Questions 1-9: Chapter 2, exercises 2.8, 2.10, 2.11, 2.13, 2.15, 2.17, 2.18, 2.20, 2.21

2.8

a. What is the largest positive number one can represent in an eight-bit 2’s complement code? Write your result in binary and decimal.

The largest positive number one can represent in eight-bit 2’s is 127 in decimal and in binary that would be 01111111.

b. What is the greatest magnitude negative number one can represent in an eight-bit 2’s complement code? Write your result in binary and decimal.

The smallest negative number one can represent in eight-bit 2’s is -128 in decimal and in binary it would be -10000000.

c. What is the largest positive number one can represent in n-bit 2’s complement code?

The largest positive number that an n-bit 2's complement code can represent is one is $2n-1, or $1.

d. What is the greatest magnitude negative number one can represent in n-bit 2’s complement code?

$-2n-1$ is the largest magnitude negative number that an n-bit 2's complement code can represent.

2.10

Convert the following 2’s complement binary numbers to decimal.

a. 1010-decimal number-10

b. 01011010-decimal number-90

c. 11111110-decimal number-254

d. 0011100111010011-decimal number-12691

2.11

Convert these decimal numbers to eight-bit 2’s complement binary numbers.

a. 102-01100110
b. 64-01000000
c. 33-00100001
d. -128 → -10000000
e. 127-01111111

2.13

Without changing their values, convert the following 2’s complement

binary numbers into eight-bit 2's complement numbers.
a. 1010→11111010
b. 011001→00011001
c. 1111111000→11111000
d. 01→00000001

2.15
It was demonstrated in Example 2.5 that shifting a binary number one bit to the left is equivalent to multiplying the number by 2. What operation is performed when a binary number is shifted one bit to the right?

A binary number can be divided by two by shifting it one bit to the right. This is due to the fact that each bit position represents a power of 2, and moving one bit to the right dilutes a power of 2 by one.

2.17
Add the following 2's complement binary numbers. Also express the answer in decimal.
a. 01 + 1011→-4
b. 11 + 01010101→84
c. 0101 + 110→3
d. 01 + 10→-1

2.18
Add the following unsigned binary numbers. Also, express the answer in decimal.
a. 01 + 1011→12
b. 11 + 01010101→84
c. 0101 + 110→19
d. 01 + 10→3

2.20
The following binary numbers are four-bit 2's complement binary numbers. Which of the following operations generate overflow? Justify your answer by translating the operands and results into decimal.
a. 1100 + 0011→1111
b. 1100 + 0100→10000(this is an overflow)
c. 0111 + 0001→1000
d. 1000 − 0001→0111
e. 0111 + 1001→10000(this is an overflow)

2.21
Describe what conditions indicate overflow has occurred when two 2’s
complement numbers are added.
Overflow can occur when the addition of two positive numbers gives a negative number. Or when the addition of two negative numbers results in a positive result. It can also occur when the signs of the two input numbers are the same but the sign of the result is different.

Question 10: We explored three different ways of representing integers (i.e whole numbers including both positive & negative): signed magnitude (failed!); one's complement (not perfect!); and two's complement (just right!).

Perform the arithmetic operations (+5) + (-7) , and (-3) + (+5) in all three representations.

(+5) + (-7)
Binary notation: 0101 + 1001 = 1110. (14 in decimal)
Binary: 1111 1001 + 0000 0101 equals 0000 0000 (0 in decimal)
8-bit binary with two complements: 0000 0101 plus 1111 1001 equals 0000 0000 (0 in decimal)

(-3) + (+5)
Binary unsigned: 1101 + 0101 = 0010 (2 in decimal)
Binary: 0000 0101 plus 1111 1101 equals 0000 0010 (2 in decimal)
8-bit binary with two complements: 0000 0101 + 1111 1101 = 0000 0010 (2 in decimal)

What can you say about the outcomes?

The outcome for both procedures is different for unsigned binary, it is the same for binary and 8-bit 2's complement binary. Due to the fact that unsigned binary cannot represent negative values, the result of the addition operation is not within the range that can be represented. In contrast, negative numbers can be represented using the two's complement representation, thus the outcome of an addition operation can always be represented.