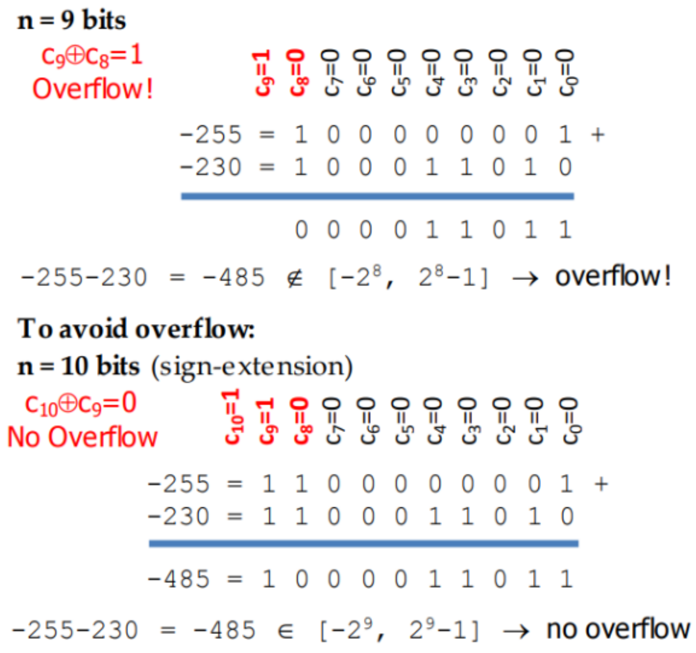
Quiz 1 Solutions
CSE 112 Computer Organization

Question 1:



n = 9 bits

$c_9 \oplus c_8 = 1$
Overflow!

$c_9=1$ $c_8=0$ $c_7=0$ $c_6=0$ $c_5=0$ $c_4=0$ $c_3=0$ $c_2=0$ $c_1=0$ $c_0=0$

```
-255 = 1 0 0 0 0 0 0 0 1 +
-230 = 1 0 0 0 1 1 0 1 0

       0 0 0 0 1 1 0 1 1
```

$-255-230 = -485 \notin [-2^8, 2^8-1]$ → overflow!

To avoid overflow:

n = 10 bits (sign-extension)

$c_{10} \oplus c_9 = 0$
No Overflow

$c_{10}=1$ $c_9=1$ $c_8=0$ $c_7=0$ $c_6=0$ $c_5=0$ $c_4=0$ $c_3=0$ $c_2=0$ $c_1=0$ $c_0=0$

```
-255 = 1 1 0 0 0 0 0 0 0 1 +
-230 = 1 1 0 0 0 1 1 0 1 0

-485 = 1 0 0 0 0 1 1 0 1 1
```

$-255-230 = -485 \in [-2^9, 2^9-1]$ → no overflow

1. **[1 mark for 2's complement notation of -255]**
2. **[1 mark for 2's complement notation of -230]**
3. **[1 mark for computation of 9 LSBs of the result]**
4. **[1 mark for computation of MSB of the result]**
5. **[1 mark to mention the number of bits to represent the result is 10]**

Question 2

0100_0000_1011_0100_0000_0000_0000_0000

a. +ve **[1 marks]**
b. 1000_0001 = 129 = 127 + 2. Therefore 2. **[2 marks]**
c. 011_0100_0000_0000_0000_0000 **[2 marks]**
d. 1.01101_2 x 2^2 = 101.101_2 = 5.625_{10} **[3 marks. Students must show the steps.]**

Question 3

Binary fixed point representation for 17.6875 = 10001.1011 = 1.00011011 x 2^4 **[2 marks]**

Sign = 1 (Negative number) **[1 marks]**

Exponent = 127 + 4 = 1000_0011_2 **[2 marks]**

Mantissa = 000_1101_1000_0000_0000_0000 **[2 marks]**

Therefore: 1100_0001_1000_1101_1000_0000_0000_0000 **[1 marks]**

Question 4

a. 21 **[2 marks]**
b. 21 **[2 marks]**
c. 6 **[2 marks]**
d. 16 **[2 marks]**
e. 2^5 = 32 **[2 marks]**
f. 2^6 = 64 **[2 marks]**

Question 5

a) 1100_1100_1100_1100_1100_1100_0000_0000.
Students should mention that the multiplication is just a left shift operation. **[1 marks. "Left shift" must be mentioned]**
Students should mention the shift is by 8 bits. **[1 mark to shift by 8 bits]**
[1 mark to copy the rest of the number correctly]

b) In case of IEEE754 floating point number when multiplying with 2, we add 1 to the exponent part of the IEEE format
Given number = 0100_1010_1011_1100_1011_0000_0000_0000
Sign = 0
Exponent = 1001_0101
New Exponent = 1001_0101 + 1 = 1001_0110
Mantissa = 011_1100_1011_0000_0000_0000
New Number after multiplying by 2 = {Sign, New Exponent, Mantissa}
= 0100_1011_0011_1100_1011_0000_0000_0000
[1 marks for sign bit]
[2 for exponent part]
[1 mark for mantissa part]