

3.

- 1) $(25.8125)_{10} = (11001.1101)_2 = (31.64)_8 = (19.D)_{16}$
- 2) $(101101.011)_2 = (45.375)_{10} = (55.3)_8 = (2D.6)_{16} = (0100\ 0101.0011\ 0111\ 0101)_{8421}$
- 3) $(0101\ 1001\ 0110.0011)_{8421} = (596.3)_{10} = (1001010100.0100110011001...)_{2} = (254.4CCC...)_{16}$
- 4) $(4E.C)_{16} = (78.75)_{10} = (1001110.11)_2$

9.

- 1) $x = 0xffff8000$
- 2) $y = 0x020a$
- 3) $z = 0x0000ffa$
- 4) $c = 0x40$
- 5) $f = 0xbf8cccd$
- 6) $d = 0x4025000000000000$

10.

- 1) $x = -65530$
- 2) $y = -8196$
- 3) $z = 4294967290$
- 4) $c = *$
- 5) $a = -800$
- 6) $b = -10.25$

15.

x	y	$x \wedge y$	$x \& y$	$x y$	$\sim x \sim y$	$x \& !y$	$x \& \& y$	$x y$	$!x !y$	$x \& \& \sim y$
0x5f	0xa0	0xff	0x0	0xff	0xff	0x0	0x1	0x1	0x0	0x1
0xc7	0xf0	0x37	0xc0	0xf7	0x3f	0x0	0x1	0x1	0x0	0x1
0x80	0x7f	0xff	0x0	0xff	0xff	0x0	0x1	0x1	0x0	0x1
0x07	0x55	0x52	0x5	0x57	0xfa	0x0	0x1	0x1	0x0	0x1

21.

M = 15;

N = 4;

29.

表示	X	x	Y	y	X+Y	x+y	OF	SF	CF	X-Y	x-y	OF	SF	CF
无符号	0xB0	176	0x8C	140	0x3C	60	1	0	1	0x24	36	0	0	0
带符号	0xB0	-80	0x8C	-116	0x3C	60	1	0	1	0x24	36	0	0	0
无符号	0x7E	126	0x5D	93	0xDB	219	1	1	0	0x21	33	0	0	0
带符号	0x7E	126	0x5D	93	0xDB	-37	1	1	0	0x21	33	0	0	0

①无符号 $0xB0+0x8C=0x13C \rightarrow 0x3C$ (高位截断), 最高位有进位, 故 $CF=1$. 结果最高位为 0, 故 $SF=0$. OF 参考带符号加法②中的解释.

$0xB0-0x8C=0x24$, 没有借位, $CF=0$. 结果可以用 8 位表示, $OF=0$. 结果最高位为 0, $SF=0$.

②带符号 $-80+(-116)=-316$, 8 位无法表示故结果溢出, $OF=1$, 补码运算最高位有进位 (同无符号加法①) 故 $CF=1$. 结果最高位为 0, 故 $SF=0$.

$-80-(-116)=36$, 结果可以用 8 位表示故没有溢出, $OF=0$. 没有产生借位故 $CF=0$. 结果最高位为 0, 故 $SF=0$.

③无符号 $126+93=219$, 结果可以用 8 位表示, 故没有产生进位, $CF=0$. 结果最高位为 1, 故 $SF=1$. OF 参考带符号加法④中的解释.

$126-93=33$, 结果可以用 8 位表示, 没有发生借位故 $CF=0$, $OF=0$. 结果最高位为 0, 故 $SF=0$.

④带符号 $126+93=219 \rightarrow -37$, 结果无法用 8 位表示, 发生溢出故 $OF=1$. 结果最高位为 1, 故 $SF=1$. 最高位没有产生进位故 $CF=0$.

$126-93=33$, 结果可用 8 位表示, 故没有发生溢出, $OF=0$. 最高位没有发生借位, 故 $CF=0$. 结果最高位为 0 故 $SF=0$.

33.

```
1  int div32(int x) {
2      return (x + ((x >> 31) & 0x1f)) >> 5;
3  }
```

40.

```
1  float fpower2(int x) {
2      unsigned exp, frac, u;
3
4      if (x < -149) { //值太小, 返回 0.0
5          exp = 0;
6          frac = 0;
7      }
8      else if (x < -126) { //返回非规格化结果
9          exp = 0;
10         frac = 0x400000 >> (-x - 127);
11     }
12     else if (x < 128) { //返回规格化结果
13         exp = x + 127;
14         frac = 0;
15     }
16     else { //值太大, 返回 + ∞
17         exp = 255;
18         frac = 0;
19     }
20     u = exp << 23 | frac;
21     return u2f(u);
22 }
```