CONVERT THE DECIMAL NUMBER -75.125 TO 32 BIT IEEE floating POINT FORMAT. EXPRESSING THE ANSWER IN HEXADECIMAL FOR CONVENIENCE

① CONVERT TO BINARY

2 | 75			
2 | 37	R	1	
2 | 18	R	1	
2 | 9	R	0	
2 | 4	R	1	
2 | 2	R	0	
2 | 1	R	0	
0	R	1	

$.125 \times 2 = \boxed{0}.25$

$.25 \times 2 = \boxed{0}.5$

$.5 \times 2 = \boxed{1}.0$

$1001011.00|_2$

(2) NORMALISE (EXPRESS IT IN SCIENTIFIC NOTATION)

1001011.001

MOVE . TO AFTER FIRST 1

1.001 011 001

HOW MANY PLACES DOES THE POINT HAVE TO BE MOVED FORWARD TO GET TO WHERE IT WAS ORIGENALLY? (6) PLACES

$1.001\ 011\ 001 \times 2^{6}$

(3) IDENTIFY SIGN EXPONENT AND MANTISSA

Sign 0 FOR + NO, 1 FOR NEG NO, 0

EXPONENT 6

MANTISSA 1.001 011 001

(BUT DROP LEADING 1 AND BINARY POINT TO SAVE SPACE) 001 011 001

MANTISSA

(3)

TABLE — SOMETHING YOU NEED TO KNOW

HEXADECIMAL	BINARY
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

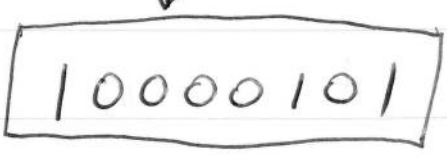
④

④ ADD 127 TO EXPONENT AND CONVERT TO BENARY

$$6 + 127 = 133$$

2	133		
2	66	R	1
2	33	R	0
2	16	R	1
2	8	R	0
2	4	R	0
2	2	R	0
2	1	R	0
	0	R	1

EXPONENT



(5) ASSEMBLE THE NO.
(AND CONVERT TO HEXADECIMAL FOR CONVENIENCE)

1 BIT	8 BITS	23 BITS
Sign	EXPONENT	MANTISSA
0	10000101	001 011 001 0000 0000 0000 00

EXPRESS THE ABOVE NO. IN HEXADECIMAL

0 1000 0101 001 011 0010 0000 0000 0000 00

4 2 9 6 4 0 0 0

42964000_{16}

EXERCISE

Convert the decimal number 123.0625_{10} to IEEE 32 bit floating point format.

① CONVERT TO BINARY

2	123		
2	61	R	1
2	30	R	1
2	15	R	0
2	7	R	1
2	3	R	1
2	1	R	1
	0	R	1

$.0625 \times 2 = \boxed{0}.125$

$.125 \times 2 = \boxed{0}.25$

$.25 \times 2 = \boxed{0}.5$

$.5 \times 2 = \boxed{1}.0$

0

$$1111011.0001_2$$

(2) NORMALIZE (EXPRESS IN SCIENTIFIC NOTATION)

1111011.0001

MOVE . TO AFTER THE FIRST OCCURENCE OF A 1

$$\underline{1.1110110001} \times 2^6$$

(THE POINT HAS TO BE MOVED FORWARD 6 PLACES TO GET TO WHERE IT WAS ORIGINALLY)

(3) IDENTIFY:

SIGN	0	(POSITIVE)
EXPONENT	6	
MANTISSA	1.1110110001	
	1110110001	

(DROP LEADING 1 AND BINARY POINT) →

④ ADD 127 TO EXPONENT AND CONVERT TO BINARY

6 + 127 = 133

2	133		
2	66	R	1
2	33	R	0
2	16	R	1
2	8	R	0
2	4	R	0
2	2	R	0
2	1	R	0
	0	R	1

EXPONENT

10000101

(5) ASSEMBLE THE NO. AND CONVERT TO HEXADECIMAL (HEX.) FOR CONVENIENCE

1 BIT	8 BITS	23 BITS
SIGN	EXPONENT	MANTISSA
0	10000101	11101100010000000000000

EXPRESS THE ABOVE NO. IN HEXADECIMAL

0 1000 0101 1110 1100 0100 0000 0000 0000

4 2 F 6 2 0 0 0

$42F62000_{16}$

EXERCISE

CONVERT -12345.625_{10} TO 32 BIT IEEE FLOATING POINT FORMAT.

