《计算机系统基础》分析应用题

第二章作业

学号	5120203245	班级	软件卓越 2001	姓名	肖尧

题目 1

1\. -8196的原码表示为 1010 0000 0000 0100

其补码表示为 1101 1111 1111 1100.

而usi的机器数与si相同，但每一位均为数值位

$usi = (2^{16}-1) - 2^{13} - 2^{1} - 2^{0} = 57340.$

题目 2

（1）从形式上看，[x]补的符号位为 1，故一定是负数。因此，绝对值越大数值越小。要满足 x<-1/2 仅需 x6 为 0，其余位至少有一个 1 即可。（至少有一个 1 是为了防止取反加 1 后符号位溢出）。

（2）若要满足-1/2<x<-1/4，则 x6 必须为 1，即 x>-1/2，且 x5 必须为 0，否则将不满足 x<-1/4。

题目 3

3\. (1) $[x]_{原} = 1001\ 1111$ $\therefore x = -11111B = -31$

(2) $x = -2^{7} = -128$

(3) $[x]_{原} = [x]_{补} = 0111\ 1111$ $\therefore x = 2^{7}-1 = 127$

(4) $[x]_{原} = 1000\ 0001$ $\therefore x = -1$

题目 4

x-y=-238

二进制表示为-11101110

z 为 short 类型，共 16 位

其原码表示为：10000000 11101110

其补码表示为：11111111 00010010

题目 5

i=65535
si 的机器数为 1111 1111 1111 1111
si 以补码形式编码，其真值为-1
Int j = si 将 j 的高位进行符号位拓展，最终得到 32 个 1
j 为带符号整数，故 j 的真值仍为-1

题目 6

（1）x>>(n-8)<<(n-8)
（2） 方法 1：与(1)同理，x<<(n-8)>>(n-8)
　　　方法 2：x&0xFF
（3）（~x）>>8<<8
（4）通过或运算将第八位变为 1：x|0xFF
实验验证：（1 过于简单，仅验证了 234 小题）

```c
#include<stdio.h>
int main(){
        int n = 32;
        int x = 0x12345679;
        int y = 0x12345679;
        int z = x;
        x = x<<(n-8)>>(n-8);
        y = (~y)>>8<<8;
        z = z|0xFF;
        printf("%x\n%x\n%x\n",x,y,z);
        return 0;
}
```

```
(gdb) s
7               x = x<<(n-8)>>(n-8);
(gdb) i r esp ebp
esp            0xffffd450       0xffffd450
ebp            0xffffd468       0xffffd468
(gdb) x/7xw $esp
0xffffd450:     0x00000020      0x12345679      0x12345679      0x12345679   原始数据
0xffffd460:     0xffffd480      0x00000000      0x00000000
(gdb) s
8               y = (~y)>>8<<8;
(gdb) x/7xw $esp
0xffffd450:     0x00000020      0x00000079      0x12345679      0x12345679
0xffffd460:     0xffffd480      0x00000000      0x00000000
(gdb) s
9               z = z|0xFF;
(gdb) x/7xw $esp
0xffffd450:     0x00000020      0x00000079      0xedcba900      0x12345679
0xffffd460:     0xffffd480      0x00000000      0x00000000
(gdb) s
10              printf("%x\n%x\n%x\n",x,y,z);
(gdb) x/7xw $esp
0xffffd450:     0x00000020      0x00000079      0xedcba900      0x123456ff
0xffffd460:     0xffffd480      0x00000000      0x00000000
(gdb)
```

题目 7

7.(1)永真。IEEE 754 标准中符号位和数值部分分开运算，溢出不影响乘除、符号。

(2)不永真。int 转为 ~~double~~ float 类型时(32 > 24)会造成有效位数丢失，即左边部分为近似值，而右边的 dx 为 int 转 double 得到不会发生精度损失，为精确值。

(3).不永真。右边 x+y 在转为 double 之前可能会溢出，~~[illegible]~~ 而左边 double 由两个 int 相加得到而不会溢出。

(4)永真。这 3 个 double 类型的数均由 int 转换得到，而 double 的尾数位数为 53，即使不计入保护位也不会在对阶时发生溢出。

(5). ~~非~~永真。相乘顺序的不同会导致舍入的不同。

(6)不永真。0/0 为 NaN 而不为 ~~非零~~ 相除的结果！

题目 8 无附加位情况

无附加位：

8.(1)① $\frac{15}{16}\times 2^7 = \frac{2^3+2^2+2^1+2^0}{2^4} = (2^{-1}+2^{-2}+2^{-3}+2^{-4})\times 2^7$

