3.1

Hex

$$
\begin{array}{r}
5ED4 \\
-\ 07A4 \\
\hline
= (5730)_{16}
\end{array}
$$

3.4

Octal

$$
\begin{array}{r}
4365 \\
-\ 3412 \\
\hline
(753)_8
\end{array}
$$

3.23 63.25

single precision format

$63.25 \times 10^0 = 111111.01 \times 2^0$

Normalize, move binary point five digits to left.

$\Rightarrow 1.1111101 \times 2^5$

Sign = positive, exponent = $127 + 5 = 132$

Final bit pattern = 0 1000 0100 1111 1010 0000 0000 0000 00

= 0100 0010 0111 1101 0000 0000 0000 0000

= 0x427D0000

3.41 IEEE 754 floating point format

Bit pattern for $\frac{-1}{4}$.

How can you represent $-\frac{1}{4}$ exactly?

[A]

S	Exponent	Fraction	sign	exponent	Exact?
1	01111101	00000000000000000000000	−	−2	Yes

(3.24)

$63.25 \times 10^{0} = 111111.01 \times 2^{0}$

Normalize, move binary point five digits to the left

1.1111101×2^{5}

Sign = positive,

Exponent = $1023 + 5 = 1028$

Final bit pattern:

0 100 0000 0100 1111 1010 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

= 0x 404FA0000000000000

(3.14) For hardware, it takes one cycle to do the add, one cycle to do the shift, and one cycle to decide if we are done. So, the loop takes (3×A) cycles, with each cycle being B time units long.

For a software implementation, it takes one cycle to decide what to add, one cycle to do the add, one cycle to do each shift, and one cycle to decide if we are done. So, the loop takes

(5×A) cycles, with each cycle being B time units long.

$(3\times 8) \times 4$ TUs = 96 TUs for hardware

$(5\times 8) \times 4$ TUs = 160 TUs for software

(3.38) $1.666015625 \times 10^{0} \times (1.976 \times 10^{4} - 1.9744 \times 10^{4})$

(A) $1.666015625 \times 10^{0} = 1.1010101010 \times 2^{0}$

(B) $1.9760 \times 10^{4} = 1.0011010011 \times 2^{14}$

(C) $-1.9744 \times 10^{4} = -1.0011010010 \times 2^{14}$

Exponents match, no shifting necessary

$$
\begin{array}{ll}
\text{(B)} & 1.0011010011 \\
\text{(C)} & \underline{-1.0011010010} \\
 & 0.0000000001 \times 2^{14} \\
\text{(B+C)} & = 1.0000000000 \times 2^{4}
\end{array}
$$

Exponent : $0+4=4$, Signs : Both positive ⇒ Result is positive

Fraction :

$$
\begin{array}{ll}
\text{(A)} & 1.1010101010 \\
\text{(B+C)} & \underline{\times 1.0000000000} \\
 & 1.1010101010\ 00000000000000 \\
 & 1.1010101010\ 0000000000
\end{array}
$$

(A) × (B+C) : 1.1010101010 0000000000

Guard = 0, Round = 0, Sticky = 0 : No round

(A) × (B+C) = $\underline{1.1010101010 \times 2^{4}}$

3.27

$-1.5625 \times 10^{-1} = -0.15625 \times 10^{0}$

$= -0.00101 \times 2^{0}$

Move the binary point three places to the right

$\Rightarrow -1.01 \times 2^{-3}$

Exponent $= -3 = \underset{\text{(bias)}}{-3 + 15} = 12$

Fraction $= -0.0100000000$

Answer $= 1011000100000000$

3.30 $-8.0546875 \times -1.79931640625 \times 10^{-1}$

$\Rightarrow -8.0546875 = -1.0000000111 \times 2^{3}$

$-1.79931640625 \times 10^{-1} = -1.0111000010 \times 2^{-3}$

Exponent : $-3 + 3 = 0$, $0 + 16 = 16$ (10000)

Signs : Both negative $\Rightarrow$ Result is positive

Fraction :

```
                 1.0000000111
               × 1.0111000010
               ______________
                  00000000000
                 10000000111
                00000000000
               00000000000
              00000000000
             00000000000
            10000000111
           10000000111
          10000000111
         00000000000
        10000000111
       00000000000
   +  10000000111
   ____________________________
   =  1.01110011 00000 1001110
   ____________________________
```

Guard = 0

Round = 0

Sticky = 1

No Rnd.

