

Что такое система счисления?

Система счисления — это совокупность приемов и правил, по которым числа записываются и читаются.

Существуют позиционные и непозиционные системы счисления.

В непозиционных системах счисления вес цифры (т. е. тот вклад, который она вносит в значение числа) не зависит от ее позиции в записи числа. Так, в римской системе счисления в числе XXXII (тридцать два) вес цифры X в любой позиции равен просто десяти.

В позиционных системах счисления вес каждой цифры изменяется в зависимости от ее положения (позиции) в последовательности цифр, изображающих число. Например, в числе 757,7 первая семерка означает 7 сотен, вторая — 7 единиц, а третья — 7 десятых долей единицы.

Сама же запись числа 757,7 означает сокращенную запись выражения

$$700 + 50 + 7 + 0,7 = 7 \cdot 10^2 + 5 \cdot 10^1 + 7 \cdot 10^0 + 7 \cdot 10^{-1} = 757,7.$$

Любая позиционная система счисления характеризуется своим **основанием**.

Основание позиционной системы счисления — количество различных цифр, используемых для изображения чисел в данной системе счисления.

За основание системы можно принять любое натуральное число — два, три, четыре и т.д. Следовательно, **возможно бесчисленное множество позиционных систем**: двоичная, троичная, четверичная и т.д. Запись чисел в каждой из систем счисления с основанием q означает сокращенную запись выражения

$$a_{n-1} q^{n-1} + a_{n-2} q^{n-2} + \dots + a_1 q^1 + a_0 q^0 + a_{-1} q^{-1} + \dots + a_{-m} q^{-m},$$

где a_i — цифры системы счисления; n и m — число целых и дробных разрядов, соответственно.

Например:

Разряды	3	2	1	0	-1
Число	1	0	1	1	

$$1_2 = 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1};$$

Разряды	2	1	0	-1	-2
Число	2	7	6	5	2

$$2_8 = 2 \cdot 8^2 + 7 \cdot 8^1 + 6 \cdot 8^0 + 5 \cdot 8^{-1} + 2 \cdot 8^{-2};$$

4.2. Как порождаются целые числа в позиционных системах счисления?

В каждой системе счисления цифры упорядочены в соответствии с их значениями: 1 больше 0, 2 больше 1 и т.д.

Продвижением цифры называют замену её следующей по величине.

Продвинуть цифру 1 значит заменить её на 2, продвинуть цифру 2 значит заменить её на 3 и т.д. **Продвижение старшей цифры** (например, цифры 9 в десятичной системе) **означает замену её на 0**. В двоичной системе, использующей только две цифры — 0 и 1, продвижение 0 означает замену его на 1, а продвижение 1 — замену её на 0.

Целые числа в любой системе счисления порождаются с помощью **Правил счета**:

Для образования целого числа, следующего за любым данным целым числом, нужно продвинуть самую правую цифру числа; если какая-либо цифра после продвижения стала нулем, то нужно продвинуть цифру, стоящую слева от неё.

Применяя это правило, запишем первые десять целых чисел

- в двоичной системе: 0, 1, 10, 11, 100, 101, 110, 111, 1000, 1001;
- в троичной системе: 0, 1, 2, 10, 11, 12, 20, 21, 22, 100;
- в пятеричной системе: 0, 1, 2, 3, 4, 10, 11, 12, 13, 14;
- в восьмеричной системе: 0, 1, 2, 3, 4, 5, 6, 7, 10, 11.

4.3. Какие системы счисления используют специалисты для общения с компьютером?

Кроме десятичной широко используются системы с основанием, являющимся целой степенью числа 2, а именно:

- **двоичная** (используются цифры 0, 1);
- **восьмеричная** (используются цифры 0, 1, ..., 7);
- **шестнадцатеричная** (для первых целых чисел от нуля до девяти используются цифры 0, 1, ..., 9, а для следующих чисел — от десяти до пятнадцати — в качестве цифр используются символы A, B, C, D, E, F).

