

NA NA

IEEE 754 Floating Point Number

S=1 = negative number

E' = 1000 1101

E'=$2^0 + 2^2 + 2^3 + 2^7 = 141$

E=141-127=14

M = 001 0111

Formula: $(-1)^S x 2^{E'-127} x 1.M$

Answer: $(-1)^1 x 2^{14} x 1.001\ 0111$

$= -(\mathbf{1.0010111})\boldsymbol{x}\mathbf{2^{14}}$

Question 3

Question 3 (Memory) *10 points*
Consider a computer that has a *byte organized* memory. A program reads numbers entered at a keyboard and stores them as *words* in successive byte locations, starting at location 1000. Show the contents of memory locations when decimals (-14) and (11) are entered, and their addition is stored in the successive location(s).

Byte organized memory means each memory location stores 8 bits (1 byte) of data.

Since we're storing the numbers as words, each number uses 16 bits (2 bytes) of data. Therefore, we need to use two memory locations per number.

Sunil Kublalsingh
Student ID: 40212432
Course: COEN 311 W – Computer Organization and Software
Professor: Anjali Agarwal

-14 in binary = 2's complement of 14 in binary

14: $(0000\ 1110)_2$

-14: $(1111\ 0010)_2 = \$F2$

$-14\ as\ a\ word: (1111\ 1111\ 1111\ 0010)_2 = \$FFF2$

11: $(0000\ 1011)_2 = \$0B$

$11\ as\ a\ word: (0000\ 0000\ 0000\ 1011)_2 = \$000B$

11-14=-3

3: $(0000\ 0011)_2$

-3: $(1111\ 1101)_2 = \$FD$

$-3\ as\ a\ word: (1111\ 1111\ 1111\ 1101)_2 = \$FFFD$

Address	**Value**
$1000	**$FFF2**
$1002	**$000B**
$1004	**$FFFD**

Question 4

Question 4 (Memory) *10 points*
Consider a microprocessor system where the processor has 16-bit data bus and 22-bit address bus. What is the maximum size of the byte addressable memory that can be connected with this processor?

n=16

k=22

Maximum capacity memory to be connected = 2^{22}

$2^{20} * 2^2$

$4 * 1MB$

$4MB$ memory storage

Sunil Kublalsingh
Student ID: 40212432
Course: COEN 311 W – Computer Organization and Software
Professor: Anjali Agarwal

Question 5

Question 5 (Memory) *50 points*
A byte organized memory chip with 12 bit address bus is used as a building block in a larger memory organization.
a) Calculate the capacity of the above chip. *(10 points)*
b) If the above chip is used to build a 64 KByte word organized (16 bit) memory, how many address lines should the CPU have, and how many of these address lines are used for the decoder. *(15 points)*
c) Draw the memory Connections to the CPU for this new Memory System. *(25 points)*

a)

$$C = 2^{12}$$
$$= 2^{10} * 2^{2}$$
$$= \boldsymbol{4KB}$$

b)

Word organized means that each memory location will store 16 bits of data.

$C = 2^k * m \; bytes$

$64KB = 2^k * \frac{16}{8} \; bytes$

$64KB = 2^k * 2 \; bytes$

$32KB = 2^k$

$2^5 * 2^{10} = 2^k$

k=15

We need 15 address lines from the CPU.

12 of the bits go to the actual address lines for the memory modules.
15-12=**3 bits go to the decoder.**

c)

We need 64 KB from multiple 4KB memory modules. 64/4=16 modules are needed.

We have 8 data pins from each module, but we need 16 data pins in total. We can have 2 memory modules that are read in parallel to make up 16 data bits. Since we have 16 modules in total, 16/2=8 rows of memory modules that are read in parallel.

Sunil Kublalsingh
Student ID: 40212432
Course: COEN 311 W – Computer Organization and Software
Professor: Anjali Agarwal