Форматы чисел

Михаил Шихов
m.m.shihov@gmail.com

Лекция по дисциплине «информатика»
(1 марта 2017 г.)

Содержание

«Как?» VS «Почему?»

В отношении кодов лекция дает ответ только на вопрос «Как?». Ответ на вопрос «Почему?» следует искать в пособии и презентациях по дискретной математике (прим. авт.).

Назначение кодов

Назначение кодов — представить число в виде двоичной последовательности[a].

[a] В общем случае — в виде последовательности символов конечного алфавита

Далее, в контексте представления целых ($\mathbb{Z}$) чисел, рассматриваются три кода:

- прямой код;
- дополнительный код;
- обратный код.

Разрядная сетка

Последовательность длиной в n символов будем называть n-разрядной сеткой

Нумеровать разряды будем с нуля, справа-налево:

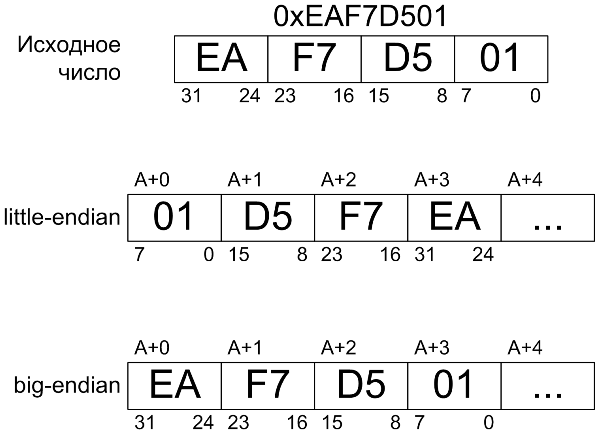
n-1	0
xxxxx...	...xxxxx

Двоичные коды

Далее, если не оговорено иное, будут использоваться *двоичные* коды.

Особенности представления многобайтовых данных

Порядок байт в памяти: little-endian (Интеловский) vs big-endian (сетевой)



Прямой код

Построение прямого кода числа x

Старший разряд прямого кода называют «знаковым». Знаковый разряд содержит 0, если $x \geq 0$, и 1, если $x < 0$. Т.е. знак «+» кодируется нулем, а «-» — единицей:

$$Sign(x) = \begin{cases} 0, & \text{если } x \geq 0, \\ 1, & \text{если } x < 0. \end{cases}$$

Остальные разряды содержат двоичное представление $|x|$:

n-1	0
$Sign(x)$	Двоичное представление $\lvert x \rvert$

Прямой код

Примеры кодирования в 4-х разрядной сетке

- $\overset{3}{\boxed{0}}\overset{2\quad\ 0}{\boxed{011}} \Leftrightarrow 3;$
- $\overset{3}{\boxed{1}}\overset{2\quad\ 0}{\boxed{101}} \Leftrightarrow -5;$
- $\overset{3}{\boxed{0}}\overset{2\quad\ 0}{\boxed{000}} \Leftrightarrow 0;$
- $\overset{3}{\boxed{1}}\overset{2\quad\ 0}{\boxed{000}} \Leftrightarrow$ запрещенный здравым смыслом $-0;$

Прямой код
Особенности

- Любим студентами «за простоту».
- Имеется запрещенная комбинация. Знаковый разряд 1, а представление модуля содержит нули:

n-1	0
1	$0 \cdots 0$

- Арифметическое сложение прямых кодов не имеет смысла.
- В n-разрядной сетке можно представить целые числа:
  - Минимальное: $-(2^{n-1}-1)$;
  - Максимальное: $(2^{n-1}-1)$.

Дополнительный код

Построение дополнительного кода числа x

Дополнительный код числа x будем обозначать ДК(x) и находить как

$$\text{ДК}(x) = \begin{cases} \overline{|x|} + 1, & \text{если } x < 0, \\ |x|, & \text{если } x \geq 0, \end{cases}$$

где $\overline{|x|}$ — инверсия бит в двоичном представлении модуля числа x.

При таком подходе[a] старший разряд кода можно считать знаковым: он как и в прямом коде будет содержать 0, если число положительно и 1 в противном случае.

[a] Если количества разрядов сетки достаточно для представления

Дополнительный код

Примеры кодирования в 4-х разрядной сетке

- $3 \Rightarrow \begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 0 & 011 \end{array}$
- $-5 \Rightarrow \begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 1 & 011 \end{array}$
- $-6 \Rightarrow \begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 1 & 010 \end{array}$
- $-8 \Rightarrow \begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 1 & 000 \end{array}$
- $\boxed{-9 \not\Rightarrow \begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 0 & 111 \end{array}}$ — за пределами представления в 4-х разрядах

Дополнительный код

Декодирование числа из n-разрядного кода ДК(x)

Если старший разряд кода — единица, то знак числа отрицательный, в противном случае — положительный.