Полезно запомнить запись в этих системах счисления первых двух десятков целых чисел:

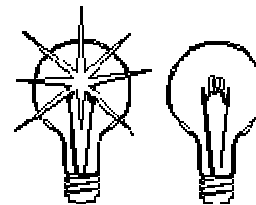
10-я	2-я	8-я	16-я
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9

10-я	2-я	8-я	16-я
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13

Из всех систем счисления **особенно проста** и поэтому **интересна для технической реализации в компьютерах двоичная система счисления**.

4.4. Почему люди пользуются десятичной системой, а компьютеры — двоичной?

Люди предпочитают десятичную систему, вероятно, потому, что с древних времен считали по пальцам, а пальцев у людей по десять на руках и ногах. Не всегда и не везде люди пользуются десятичной системой счисления. В Китае, например, долгое время пользовались пятеричной системой счисления.



А компьютеры используют двоичную систему потому, что она имеет ряд преимуществ перед другими системами:

- для ее реализации нужны **технические устройства с двумя устойчивыми состояниями** (есть ток — нет тока, намагничен — не намагничен и т.п.), а не, например, с десятью, — как в десятичной;
- представление информации посредством только двух состояний **надежно и помехоустойчиво**;
- возможно **применение аппарата булевой алгебры** для выполнения логических преобразований информации;
- двоичная арифметика намного проще десятичной.

Недостаток двоичной системы — **быстрый рост числа разрядов**, необходимых для записи чисел.

4.5. Почему в компьютерах используются также восьмеричная и шестнадцатеричная системы счисления?

Двоичная система, удобная для компьютеров, для человека неудобна из-за ее громоздкости и непривычной записи.

Перевод чисел из десятичной системы в двоичную и наоборот выполняет машина. Однако, чтобы профессионально использовать компьютер, следует научиться понимать слово машины. Для этого и разработаны восьмеричная и шестнадцатеричная системы.

Числа в этих системах читаются почти так же легко, как десятичные, требуют соответственно в три (восьмеричная) и в четыре (шестнадцатеричная) раза меньше разрядов, чем в двоичной системе (ведь числа 8 и 16 — соответственно, третья и четвертая степени числа 2).

Перевод восьмеричных и шестнадцатеричных чисел в двоичную систему очень прост: достаточно каждую цифру заменить эквивалентной ей двоичной триадой (тройкой цифр) или тетрадой (четверкой цифр).

Например:

$$537,1_8 = 101\ 011\ 111,001_2 ; 1A3, F_{16} = 1\ 1010\ 0011, 1111_2$$

$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$
 5 3 7 1

$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$
 1 A 3 F

Чтобы перевести число из двоичной системы в восьмеричную или шестнадцатеричную, его нужно разбить влево и вправо от запятой на триады (для восьмеричной) или тетрады (для шестнадцатеричной) и каждую такую группу заменить соответствующей восьмеричной (шестнадцатеричной) цифрой.

Например,

$$10101001,10111_2 = 10\ 101\ 001, 101\ 110_2 = 251,56_8$$

$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow$
 2 5 1 5 6

$$10101001,10111_2 = 1010\ 1001, 1011\ 1000_2 = A9,B8_{16}$$

$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$
 A 9 B 8

4.6. Как перевести целое число из десятичной системы в любую другую позиционную систему счисления?