$= 0.1111B\times 2^7$ 浮点表示为 $00.\underbrace{1111}_{\text{阶码}}\ \underbrace{1111}_{\text{尾数}}$

阶码 $7+8=15=1111B$

② $\frac{2}{16}\times 2^5 = 0.0010B\times 2^5 = 0.1B\times 2^3$. 阶码 $3+8=11=1011B$.

浮点表示为 $00.\underline{1011}\ \underline{1000}$

③ 对阶：$[\Delta E]_{补} = 1111-1011 = 0100$ 即 $\Delta E = 4$.

④ 将②的尾数右移 4 位得到 $\underline{0000}$ (偶数)，阶码变为 $\underline{1111}$.

尾数相加得到 1111

即 $\frac{15}{16}\times 2^7 + \frac{2}{16}\times 2^5 = 00.\underline{1111}\ \underline{1111} = \frac{15}{16}\times 2^7$.

(2). 与(1)基本相同，尾数相减仍为 1111

即 $\frac{15}{16}\times 2^7 - \frac{2}{16}\times 2^5 = 00.\underline{1111}\ \underline{1111} = \frac{15}{16}\times 2^7$.

(3) ① $\frac{15}{16}\times 2^5 = 0.1111B\times 2^5$. 浮点表示为 $00.\underline{1101}\ \underline{1111}$

② $\frac{2}{16}\times 2^7 = 0.001B\times 2^7 = 0.1B\times 2^5$ 阶码 $5+8=11=1011B$.

浮点表示为 $00.\underline{1011}\ \underline{1000}$

③ 对阶：$[\Delta E]_{补} = 1101 - 1011 = 0000$ 无需对阶直接尾数相加

④. 尾数相加：1111 + 1000 = 1|0111 尾数溢出进行右规.
阶码加1变为 1110
尾数~~右规~~舍入利用就近舍入到偶数为 1100
即 $\frac{15}{16}\times 2^5+\frac{2}{16}\times 2^7 = 00.\underline{1110}\ \underline{1100} = \frac{12}{16}\times 2^6 = \frac{3}{4}\times 2^6$.
(4) 与(3)基本相同，尾数相减：1111 − 1000 = 0111 需左规.
阶码减1变为 1100
尾数左规得到 1110.
即 $\frac{15}{16}\times 2^5+\frac{2}{16}\times 2^7 = 00.\underline{1100}\ \underline{1110} = \frac{14}{16}\times 2^4 = \frac{7}{8}\times 2^4$.

题目 8 带 2 位附加位情况

利用2位附加位（基本信息与上述一致且这里序号按上面通用）
(1). ΔE = 4
对阶时对②的尾数右移4位得到 0000$\underset{=\text{附加位}}{\underline{\underline{10}}}$
尾数相加为 1111$\underline{\underline{00}}$ + 0000$\underline{\underline{10}}$ = 1111$\underline{\underline{10}}$
对尾数附加位进行舍入，10为中间值强迫结果为偶数，
得尾数 $\underline{\underline{01}}$.0000，右规得到 00.1000
（01 → 尾数符号位）
阶码 1111 + 1 产生阶码上溢 1|0000 故结果溢出.
(2). ΔE = 4 ~~对阶时②的尾数~~ 对阶与(1)相同.
尾数相减 1111$\underline{\underline{00}}$ − 0000$\underline{\underline{10}}$ = 1110$\underline{\underline{10}}$.
对尾数附加位进行舍入，10为中间值，末位为偶数故舍去
得尾数 1110（与无附加位的差异）
结果为 00.$\underline{1111}\ \underline{1110}$ = $\frac{14}{16}\times 2^7 = \frac{7}{8}\times 2^7$
(3). ΔE = 0. 尾数相加 00.1111$\underline{\underline{00}}$ + 01.000$\underline{\underline{00}}$ = 01.011100 尾数溢出右规.
阶码加1得 1110
尾数利用就近舍入到偶数为 $\underline{1011}$(100) $\xrightarrow{\text{中间值}}$ 1100
结果为 00.$\underline{1110}\ \underline{1100}$ 与无附加位结果相同.
(4) ΔE = 0 尾数相减 111100 − 100000 = 011100. 尾数需左规.
阶码减1得 1100.
尾数为 1110
结果为 00.$\underline{1100}\ \underline{1110}$ 与无附加位结果相同

附：解题过程手稿

1、-8196的原码表示为 1010000000000100

其补码表示为 ~~1011111111111000~~ 1101,1111,1111,1100.