EXPRESS YOUR ANSWER IN HEXADECIMAL.

ANSWER: C640E680

EXERCISE

CONVERT 8193.125_{10} TO 32 BIT IEEE FLOATING POINT FORMAT

① CONVERT TO BINARY

1000000000001.001_2

② NORMALIZE

$1.000\ 000\ 000\ 000\ 1001 \times 2^{13}$

(3) IDENTIFY:

SIGN | 0 |

EXPONENT 13

MANTISSA 1.0000000000000|001|

DROP LEADING 1 AND .

| 0000000000000|001 |

← MANTISSA

(4) ADD 127 TO EXPONENT AND CONVERT TO BINARY

$13 + 127 = 140$

2	140		
2	70	R	0
2	35	R	0
2	17	R	1
2	8	R	1
2	4	R	0
2	2	R	0
2	1	R	0
	0	R	1

EXPONENT

10001100

(5) ASSEMBLE THE NO.
(AND CONVERT TO HEX.)

1 BIT	8 BITS	23 BITS
Sign	EXPONENT	MANTISSA
0	10001100	00000000000010010000000

0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0

4 6 0 0 0 4 8 0

46000480

CONVERT -8193.125 TO 32 BIT IEEE FLOATING POINT FORMAT

SIGN	EXPONENT	MANTISSA
1	10001100	00000000000010010000000