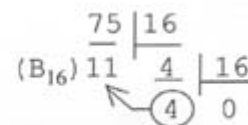
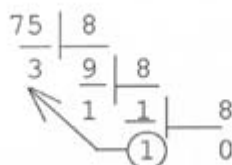
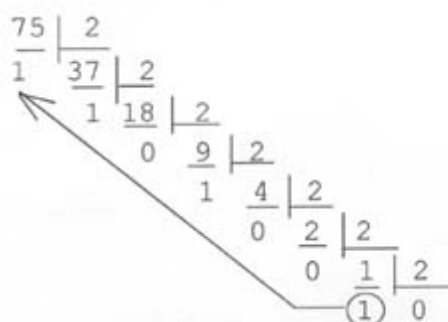
Для перевода целого десятичного числа N в систему счисления с основанием q необходимо N разделить с остатком ("нацело") на q , записанное в той же десятичной системе. Затем неполное частное, полученное от такого деления, нужно снова разделить с остатком на q , и т.д., пока последнее полученное неполное частное не станет равным нулю. Представлением числа N в новой системе счисления будет последовательность остатков деления, изображенных одной q -ичной цифрой и записанных в порядке, обратном порядку их получения.

Пример: Переведем число 75 из десятичной системы в двоичную, восьмеричную и шестнадцатеричную:

В ДВОИЧНУЮ

В ВОСЬМЕРИЧНУЮ

в шестнадцатеричную



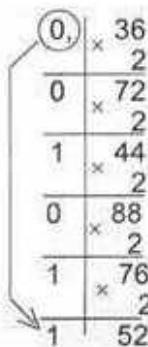
Напоминание: первый остаток 11_{10} в этом примере записывается шестнадцатеричной цифрой B_{16} .

Ответ: $75_{10} = 1\ 001\ 011_2 = 113_8 = 4B_{16}$.

4.7. Как перевести правильную десятичную дробь в любую другую позиционную систему счисления?

Для перевода правильной десятичной дроби F в систему счисления с основанием q необходимо F умножить на q , записанное в той же десятичной системе, затем дробную часть полученного произведения снова умножить на q , и т. д., до тех пор, пока дробная часть очередного произведения не станет равной нулю, либо не будет достигнута требуемая точность изображения числа F в q -ичной системе. Представлением дробной части числа F в новой системе счисления будет последовательность целых частей полученных произведений, записанных в порядке их получения и изображенных одной q -ичной цифрой. Если требуемая точность перевода числа F составляет k знаков после запятой, то предельная абсолютная погрешность при этом равняется $q^{-(k+1)} / 2$.

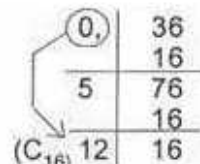
Пример. Переведем число 0,36 из десятичной системы в двоичную, восьмеричную и шестнадцатеричную:



Ответ: $0,36_{10} = 0,01011_2$
с предельной абсолютной погрешностью $(2^{-6})/2 = 2^{-7}$.



Ответ: $0,36_{10} = 0,270_8$ с предельной абсолютной погрешностью $(8^{-4})/2 = 2^{-13}$.



Ответ: $0,36_{10} = 0,5C_{16}$ с предельной абсолютной погрешностью $(16^{-3})/2 = 2^{-13}$.

Для чисел, имеющих как целую, так и дробную части, перевод из десятичной системы счисления в другую осуществляется отдельно для целой и дробной частей по правилам, указанным выше.

4.8. Как перевести число из двоичной (восьмеричной, шестнадцатеричной) системы в десятичную?

Перевод в десятичную систему числа x , записанного в q -ичной системе счисления ($q = 2, 8$ или 16) в виде $x_q = (a_n a_{n-1} \dots a_0, a_{-1} a_{-2} \dots a_{-m})_q$ сводится к вычислению значения многочлена

$$x_{10} = a_n q^n + a_{n-1} q^{n-1} + \dots + a_0 q^0 + a_{-1} q^{-1} + a_{-2} q^{-2} + \dots + a_{-m} q^{-m}$$

средствами десятичной арифметики.

Примеры:

Разряды 3 2 1 0 -1

Число 1 0 1 1, 1₂ = $1 \cdot 2^3 + 1 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} = 11,5_{10}$.