Двоичное представление модуля числа извлекается по схожим правилам:

$$|x| = \begin{cases} \overline{\text{ДК}(x)} + 1, & \text{если в знаковом разряде кода } 1, \\ \text{ДК}(x), & \text{если в знаковом разряде кода } 0. \end{cases}$$

Дополнительный код

Примеры декодирования в 4-х разрядной сетке

- $\begin{array}{|c|c|} \multicolumn{1}{c}{\scriptstyle 3} & \multicolumn{1}{c}{\scriptstyle 2 \qquad 0} \\ \hline 0 & 010 \\ \hline \end{array} \Rightarrow 2$
- $\begin{array}{|c|c|} \multicolumn{1}{c}{\scriptstyle 3} & \multicolumn{1}{c}{\scriptstyle 2 \qquad 0} \\ \hline 1 & 101 \\ \hline \end{array} \Rightarrow -3$
- $\begin{array}{|c|c|} \multicolumn{1}{c}{\scriptstyle 3} & \multicolumn{1}{c}{\scriptstyle 2 \qquad 0} \\ \hline 1 & 110 \\ \hline \end{array} \Rightarrow -2$
- $\begin{array}{|c|c|} \multicolumn{1}{c}{\scriptstyle 3} & \multicolumn{1}{c}{\scriptstyle 2 \qquad 0} \\ \hline 0 & 111 \\ \hline \end{array} \Rightarrow 7$
- $\begin{array}{|c|c|} \multicolumn{1}{c}{\scriptstyle 3} & \multicolumn{1}{c}{\scriptstyle 2 \qquad 0} \\ \hline 1 & 111 \\ \hline \end{array} \Rightarrow -1$

Дополнительный код

Характерные дополнительные коды чисел в n-разрядной сетке

- $\text{ДК}(-1) = \underbrace{11\cdots11}_{n}$
- $\text{ДК}(-2^{n-1}) = \underbrace{10\cdots00}_{n}$
- $\text{ДК}(2^{n-1} - 1) = \underbrace{01\cdots11}_{n}$
- $\text{ДК}(-2) = \underbrace{11\cdots10}_{n}$
- $\text{ДК}(0) = \underbrace{00\cdots00}_{n}$
- $\text{ДК}(1) = \underbrace{00\cdots01}_{n}$

Дополнительный код
Особенности

- Запрещенных комбинаций нет.
- На практике в целочисленной арифметике используется в 100% случаев.
- Арифметическое сложение дополнительных кодов дает дополнетельный код результата или переполнение разрядной сетки.
- В n-разрядной сетке можно представить целые числа:
  - Минимальное: -2^{n-1};
  - Максимальное: $(2^{n-1} - 1)$.

Дополнительный код

Сложение дополнительных кодов

В результате сложения дополнительных кодов операндов, получается дополнительный код результата или разрядная сетка переполняется. Признаком переполнения может служить следующий признак:

если складывались дополнительные коды операндов одного знака, а получился результат противоположного знака — призошло ПРС[a]

[a] Переполнение Разрядной Сетки

Дополнительный код

Примеры сложения в 4-х разрядной сетке

Корректные результаты:

$2+3=5$

$$\begin{array}{r} 0010 \\ +\ 0011 \\ \hline 0101 \end{array}$$

$-5-3=-8$

$$\begin{array}{r} 1011 \\ +\ 1101 \\ \hline 1000 \end{array}$$

$-1+6=5$

$$\begin{array}{r} 1111 \\ +\ 0110 \\ \hline 0101 \end{array}$$

$3-6=-3$

$$\begin{array}{r} 0011 \\ +\ 1010 \\ \hline 1101 \end{array}$$

ПРС:

$5+7 \neq -4$

$$\begin{array}{r} 0101 \\ +\ 0111 \\ \hline 1100 \end{array}$$

$5+3 \neq -8$

$$\begin{array}{r} 0101 \\ +\ 0011 \\ \hline 1000 \end{array}$$

$-3-7 \neq 6$

$$\begin{array}{r} 1101 \\ +\ 1001 \\ \hline 0110 \end{array}$$

$-8-1 \neq 7$

$$\begin{array}{r} 1000 \\ +\ 1111 \\ \hline 0111 \end{array}$$

Дополнительный код

Модифицированный дополнительный код

Видно, что для того, чтобы устранить ПРС, к исходной *n*-разрядной сетке достаточно добавить слева один разряд. При этом диапазон представления исходных операндов остается прежним:$[-2^{n-1}, (2^{n-1} - 1)]$.

В *модифицированных* кодах под знаковый разряд выделяется два разряда.

И если у результата в знаковых разрядах получилась комбинация, отличная от 00 или 11, то произошло ПРС[1].

О ПРС теперь можно судить, не зная знаки исходных операндов.