而usi的机器数与si相同，但每一位均为数值位

$usi = 2^{16} - 1 - 2^{13} - 2^{1} - 2^{0}$

$= 65536 - 1 - 8192 - 2 - 1$

$= 57340$

2.(1) 从形式上来看，$[x]_补$的符号位为1，故x一定是负数。因此绝对值越大，数值越小，要满足 $x < -\frac{1}{2}$ 仅需 x_6 为0，其余位至少有一个为1即可。(这是为了防止取反加1后符号位溢出)

(2) 若要满足 $-\frac{1}{2} < x < -\frac{1}{4}$，则 x_6 必须为1，即 $x > -\frac{1}{2}$，且 x_5 必须为0，否则将不满足 $x < -\frac{1}{4}$。

~~3.(1) $[x]_原$ = 1001 1111~~

~~x = -11111B = -31~~

~~(2) $[x]_原$ = 10000000~~

~~= 10000000~~

~~x = -128~~

~~(3) $[x]_原$ = $[x]_补$~~

~~$x = 2^7 - 1 =$~~

~~(4)~~

3.(1) $[x]_原$ = 1001 1111 ∴ x = -11111B = -31

(2) $x = -2^7 = -128$

(3) $[x]_原 = [x]_补$ = 0111,1111 ∴ $x = 2^7 - 1 = 127$

4、 $x-y=-238$.

二进制表示为 -11101110

z为short类型，共16位

其原码表示为 10000000 11101110

其补码表示为 11111111 00010010.

5. $i=65535$

si的机器码为 1111111111111111

si以补码形式编码，其真值为 -1

int j = si 将对j的高位进行符号位扩展可为32个1

故j的真值仍为-1.

~~8. (1). 不采用任何附加位~~

$\cancel{\frac{15}{16} = \frac{1111B}{10000B}}$ $\frac{2^3+2^2+2^1+2^0}{2^4} = 2^{-1}+2^{-2}+2^{-3}+2^{-4}$

$\cancel{= 0.1 + 0.01 + 0.001 + 0.0001 = 0.1111B}$

$\cancel{= 2^4+2^3+2^1+2^0 = 11011B}$

$\cancel{\frac{15}{16}\times 2 = (\frac{15}{16})\times 2^4 + (\frac{15}{16})\times 2^3 + (\frac{15}{16})\times 2^1} + (\frac{15}{16})\times 2^0$

$\cancel{= 1111B + 111.1B} + 1.111B + 0.1111B$

先右移将除最高8位的位清空，再左移回去补0

6. (1). x >> (n-8) << (n-8).

(2). 与(1)同理.

x << (n-8) >> (n-8).　或 x & 0xFF.

(3) (~x) >> 8 << 8.

(4). x | 0xFF　通过或操作将低8位变为1.

7. (1) 永真。IEEE 754 标准中符号位和数值部分分开运算，溢出不影响乘积、符号

(2) 不永真。int 转为 ~~double~~ float 类型时 (32 > 24) 会造成有效位数丢失，即左边部分为近似值，而右边的 dx 为 int 转 double 得到不会发生精度损失，为精确值。

(3) 不永真。右边 x+y 在转为 double 之间可能会溢出，而左边 double 由两个 int 相加得到而不会溢出。

(4) 永真。这 3 个 double 类型的数均由 int 转换得到，而 double 的尾数位数为 53，即使不计入保护位 也不会在对阶时发生溢出。

(5) 永真。相乘顺序的不同会导致舍入的不同。

(6) 不永真。0/0 为 NaN 而不为 ~~非零~~ 相除的结果 1

8. 无附加位：

(1) ① $\frac{15}{16} \times 2^7 = \frac{2^3+2^2+2^1+2^0}{2^4} = (2^{-1}+2^{-2}+2^{-3}+2^{-4}) \times 2^7$

$= 0.1111B \times 2^7$ 浮点表示为 00. $\underset{\text{阶码}}{\underline{1111}}$ $\underset{\text{尾数}}{\underline{1111}}$

阶码 $7+8=15=1111B$

② $\frac{2}{16} \times 2^5 = 0.001B \times 2^5 = 0.1B \times 2^3$. 阶码 $3+8=11$ $=1011B$.