Разряды 2 1 0 -1

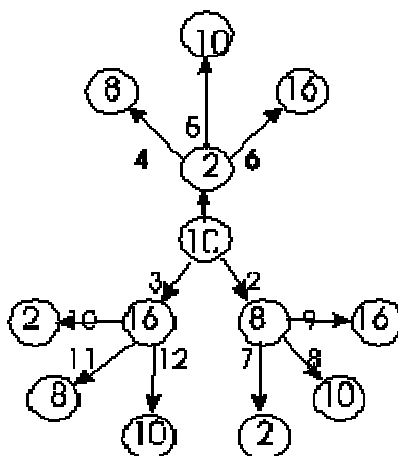
Число 2 7 6, 5₈ = $2 \cdot 8^2 + 7 \cdot 8^1 + 6 \cdot 8^0 + 5 \cdot 8^{-1} = 190,625_{10}$.

Разряды 2 1 0

Число 1 F 3₁₆ = $1 \cdot 16^2 + 15 \cdot 16^1 + 3 \cdot 16^0 = 499_{10}$.

4.9. Сводная таблица переводов целых чисел из одной системы счисления в другую

Рассмотрим только те системы счисления, которые применяются в компьютерах — десятичную, двоичную, восьмеричную и шестнадцатеричную. Для определенности возьмем произвольное десятичное число, например 46, и для него выполним все возможные последовательные переводы из одной системы счисления в другую. Порядок переводов определим в соответствии с рисунком:



На этом рисунке использованы следующие обозначения:

- в кружках записаны основания систем счисления;
- стрелки указывают направление перевода;
- номер рядом со стрелкой означает порядковый номер соответствующего примера в сводной таблице 4.1.

Например: $\textcircled{2} \xrightarrow{6} \textcircled{16}$ означает перевод из двоичной системы в шестнадцатеричную, имеющий в таблице порядковый номер 6.

Сводная таблица переводов целых чисел

Таблица 4.1.

№ п/п	Перевод	№ п/п	Перевод
1	$10 \rightarrow 2$ $\begin{array}{r} 46 \overline{) 2} \\ 0 \end{array}$ $\begin{array}{r} 23 \overline{) 2} \\ 1 \end{array}$ $\begin{array}{r} 11 \overline{) 2} \\ 1 \end{array}$ $\begin{array}{r} 5 \overline{) 2} \\ 1 \end{array}$ $\begin{array}{r} 2 \overline{) 2} \\ 0 \end{array}$ $\begin{array}{r} 2 \overline{) 2} \\ 0 \end{array}$ Ответ: 101110_2	5	$2 \rightarrow 10$ $\begin{array}{r} 5 \ 4 \ 3 \ 2 \ 1 \ 0 \\ 1 \ 0 \ 1 \ 1 \ 1 \ 0_2 = 2^4 + 2^3 + 2^2 + 2^1 = \\ = 46_{10} \\ \text{Ответ: } 46_{10} \end{array}$
		6	$2 \rightarrow 16$ $101110_2 = 10 \ 1110_2 = 2E_{16}$ Ответ: $2E_{16}$
		7	$8 \rightarrow 2$ $56_8 = 101 \ 110_2$ Ответ: $1011'0_2$
2	$10 \rightarrow 8$ $\begin{array}{r} 16 \overline{) 8} \\ 6 \ 5 \end{array}$ Ответ: 56_8	8	$8 \rightarrow 10$ $\begin{array}{r} 1 \ 0 \\ 5 \ 6_8 = 5 \cdot 8^1 + 6 \cdot 8^0 = 40 + 6 = \\ = 46_{10} \\ \text{Ответ: } 46_{10} \end{array}$
		9	$8 \rightarrow 16$ $56_8 = 101 \ 110_2 = 10 \ 1'10_2 =$ $= 2E_{16}$ Ответ: $2E_{16}$
3	$10 \rightarrow 16$ $\begin{array}{r} 46 \overline{) 16} \\ 14 \ 2 \end{array}$ Ответ: $2E_{16}$	10	$16 \rightarrow 2$ $2E_{16} = 0010 \ 11'0_2 = 101110_2$ Ответ: 101110_2
		11	$16 \rightarrow 8$ $2E_{16} = 10 \ 1110_2 = 101 \ 110_2 =$ $= 56_8$ Ответ: 56_8
4	$2 \rightarrow 8$ $101110_2 = 101 \ 110_2 = 56_8$ Ответ: 56_8	12	$16 \rightarrow 10$ $\begin{array}{r} 1 \ 0 \\ 2 \ E_{16} = 2 \cdot 16^1 + E \cdot 16^0 = \\ = 32 + 14 = 46_{10} \\ \text{Ответ: } 46_{10} \end{array}$