[1]Хотя в $(n+1)$-разрядной сетке (в немодифицированном $(n+1)$-разрядном коде) результат корректен

Модифицированный дополнительный код

Примеры сложения в модифицированной 4-х разрядной сетке

Корректные результаты (комбинации знаков 00 , 11):

$2+3=5$

$$\begin{array}{r} 00010 \\ +\ 00011 \\ \hline 00101 \end{array}$$

$-5-3=-8$

$$\begin{array}{r} 11011 \\ +\ 11101 \\ \hline 11000 \end{array}$$

$-1+6=5$

$$\begin{array}{r} 11111 \\ +\ 00110 \\ \hline 00101 \end{array}$$

$3-6=-3$

$$\begin{array}{r} 00011 \\ +\ 11010 \\ \hline 11101 \end{array}$$

ПРС (комбинации знаков 01 , 10):

$5+7\neq 12$

$$\begin{array}{r} 00101 \\ +\ 00111 \\ \hline 01100 \end{array}$$

$5+3\neq 8$

$$\begin{array}{r} 00101 \\ +\ 00011 \\ \hline 01000 \end{array}$$

$-3-7\neq -10$

$$\begin{array}{r} 11101 \\ +\ 11001 \\ \hline 10110 \end{array}$$

$-8-1\neq -9$

$$\begin{array}{r} 11000 \\ +\ 11111 \\ \hline 10111 \end{array}$$

Сдвиг дополнительного кода

При сдвиге дополнительного кода числа *вправо*, освобождающийся старший разряд (msb) должен заполняться своим значением до сдвига (т.к. знак числа не должен меняться). Это не всегда приводит к ожидаемому уменьшению (отрицательного) числа вдвое:

$shl(10) = \overrightarrow{001010} = 000101 = 5$	$shl(-10) = \overrightarrow{110110} = 111011 = -5$
$shl(9) = \overrightarrow{001001} = 000100 = 4$	$shl(-9) = \overrightarrow{110111} = 111011 = -5$
$shl(1) = \overrightarrow{000001} = 00000 = 0$	$shl(-1) = \overrightarrow{111111} = 111111 = -1$

Если требуется, чтобы при сдвиге кода вправо модуль представляемого числа уменьшался вдвое, то к результату после сдвига нужно прибавить ($lsb \wedge msb$) до сдвига.

При сдвиге дополнительного кода *влево* освобождающийся младший разряд заполняется нулём. Это (при отсутствии ПРС) приводит к увеличению числа вдвое.

Дополнительный код

Естественность представления отрицательных чисел

$$
\begin{aligned}
-52 &= 2\cdot -26 + 0; & \Rightarrow a_0 &= 0;\\
-26 &= 2\cdot -13 + 0; & \Rightarrow a_1 &= 0;\\
-13 &= 2\cdot -7 + 1; & \Rightarrow a_2 &= 1;\\
-7 &= 2\cdot -4 + 1; & \Rightarrow a_3 &= 1;\\
-4 &= 2\cdot -2 + 0; & \Rightarrow a_4 &= 0;\\
-2 &= 2\cdot -1 + 0; & \Rightarrow a_5 &= 0;\\
-1 &= 2\cdot -1 + 1; & \Rightarrow a_6 &= 1;\\
-1 &= 2\cdot -1 + 1; & \Rightarrow a_7 &= 1;\\
&\dots & &\dots
\end{aligned}
$$

$$-52 = (1\cdots 11001100)_2.$$

Обратный код

Построение обратного кода числа x

Обратный код числа x будем обозначать $\text{ОК}(x)$ и находить как

$$\text{ОК}(x) = \begin{cases} \overline{|x|}, & \text{если } x < 0, \\ |x|, & \text{если } x \geq 0, \end{cases}$$

где $\overline{|x|}$ — инверсия бит в двоичном представлении модуля числа x.

При таком подходе[a] старший разряд кода можно считать знаковым: он как и в прямом коде будет содержать 0, если число положительно и 1 в противном случае.

[a] Если количества разрядов сетки достаточно для представления

Обратный код

Декодирование числа из n-разрядного кода $\mathrm{OK}(x)$

Если старший разряд кода — единица, то знак числа отрицательный, в противном случае — положительный.
Модуль числа:

$$|x| = \begin{cases} \overline{\mathrm{OK}(x)}, & \text{если в знаковом разряде кода 1,} \\ \mathrm{OK}(x), & \text{если в знаковом разряде кода 0.} \end{cases}$$

Обратный код

Примеры кодирования-декодирования в 4-х разрядной сетке

- $\begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 0 & 011 \end{array} \Leftrightarrow 3;$
- $\begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 1 & 010 \end{array} \Leftrightarrow -5;$
- $\begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 1 & 001 \end{array} \Leftrightarrow -6;$
- $\begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 0 & 000 \end{array} \Leftrightarrow 0;$
- $\begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 1 & 000 \end{array} \Leftrightarrow -7;$
- $\begin{array}{c|c} 3 & 2 \quad 0 \\ \hline 1 & 111 \end{array}$ — запрещенный здравым смыслом -0.