3.13 62 × 12

Step	Action	Multiplicand	Product / Multiplier
0	Initial Vals	110 010	000 000 001 010
1	lsb = 0, no op	110 010	000 000 001 010
	Rshift product	110 010	000 000 000 101
2	Prod = Prod + Mcand	110 010	110 010 000 101
	Rshift Multiplier	110 010	011 001 000 010
3	lsb = 0, no op	110 010	011 001 000 010
	Rshift Multiplier	110 010	001 100 100 001
4	Prod = Prod + Mcand	110 010	111 110 100 001
	Rshift Multiplier	110 010	011 111 010 000
5	lsb = 0, no op	110 010	011 111 010 000
	Rshift Multiplier	110 010	001 111 101 000
6	lsb = 0, no op	110 010	001 111 101 000
	Rshift Mplier	110 010	000 111 110 100

3.18

Assume that 74 and 21 are octal $\Rightarrow$ 60 % 17 $\equiv$ 9

Iteration	Step	Quotient	Divisor	Remainder
0	Initial values	000000	010001 000000	000000 111100
1	1. Rem = Rem − Div	000000	010001 000000	101111 111100
	2b: Rem < 0 $\Rightarrow$ + Div, sll Q, $Q_0 = 0$	000000	010001 000000	000000 111100
	3: Shift Div right	000000	001000 100000	000000 111100
2.	1. Rem = Rem − Div	000000	001000 100000	111000 011100
	2b: Rem < 0 $\Rightarrow$ + Div, sll Q, $Q_0 = 0$	000000	001000 100000	000000 111100
	3. Shift Div right	000000	000100 010000	000000 111100
3	1. Rem = Rem − Div	000000	000100 010000	111100 101100
	2b: Rem < 0 $\Rightarrow$ + Div, sll Q, $Q_0 = 0$	000000	000100 010000	000000 111100
	3. Shift Div right	000000	000010 001000	000000 111100

4

Iteration	Step	Quotient	Divisor	Remainder
4	1: Rem = Rem − Div	000000	000010 001000	111110 110100
	2b: Rem<0 ⇒ +Div, sll Q, $Q_0 = 0$	000000	000010 001000	000000 111100
	3: Shift Div right	000000	000001 000100	000000 111100
5.	1: Rem = Rem − Div	000000	000001 000100	111111 111000
	2b: Rem<0 ⇒ +Div, sll Q, $Q_0 = 0$	000000	000001 000100	000000 111100
	3: Shift Div Right	000000	000000 100010	000000 111100
6.	1: Rem = Rem − Div	000000	000000 100010	000000 011010
	2a: Rem ≥ 0 ⇒ sll Q, $Q_0 = 1$	000001	000000 100010	000000 011010
	3: Shift Div right	000001	000000 010001	000000 011010
7.	1: Rem = Rem − Div	000001	000000 010001	000000 001001
	2a: Rem ≥ 0 ⇒ sll Q, $Q_0 = 1$	000011	000000 010001	000000 001001
	3: Shift Div Right	000011	000000 001000	000000 001001

3.31. Consider the following values:

$A = 8.625 \times 10^1 = 86.25 = 1010110.01 \times 2^0$

$= 1.01011001 \times 2^6$

$B = -4.875 \times 10^0 = -4.875 = -100.111 \times 2^0$

In this step, subtracting exponent without bias and with bias is done to calculate the new bias. Adding exponents without bias:

$6-2=4$

Using the biased representation: $(6+15)+(2+15)-15 = 19$

. Division of significand is done:

$$\begin{array}{r|l} & 10010 \\ \hline 1.00111 & 1.01011001 \\ & 1.00111 \\ \hline & 0.0111111 \\ \hline \end{array}$$

Rounding product makes no change: 1.0010×2^4

Since sign of one operand is negative, sign of product is negative.

Result in 16-bit floating point format:

Consider the following binary number: -1.0010×2^4.

The general representation of half precision number is:

$(-1) \times (1+\text{fraction}) \times 2^{(\text{Exponent}-15)}$

$= (-1)^1 \times (1+0.0010000000) \times 2^{19-15}$.

Half precision of binary representation of the number is then

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
11 0011	001 000 0000														

Result in decimal format :

Converting result to decimal to check results :

Converting to decimal = $-1.0010 \times 2^4 = -10010 = -18$

By using calculator, division of 86.25 and -4.875 results in 17.692, which is approximately equal to $\underline{\underline{18}}$.