4.10. Как производятся арифметические операции в позиционных системах счисления?

Рассмотрим основные арифметические операции: **сложение, вычитание, умножение и деление**. Правила выполнения этих операций в десятичной системе хорошо известны — это сложение, вычитание, умножение столбиком и деление углом. Эти правила применимы и ко всем другим позиционным системам счисления. Только таблицами сложения и умножения надо пользоваться особыми для каждой системы.

С л о ж е н и е

Таблицы сложения легко составить, используя Правило Счета.

Сложение в двоичной системе

+	0	1
0	0	1
1	1	10

Сложение в восьмеричной системе

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	10
2	2	3	4	5	6	7	10	11
3	3	4	5	6	7	10	11	12
4	4	5	6	7	10	11	12	13
5	5	6	7	10	11	12	13	14
6	6	7	10	11	12	13	14	15
7	7	10	11	12	13	14	15	16

Сложение в шестнадцатеричной системе

+	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
2	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11
3	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12
4	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13
5	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14
6	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15
7	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16
8	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17
9	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18
A	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19
B	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A
C	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B
D	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C
E	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D
F	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E

При сложении цифры суммируются по разрядам, и если при этом возникает избыток, то он переносится влево.

Пример 1. Сложим числа 15 и 6 в различных системах счисления.

Десятичная: $15_{10} + 6_{10}$

$$\begin{array}{r}
 1 \\
 + 15 \\
 + 6 \\
 \hline
 21 \\
 \hline
 \end{array}
 \begin{array}{l}
 5+6=11=10+1 \\
 1+1=2
 \end{array}$$

Двоичная: $1111_2 + 110_2$

$$\begin{array}{r}
 111 \\
 + 1111 \\
 + 0110 \\
 \hline
 10101 \\
 \hline
 \end{array}
 \begin{array}{l}
 1+0=1 \\
 1+1=2=2+0 \\
 1+1+1=3=2+1 \\
 1+1=2=2+0
 \end{array}$$

Восьмеричная: $17_8 + 6_8$

$$\begin{array}{r}
 1 \\
 + 17 \\
 + 6 \\
 \hline
 25 \\
 \hline
 \end{array}
 \begin{array}{l}
 7+6=13=8+5 \\
 1+1=2
 \end{array}$$

Шестнадцатеричная: $F_{16} + 6_{16}$

$$\begin{array}{r}
 1 \\
 + F \\
 + 6 \\
 \hline
 15 \\
 \hline
 \end{array}
 \begin{array}{l}
 15+6=21=16+5
 \end{array}$$

Ответ: $15+6 = 21_{10} = 10101_2 = 25_8 = 15_{16}$.

Проверка. Преобразуем полученные

суммы к десятичному виду:

$$10101_2 = 2^4 + 2^2 + 2^0 = 16 + 4 + 1 = 21,$$

$$25_8 = 2 \cdot 8^1 + 5 \cdot 8^0 = 16 + 5 = 21,$$

$$15_{16} = 1 \cdot 16_1 + 5 \cdot 16_0 = 16 + 5 = 21.$$

Пример 2. Сложим числа 15, 7 и 3.

