Q1.a) Find the 16-bit 2’s complementary binary representation for the decimal number 1987.

Ans) Divide 1987 by 2 until the quotient is 0

1987/2 = 993, remainder 1

993/2 = 496, remainder 1

496/2 = 248 remainder 0

248/2 = 124 remainder 0

124/2 = 62 remainder 0

62/2 = 31 remainder 0

31/2 = 15 remainder 1

15/2 = 7 remainder 1

7/2 = 3 remainder 1

3/2 = 1 remainder 1

1/2 = 0 remainder 1

Taking reminder from bottom to up as 11111000011

Adding 5 zeros on left hand side will make 16 bit

0000 0111 1100 0011

So, 2’s complementary binary representation of 1987 is 0000 0111 1100 0011

1.b) Find the 16-bit 2’s complementary binary representation for the decimal number -1987.

-1987 is the 2’s complement of 1987

So, we will take 1987 2’s complement as input and invert it and then add 1 to it to get 2’s

Complement of -1987.

0000 0111 1100 0011 => invert it => 1111 1000 0011 1100

Now, add 1 to it.

So, 1111 1000 0011 1101 is the 2’s complementary binary representation of -1987

1.c) From your answer in (b), find the six-digit 16’s complement hexadecimal representation

for the decimal number -1987.

First converting 1987 to hexadecimal which is 7C3 => 0007C3

Now, we will take 15’s complement by subtracting every digit from F

F-0=>F

F-0=>F

F-0=>F

F-7=>8

F-C=>3

F-3=>C

Now that we have got 15’s complement which is FFF83C, we will add 1 to it, to make

16’s complement, so adding 1 to least significant digit will give us FFF83D.

So, the six-digit 16’s complement hexadecimal representation for the decimal number of

-1987 is FFF83D

Q2) What are the 16-bit 1’s and 2’s complements of the following numbers?

a)10000

10000 =>16-bit format => 0000 0000 0001 0000

For one’s complement invert it.

1’s complement - 1111 1111 1110 1111

For 2’s complement adds 1 to least significant bit.

2’s complement is 1111 1111 1111 0000

b)100111100001001

100111100001001=>16-bit format=> 0100 1111 0000 1001

1’s complement => invert it => 1011 0000 1111 0110

2’s complement- add 1 to 1’s complement=> 1011 0000 1111 0111

c) 0100111000100100

0100111000100100=>16-bit format=> 0100 1110 0010 0100

1’s complement => invert it => 1011 0001 1101 1011

2’s complement=> add 1 to 1’s complement => 1011 0001 1101 1100