浮点表示为 00. $\underline{1011}$ $\underline{1000}$

③ 对阶：$[\Delta E]_{补} = 1111 - 1011 = 0100$ 即 $\Delta E = 4$.

④ 将 B 的尾数右移 4 位得到 $\underset{(偶数)}{\underline{0000}}$ 阶码变为 $\underline{1111}$.

尾数相加得到 1111

即 $\frac{15}{16} \times 2^7 + \frac{2}{16} \times 2^5 = 00.\ \underline{1111}\ \underline{1111} = \frac{15}{16} \times 2^7$.

(2). 与 (1) 基本相同，尾数相减仍为 1111

即 $\frac{15}{16} \times 2^7 - \frac{2}{16} \times 2^5 = 00.\ \underline{1111}\ \underline{1111} = \frac{15}{16} \times 2^7$.

(3) ① $\frac{15}{16} \times 2^5 = 0.1111B \times 2^5$. 浮点表示为 00. $\underline{1101}$ $\underline{1111}$

② $\frac{2}{16} \times 2^7 = 0.001B \times 2^7 = 0.1B \times 2^5$ 阶码 $5+8=11=1101B$.

浮点表示为 00. $\underline{1101}$ $\underline{1000}$

③ 对阶：$[\Delta E]_{补} = 1101 - 1101 = 0000$ 无需对阶 直接尾数相加

④. 尾数相加：1111 + 1000 = 1,0111 尾数溢出进行右规.

阶码加1变为 1110

尾数~~右规~~舍入采用就近舍入到偶数为 1100

即 $\frac{15}{16}\times2^5+\frac{2}{16}\times2^7=00.\underline{1110}\ \underline{1100}=\frac{12}{16}\times2^6=\frac{3}{4}\times2^6$.

(4) 与 (3) 基本相同，尾数相减：1111 − 1000 = 0111 需左规.

阶码减1变为 1100

尾数左规得到 1110.

即 $\frac{15}{16}\times2^5+\frac{2}{16}\times2^7=00.\underline{1100}\ \underline{1110}=\frac{14}{16}\times2^4=\frac{7}{8}\times2^4$.

采用2位附加位（基本信息与上述一致且这里序号接上面通用）

(1). ΔE = 4

对阶时对②的尾数右移4位得到 0000$\underline{\underline{10}}$ 附加位.

尾数相加为 1111$\underline{\underline{00}}$ + 0000$\underline{\underline{10}}$ = 1111$\underline{\underline{10}}$

对尾数附加位进行舍入，10为中间值强迫结果为偶数，

得尾数 $\underline{\underline{01}}$.0000，右规得到 00.1000

→ 尾数符号位

阶码 1111 + 1 产生阶码上溢 1|0000 故结果溢出.

(2). ΔE = 4 ~~对阶时对②的尾数~~ 对阶与(1)相同.

尾数相减 1111$\underline{\underline{00}}$ − 0000$\underline{\underline{10}}$ = 1110$\underline{\underline{10}}$.

对尾数附加位进行舍入，10为中间值，末位为偶数故舍去

得尾数 1110（与无附加位的差异）

结果为 $00.\underline{1111}\ \underline{1110}=\frac{14}{16}\times2^7=\frac{7}{8}\times2^7$.

(3). ΔE = 0. 尾数相加. 00.1111$\underline{\underline{00}}$ + 00.1000$\underline{\underline{00}}$ = 01.011100 尾数溢出右规.

阶码加1得 1110

尾数采用就近舍入到偶数为 $\underline{1011}$100 中间值 → .1100

结果为 $00.\underline{1110}\ \underline{1100}$. 与无附加位结果相同.

(4) ΔE = 0 尾数相减 111100 − 100000 = .011100. 尾数需左规.

阶码减1得 1100.

尾数为 1110

结果为 $00.\underline{1100}\ \underline{1110}$. 与无附加位结果相同