↑

THIS HAS CHANGED FROM PREVIOUS EXERCISE

1100

↓

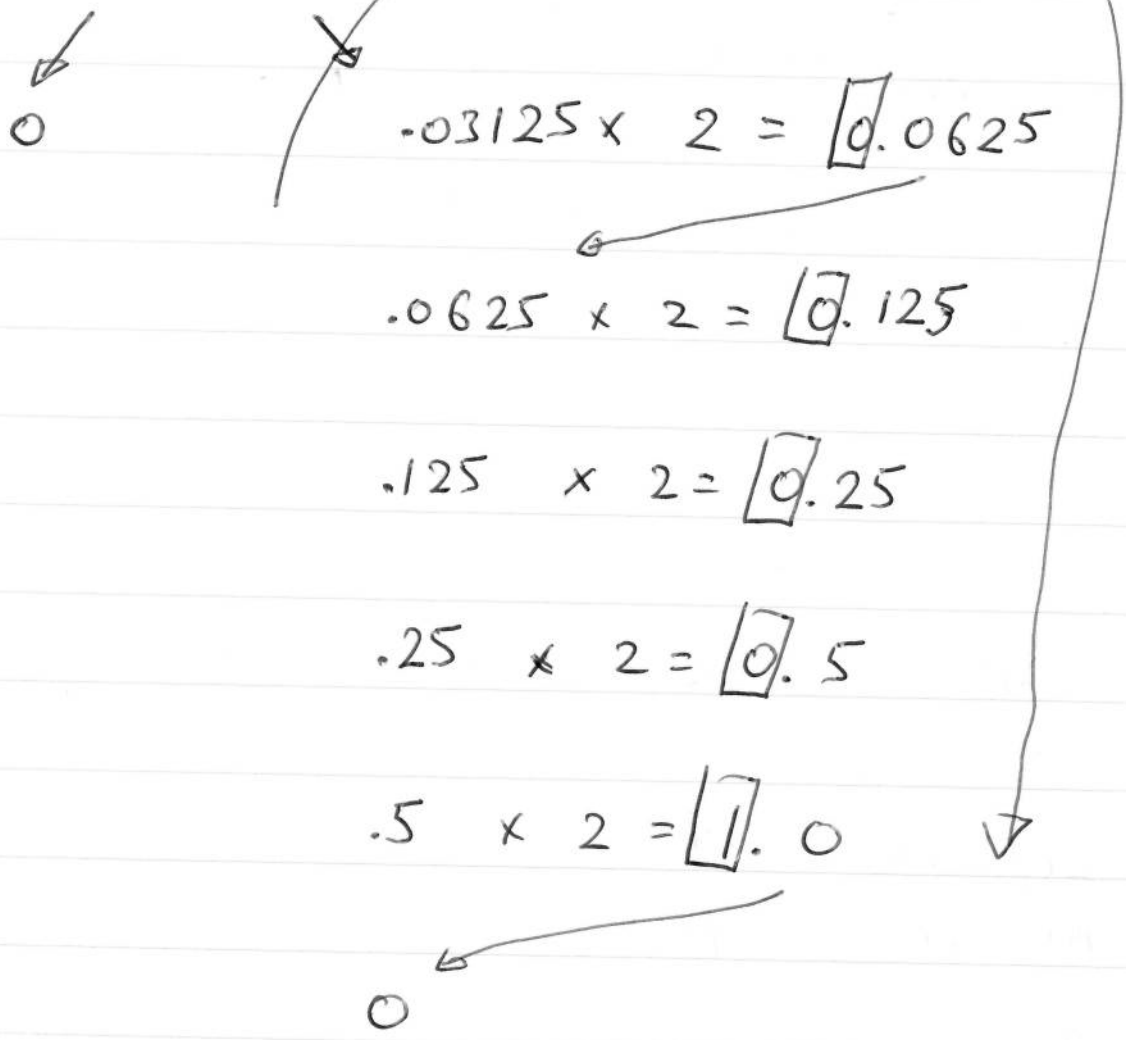
C6000480

EXERCISE

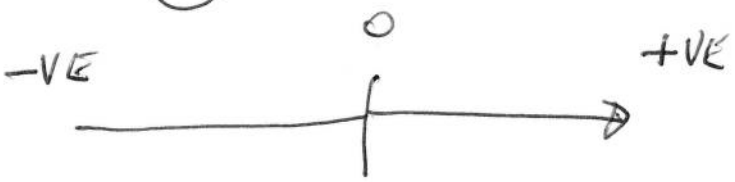
CONVERT 0.03125_{10} TO 32 BIT IEEE FLOATING POINT FORMAT

① CONVERT TO BINARY

0.03125



0.00001_2



(2) NORMALIZE

0.00001

5 PLACES TO THE RIGHT

1.0×2^{-5}

RIGHT =>
NEGATIVE
EXPONENT

(3) IDENTIFY SIGN EXPONENT AND MANTISSA

SIGN 0
EXPONENT -5
MANTISSA 1.0

DROP LEADING 1 AND .

0 & MANTISSA

④ ADD 127 TO EXPONENT AND CONVERT TO BINARY

$-5 + 127 = 122$

2 | 122			
2 | 61	R	0	
2 | 30	R	1	
2 | 15	R	0	
2 | 7	R	1	
2 | 3	R	1	
2 | 1	R	1	
0	R	1	

7 BITS

1111010

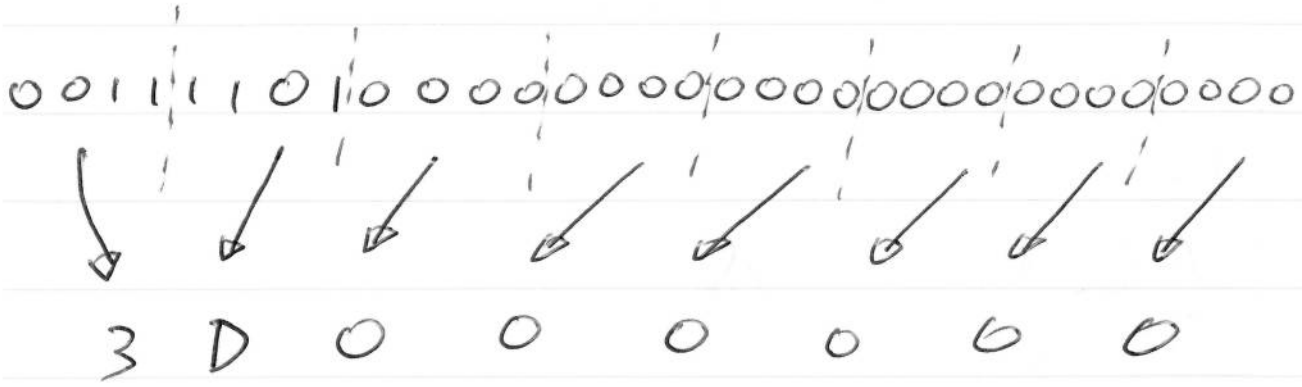
01111010

8 BIT EXPONENT

0 ADDED HERE TO MAKE IT 8 BITS

(5) ASSEMBLE NO. AND CONVERT TO HEX

S	E	M
0	01111010	00000000000000000000000



3D000000