Обратный код

Особенности

- Любим «за простоту» получения, ненавидим «за необходимость коррекции».
- Запрещенной комбинацией являются единицы во всех разрядах кода.
- Арифметическое сложение обратных кодов дает:
  - правильный обратный код результата;
  - обратный код результата, *требующий коррекции*;
  - переполнение разрядной сетки.
- В n-разрядной сетке можно представить целые числа:
  - Минимальное: $-(2^{n-1} - 1)$;
  - Максимальное: $(2^{n-1} - 1)$.

Сложение обратных кодов

Алгоритм проверки результата сложения обратных кодов следующий.

1. Если требуется, выполнить коррекцию результата.

Признак: единица переноса из старшего разряда.
Коррекция: прибавить 1 к младшему разряду кода результата.

2. Если требуется, зафиксировать ПРС и выйти.

Признак: складывались обратные коды операндов одного знака, а получился код противоположного знака.

3. Если требуется, скорректировать $(-0 \mapsto 0)$.

Коррекция: $(11\cdots11)_2 \mapsto (00\cdots00)_2$.

4. Фиксировать правильный результат.

Обратный код

Примеры сложения в 4-х разрядной сетке

Корректные результаты:

$2+3=5$

$$\begin{array}{r} 0010 \\ +\ 0011 \\ \hline 0101 \end{array}$$

$-5-2=-7$

$$\begin{array}{r} 1010 \\ +\ 1101 \\ \hline 0111 \end{array}$$

Коррекция:

$$\begin{array}{r} 0111 \\ +\ 0001 \\ \hline 1000 \end{array}$$

$-1+6=5$

$$\begin{array}{r} 1110 \\ +\ 0110 \\ \hline 0100 \end{array}$$

Коррекция:

$$\begin{array}{r} 0100 \\ +\ 0001 \\ \hline 0101 \end{array}$$

$3-6=-3$

$$\begin{array}{r} 0011 \\ +\ 1001 \\ \hline 1100 \end{array}$$

Обратный код

Примеры сложения в 4-х разрядной сетке

Ноль:

$2 - 2 = 0$

$$\begin{array}{r} 0010 \\ +\;\underline{1101} \\ 1111 \end{array}$$

Замена $-0 \mapsto 0$:

$$0000$$

Обратный код

Примеры сложения в 4-х разрядной сетке

ПРС:

$5+7 \neq -3$

$$\begin{array}{r} 0101 \\ +\ 0111 \\ \hline 1100 \end{array}$$

$5+3 \neq -7$

$$\begin{array}{r} 0101 \\ +\ 0011 \\ \hline 1000 \end{array}$$

$-3-7 \neq 5$

$$\begin{array}{r} 1100 \\ +\ 1000 \\ \hline 0100 \end{array}$$

Коррекция:

$$\begin{array}{r} 0100 \\ +\ 0001 \\ \hline 0101 \end{array}$$

$-5-3 \neq 7$

$$\begin{array}{r} 1010 \\ +\ 1100 \\ \hline 0110 \end{array}$$

Коррекция:

$$\begin{array}{r} 0110 \\ +\ 0001 \\ \hline 0111 \end{array}$$

Обратный код

Модифицированный обратный код

Обоснование модификации такое же, как и в дополнительном коде. Если исходная сетка n-разрядная, а модифицированная $(n+1)$-разрядная, то диапазон представления исходных операндов остается прежним:$[-(2^{n-1}-1),(2^{n-1}-1)]$.

В *модифицированном* коде знак кодируется двумя разрядами.

И если у результата в знаковых разрядах получилась комбинация, отличная от 00 или 11 , то произошло ПРС. Алгоритм сложения остается прежним.

О ПРС можно судить, не зная знаки исходных операндов.

Модифицированный обратный код

Примеры сложения в модифицированной 4-х разрядной сетке

Корректные результаты:

$2+3=5$

$$\begin{array}{r} 00010 \\ +\ 00011 \\ \hline 00101 \end{array}$$

$-5-2=-7$

$$\begin{array}{r} 11010 \\ +\ 11101 \\ \hline 10111 \end{array}$$

Коррекция:

$$\begin{array}{r} 10111 \\ +\ 10001 \\ \hline 11000 \end{array}$$

$-1+6=5$

$$\begin{array}{r} 11110 \\ +\ 00110 \\ \hline 00100 \end{array}$$

Коррекция:

$$\begin{array}{r} 00100 \\ +\ 00001 \\ \hline 00101 \end{array}$$

$3-6=-3$

$$\begin{array}{r} 00011 \\ +\ 11001 \\ \hline 11100 \end{array}$$

Модифицированный обратный код

Примеры сложения в модифицированной 4-х разрядной сетке

Ноль:

$2 - 2 = 0$

$$
\begin{array}{r}
 & 00010 \\
+ & \\
 & \underline{11101} \\
 & 11111
\end{array}
$$

Замена $-0 \mapsto 0$:

00000

Модифицированный обратный код

Примеры сложения в модифицированной 4-х разрядной сетке

ПРС:

$5+7 \not\equiv 12$

$$\begin{array}{r} 00101 \\ +\ 00111 \\ \hline 01100 \end{array}$$

$5+3 \not\equiv 8$

$$\begin{array}{r} 00101 \\ +\ 00011 \\ \hline 01000 \end{array}$$

$-3-7 \not\equiv -10$

$$\begin{array}{r} 11100 \\ +\ 11000 \\ \hline 10100 \end{array}$$

Коррекция:

$$\begin{array}{r} 10100 \\ +\ 00001 \\ \hline 10101 \end{array}$$

$-5-3 \not\equiv -8$

$$\begin{array}{r} 11010 \\ +\ 11100 \\ \hline 10110 \end{array}$$

Коррекция:

$$\begin{array}{r} 10110 \\ +\ 00001 \\ \hline 10111 \end{array}$$

Сдвиг обратного кода

При сдвиге обратного кода числа *вправо*, освобождающийся старший разряд (*msb*) должен заполняться своим значением до сдвига:

$shl(10) = \overrightarrow{001010} = 000101 = 5$	$shl(-10) = \overrightarrow{110101} = 111010 = -5$
$shl(9) = \overrightarrow{001001} = 000100 = 4$	$shl(-9) = \overrightarrow{110110} = 111011 = -4$
$shl(1) = \overrightarrow{000001} = 00000 = 0$	$shl(-1) = \overrightarrow{111110} = 111111 = -0$

При сдвиге обратного кода *влево* освобождающийся младший разряд заполняется знаковым (*msb*) разрядом. Это (при отсутствии ПРС) приводит к увеличению числа вдвое.

$shr(10) = \overleftarrow{001010} = 010100 = 20$	$shr(-10) = \overleftarrow{110101} = 101011 = -20$
$shr(9) = \overleftarrow{001001} = 010010 = 18$	$shr(-9) = \overleftarrow{110110} = 101101 = -18$
$shr(1) = \overleftarrow{000001} = 000010 = 2$	$shr(-1) = \overleftarrow{111110} = 111101 = -2$

Обратный и дополнительный коды в любой СС

Основание: k. Цифра: x_i. Дополнение $\overline{x_i} = (k - x_i - 1)$

Например, дополнительный код в десятичной СС (4 разряда):

$731 - 485 = 246$

$$\begin{array}{r} 0731 \\ +\ 9515 \\ \hline 0246 \end{array}$$

$204 - 690 = -486$

$$\begin{array}{r} 0204 \\ +\ 9310 \\ \hline 9514 \end{array}$$

$100 - 1000 = -900$

$$\begin{array}{r} 0100 \\ +\ 9000 \\ \hline 9100 \end{array}$$

Фиксированная точка

Для представления вещественного числа x:

- В n-разрядном формате с фиксированной точкой, разделитель целой и дробной части *мысленно* «фиксируется» между k и $(k-1)$ разрядами разрядной сетки.

$$\underbrace{\mathtt{xxxx}\cdots\mathtt{xxxx}}_{(n-k)}\,,\,\underbrace{\mathtt{xx}\cdots\mathtt{xx}}_{k}$$

- В старших $(n-k)$ разрядах сохраняется целая часть x, а в младших k разрядах — дробная.

Фиксированная точка
Масштабирование

- Для *удобства рассуждений*, договариваются о масштабирующем множителе $M = 2^m$ — *масштабе*. $m \in \mathbb{Z}$.
- Масштаб одинаков для всех чисел и не меняется.
- Исходное число x представляется в разрядной сетке некоторым кодом числа y. Исходное x из y получается по формуле:

$$x = y \cdot M.$$

- Компьютер, выполняя операции над представлениями y, о существовании масштаба «не догадывается».
- Выделяют два типа масштабирования:
  - дробное (мы будем использовать в лекциях для выкладок);
  - целое (как правило, используется программистами).

Масштабирование
Дробное

$$x = y \cdot M$$

При дробном масштабировании представление числа x, т.е.

y — дробное число с нулевой целой частью.

Рарзядная сетка хранит разряды дробной части y.

$$0.\underset{}{\overset{\mathbf{n\text{-}1} \qquad\qquad\qquad \mathbf{0}}{\boxed{\mathtt{yyyyy\cdots yyyyy}}}}$$

Чтобы зафиксировать запятую между k и $(k-1)$ разрядами, выбирается масштаб

$$M = 2^{(n-k)}.$$

Масштабирование
Целое

$$x = y \cdot M$$

При целом масштабировании представление числа x, т.е.

y — целое число.

Рарзядная сетка хранит разряды целой части y.

$$\begin{array}{c}\text{n-1} \qquad\qquad\qquad\qquad 0 \\ \boxed{\mathtt{yyyyy}\cdots\mathtt{yyyyy}}\end{array}\ .0$$

Чтобы зафиксировать запятую между k и $(k-1)$ разрядами, выбирается масштаб

$$M = 2^{-k}.$$

Масштабирование

Пример

Дано число

$$x = (10111.1011)_2.$$

- При дробном масштабировании в ДК(y) с масштабом $M = 2^{10}$ в 16-разрядной сетке ($y = (0.0000010111011)_2$):

15	0
0000010111101100	

- При целом масштабировании в ДК(y) с масштабом $M = 2^{-5}$ в 16-разрядной сетке ($y = (1011110110.0)_2$):

15	0
0000001011110110	

Масштабирование

Пример

Дано число

$$x = -(10111.1011)_2.$$

- При дробном масштабировании в ДК(y) с масштабом $M = 2^7$ в 16-разрядной сетке ($y = -(0.00101111011)_2$):

15	0
1101000010100000	

- При целом масштабировании в ДК(y) с масштабом $M = 2^{-6}$ в 16-разрядной сетке ($y = -(10111101100.0)_2$):

15	0
1111101000010100	

Фиксированная точка

Форматы, используемые на практике

На практике, на уровне команд процессора, форматы с фиксированной запятой используют:

- целочисленное масштабирование;
- дополнительный код для представления y;
- разрядности 1, 2, 4, 8 байт.

Фиксированная точка

Оценка погрешности

Используем целое масштабирование для представления числа x в n-разрядной сетке.

- Абсолютная погрешность Δ — половина цены деления. Цена деления y — 1. Цена деления $x = 1 \cdot M = 2^{-k}$.

$$\Delta = \frac{M}{2} = 2^{-(k+1)}.$$

- Для относительной погрешности $\delta = \frac{\Delta}{|x|}$ оценим диапазон.

$$\delta_{max} = \frac{\Delta}{|x|_{min}} = \frac{2^{-(k+1)}}{1 \cdot M} = \frac{2^{-(k+1)}}{2^{-k}} = \frac{1}{2}$$

$$\delta_{min} = \frac{\Delta}{|x|_{max}} = \frac{2^{-(k+1)}}{2^{n-1} \cdot M} = \frac{2^{-(k+1)}}{2^{n-1} \cdot 2^{-k}} = \frac{1}{2^n}$$

Фиксированная точка

Оценка погрешности — выводы

- Абсолютная погрешность — константа $\Delta = 2^{-(k+1)}$.
- Относительная погрешность δ изменяется от чудовищной 50% до ничтожной $\frac{100}{2^n}\%$.

Плавающая точка

В форматах с плавающей точкой, в системе счисления с основанием k, вещественное число x представляется следующим образом:

$$x = m_x \cdot k^{p_x},$$

где

- m_x — *мантисса* числа x;
- p_x — *порядок* числа x.

При этом мантисса m_x обязательно нормализуется.

Плавающая точка

Правила нормализации

$$x = m_x \cdot k^{p_x}$$

Умножение на k^{p_x} приводит к переносу точки в m_x

- влево, если $p_x < 0$;
- вправо, если $p_x > 0$.

Порядок числа p_x подбирается так, чтобы мантисса была *нормализованной*.

Нормализованная мантисса m_x представляет собой дробное число, старший (-1-й) разряд которого содержит ненулевую цифру.

Плавающая точка

Порядок. Описание формата для примера

15	14 6	5	4 0
X	XXXXXXXXX	X	XXXXX

Мантисса и порядок представляются в прямом коде. Модуль мантиссы представлен в рязрядах [14 : 6], знак мантиссы в 15-м разряде. Модуль порядка представлен в разрядах [4 : 0], знак порядка в 5-м разряде.

Плавающая точка

Порядок. Пример

Представить число -78.4453125.

$$-78.4453125 = (-1001110.0111001)_2.$$

Нормализованное представление:

$$-78.4453125 = (-0.10011100111001)_2 \cdot 2^{(+111)_2}.$$

Результат:

15	14 … 6	5	4 … 0
1	100111001	0	00111

Плавающая точка
Порядок. Пример

Представить число 0.00537109375.

$$0.00537109375 = (0.00000001011)_2.$$

Нормализованное представление:

$$0.00537109375 = (0.1011)_2 \cdot 2^{(-111)_2}.$$

Результат:

15	14 … 6	5	4 … 0
0	101100000	1	00111

Плавающая точка
Характеристика

В машинных форматах, применяемых на практике, вместо порядка используют *смещённый* порядок — *характеристику*. В отличие от порядка,

характеристика — всегда положительное число.

Чтобы получить характеристику c_x числа x, нужно к его порядку прибавить фиксированную константу Δ — смещение:

$$c_x = p_x + \Delta.$$

Тогда число, представленное в формате, будет определяться следующим образом:

$$x = m_x \cdot k^{(c_x - \Delta)}.$$

Плавающая точка

Характеристика

Смещение Δ выбирается исходя из количества разрядов, отведенных под представлние порядка (а также и характеристики). Допустим, что под представление порядка отведено n двоичных разрядов.
Если использовать дополнительный код, то диапазон представления порядков будет следующим:

$$[-2^{n-1}, (2^{n-1} - 1)].$$

Таким образом, смещение Δ для получения характеристики выбирается так, чтобы при сложении с наименьшим отрицательным числом диапазона представления порядка получался ноль.
Для дополнительного кода: $\Delta = 2^{n-1}$.

Плавающая точка
Характеристика

Использование характеристики вместо порядка дает на практике несколько преимуществ, одно из которых заключается в том, что положительные числа (характеристики) гораздо проще сравнивать друг с другом.

Плавающая точка

Характеристика. Описание формата для примера

15	14 … 6	5 … 0
X	XXXXXXXXX	XXXXXX

Мантисса представляется в прямом коде. Модуль мантиссы представлен в разрядах [14 : 6], знак мантиссы в 15-м разряде. Характеристика представлена в разрядах [5 : 0], смещение порядка $\Delta = 2^5$.

Плавающая точка
Характеристика. Пример

Представить число -78.4453125.

$$-78.4453125 = (-1001110.0111001)_2.$$

Нормализованное представление:

$$-78.4453125 = (-0.10011100111001)_2 \cdot 2^{(+111)_2}.$$

Результат:

15	14 ... 6	5 ... 0
1	100111001	100111

Плавающая точка

Характеристика. Пример

Представить число 0.00537109375.

$$0.00537109375 = (0.00000001011)_2.$$

Нормализованное представление:

$$0.00537109375 = (0.1011)_2 \cdot 2^{(-111)_2}.$$

Результат:

15	14 … 6	5 … 0
0	101100000	011001

Плавающая точка

Форматы, применяемые на практике: ЕС ЭВМ

ЕС ЭВМ. Используется три варианта формата: короткий (32 бита), длинный (64 бита), расширенный (128 бит). Во всех вариантах используются смешенные порядки (характеристики), занимающие 7 бит (смещение $\Delta = 64$). Старший бит формата содержит знак числа, затем следуют 7 бит характеристики, а оставшиеся разряды занимает модуль мантиссы. Мантисса изображается в 16-ичной системе счисления, т.е. каждые 4 бита мантиссы воспринимаются ЭВМ как шестнадцатеричная цифра. Т.е.

$$x = m_x \cdot 16^{(c_x - \Delta)}.$$

Мантисса нормализуется так, что после точки (запятой) следует ненулевая шестнадцатеричная цифра, а её целая часть равна нолю.

Плавающая точка

Форматы, применяемые на практике: ЕС ЭВМ

$$-78.4453125 = (-1001110.0111001)_2.$$

31	30 … 24	23 … 0
1	1000010	0100 1110 0111 0010 0000 0000

$$0.00537109375 = (0.00000001011)_2.$$

31	30 … 24	23 … 0
0	0111111	0001 0110 0000 0000 0000 0000

Плавающая точка

Форматы, применяемые на практике: СМ ЭВМ (ПЭВМ)

СМ ЭВМ. Используется два варианта формата: короткий (32 бита) и длинный (64 бита). Характеристика в обоих вариантах занимает 8 бит ($\Delta = 128$). Старший разряд отводится под знак числа, далее следуют 8 бит характеристики, а остальные разряды занимает модуль мантиссы. Характеристика отражает положение точки в двоичном представлении числа:

$$x = m_x \cdot 2^{(c_x - \Delta)}.$$

Плавающая точка

Форматы, применяемые на практике: СМ ЭВМ (ПЭВМ)

$$-78.4453125 = (-1001110.0111001)_2.$$

31	30 23	22 0
1	10000111	10011100111001000000000

$$0.00537109375 = (0.00000001011)_2.$$

31	30 23	22 0
0	01111001	01100000000000000000000

Плавающая точка
Оценка погрешности

$$x = m_x \cdot 2^{p_x}$$

- Абсолютная погрешность зависит от порядка числа:

$$\Delta = \frac{2^{-\|m_x\|}}{2} \cdot 2^{p_x} = \frac{2^{(p_x - \|m_x\|)}}{2}.$$

- Для относительной погрешности $\delta = \frac{\Delta}{|x|}$ оценим диапазон, который, как видно, от порядка числа не зависит.

$$\delta_{max} = \frac{\Delta}{2^{-1} \cdot 2^{p_x}} = 2^{-\|m_x\|}$$

$$\delta_{min} = \frac{\Delta}{(1 - 2^{-\|m_x\|}) \cdot 2^{p_x}} = \frac{2^{-\|m_x\|}}{2 \cdot (1 - 2^{-\|m_x\|})} \approx \frac{2^{-\|m_x\|}}{2}$$

Плавающая точка
Оценка погрешности — выводы

- Абсолютная погрешность зависит от порядка: чем он больше, тем больше и погрешность представления чисел.

$$\Delta = \frac{2^{(p_x - \|m_x\|)}}{2}$$

- Относительная погрешность δ — практически константа и равна (в худшем случае) вкладу младшего разряда мантиссы.

$$\delta = 2^{-\|m_x\|}$$

1)

Перевести число 115.43

$$10\text{СС} \to 8\text{СС} \to 2\text{СС} \to 16\text{СС} \to 10\text{СС}$$

выбрать необходимое количество знаков в дробной части[2]. Оценить абсолютную и относительную погрешности представления в 2СС.

---

[2] Исходя из соображений максимально точного приближения, при переводе дробной части следует использовать бесконечное количество знаков после запятой. Исходя же из экономии аппаратных затрат, следует взять необходимый минимум знаков после запятой, дающий не худшую точность представления. Если число представлено в СС с основанием k_1 и дробная часть состоит из n_1-й цифры, то при переводе в систему счисления с основанием k_2 следует взять n_2 знаков после запятой

$$n_2 \geq \left\lceil n_1 \frac{\ln k_1}{\ln k_2} \right\rceil .$$

2)

Какое число представлено в дополнительном коде в формате с фиксированной точкой

15	0
1101101000011101	

если:

1. использовалась дробная нормализация и масштаб 2^9;
2. использовалась целая нормализация и масштаб 2^{-6}.

Дать оценку минимальной абсолютной погрешности в обоих случаях.

3)

Какое число представлено в формате с плавающей точкой

15 … 6	5 … 0
1100100000	100011

15 … 6	5 … 0
0110110000	000111

если разряды $[15:6]$ — мантисса в прямом коде (нормализация: 0 в целой части, после точки — 1), разряды $[5:0]$ — порядок в прямом коде.

4)

Какое число представлено в формате с плавающей точкой

15 … 6	5 … 0
0101100000	100011

15 … 6	5 … 0
1101110000	011110

если разряды [15 : 6] — мантисса в прямом коде (нормализация: 0 в целой части, после точки — 1), разряды [5 : 0] — характеристика.

5)

Представить число $-0,01953125$ в коротком формате ЕС ЭВМ.

6)

Представить число 14.07421875 в коротком формате СМ ЭВМ.

7)

Сложить числа в 16-разрядной сетке в формате с фиксированной точкой (целое масштабирование, масштаб 2^{-6}).

1. $21\frac{18}{32}$ и $-37\frac{3}{16}$;
2. $-17\frac{11}{32}$ и $-43\frac{13}{16}$.

Выполнить проверку результата и оценить погрешности.

8)

Определим собственный формат с плавающей точкой. Для представления числа используется шестнадцать двоичных разрядов. Мантисса представляется в прямом коде. Модуль мантиссы представлен в рязрядах [15 : 6], знак мантиссы в 15-м разряде. Характеристика представлена в разрядах [5 : 0], смещение порядка $\Delta = 2^5$.

15	14 … 6	5 … 0
X	XXXXXXXXX	XXXXXX

Сложить перечисленные ниже пары чисел (изобразив необходимые действия в формате), выполнить проверку результата и оценить погрешности.

1. 125.75 и $\frac{5}{128}$;
2. 65 и $-\frac{6}{128}$;
3. 125.625 и -126.5.

1. Сложите десятичные числа в дополнительном коде, выберите масштаб и разрядную сетку самостоятельно.
   1. 975.48 и -729.503;
   2. -795.804 и 279.35;
   3. -795.034 и -729.13.
2. Сложите десятичные числа -975.48 и -729.53 в обратном коде, выберите масштаб и разрядную сетку самостоятельно.
   1. 482.07 и -195.053;
   2. -675.015 и 511.39;
   3. -851.11 и -163.509.
3. Как представить ноль в формате с плавающей запятой?
4. Когда нет необходимости выполнять сложение мантисс числа в формате с плавающей точкой.

1. Обоснуйте корректность декодирования дополнительного кода.
2. Доказать, что после коррекции результата сложения обратных кодов запрещенная комбинация «все единицы» возникнуть не может.
3. Доказать или опровергнуть, что в результате сложения обратных кодов комбинация «все нули» может получиться только вследствие замены комбинации «все единицы».
4. Чем «чревато» не запрещать комбинацию «все единицы» в обратном коде. Если считать комбинацию «все единицы» еще одним представлением нуля, повлечет ли это проблемы с определением ПРС или коррекцией?
5. Продумайте правила сложения в троичной симметричной системе счисления, признаки ПРС, модификацию кода?

Советы самоучке

Рекомендуемые книги, ставшие классикой: [1, 2].

Библиография I

- *Б.Г.Лысиков.* Арифметические и логические основы цифровых автоматов / Б.Г.Лысиков. —
  2 изд. —
  Мн.: Выш. школа, 1980. —
  336 с.
- *А.Я.Савельев.* Прикладная теория цифровых автоматов / А.Я.Савельев. —
  М.: Высшая школа, 1987. —
  272 с.