Десятичная: $15_{10} + 7_{10} + 3_{10}$ **Двоичная:** $1111_2 + 111_2 + 11_2$ **Восьмеричная:** $17_8 + 7_8 + 3_8$

$$\begin{array}{r}
 1 \\
 + 15 \\
 7 \\
 3 \\
 \hline
 25 \\
 \hline
 \begin{array}{l}
 5+7+3=15=10+5 \\
 1+1=2
 \end{array}
 \end{array}$$

$$\begin{array}{r}
 11+1 \quad 1 \\
 + 1111 \\
 111 \\
 11 \\
 \hline
 11001 \\
 \hline
 \begin{array}{l}
 1+1+1=3=2+1 \\
 1+1+1+1=4=2 \cdot 2+0 \\
 1+1=2=2+0 \\
 1+1+1=3=2+1
 \end{array}
 \end{array}$$

$$\begin{array}{r}
 2 \\
 + 17 \\
 7 \\
 3 \\
 \hline
 31 \\
 \hline
 \begin{array}{l}
 7+7+3=17=2 \cdot 8+1 \\
 2+1=3
 \end{array}
 \end{array}$$

Шестнадцатеричная: $F_{16} + 7_{16} + 3_{16}$

$$\begin{array}{r}
 F \\
 + 7 \\
 3 \\
 \hline
 19 \\
 \hline
 15+7+3=25=16+9
 \end{array}$$

Ответ: $5+7+3 = 25_{10} = 11001_2 = 31_8 = 19_{16}$.

Проверка:

$$11001_2 = 2^4 + 2^3 + 2^0 = 16+8+1=25,$$

$$31_8 = 3 \cdot 8^1 + 1 \cdot 8^0 = 24 + 1 = 25,$$

$$19_{16} = 1 \cdot 16^1 + 9 \cdot 16^0 = 16+9 = 25.$$

Пример 3. Сложим числа 141,5 и 59,75.

Десятичная: $141,5_{10} + 59,75_{10}$

Двоичная: $10001101,1_2 + 111011,11_2$

$$\begin{array}{r}
 111 \\
 + 141,50 \\
 59,75 \\
 \hline
 201,25 \\
 \hline
 \begin{array}{l}
 0+5=5 \\
 5+7=12=10+2 \\
 1+9+1=11=10+1 \\
 4+5+1=10=10+0 \\
 1+1=2
 \end{array}
 \end{array}$$

$$\begin{array}{r}
 1111111 \\
 + 10001101,1 \\
 111011,11 \\
 \hline
 11001001,01 \\
 \hline
 \begin{array}{l}
 1+0=1 \\
 1+1=2=2+0 \\
 1+1=2=2+0 \\
 1+1+1=3=2+1 \\
 1+1=2=2+0 \\
 1+1=2=2+0 \\
 1+1=2=2+0
 \end{array}
 \end{array}$$

Восьмеричная: $215,4_8 + 73,6_8$

Шестнадцатеричная: $8D,8_{16} + 3B,C_{16}$

$$\begin{array}{r}
 111 \\
 + 215,4 \\
 73,6 \\
 \hline
 311,2 \\
 \hline
 \begin{array}{l}
 4+6=10=8+2 \\
 5+3+1=9=8+1 \\
 1+7+1=9=8+1 \\
 2+1=3
 \end{array}
 \end{array}$$

$$\begin{array}{r}
 11 \\
 + 8D,8 \\
 3B,C \\
 \hline
 C9,4 \\
 \hline
 \begin{array}{l}
 8+12=20=16+4 \\
 13+11+1=25=16+9 \\
 8+3+1=12=C_{16}
 \end{array}
 \end{array}$$

Ответ: $141,5 + 59,75 = 201,25_{10} = 11001001,01_2 = 311,2_8 = C9,4_{16}$

Проверка. Преобразуем полученные суммы к десятичному виду:

$$11001001,01_2 = 2^7 + 2^6 + 2^3 + 2^0 + 2^{-2} = 201,25$$

$$311,2_8 = 3 \cdot 8^2 + 1 \cdot 8^1 + 1 \cdot 8^0 + 2 \cdot 8^{-1} = 201,25$$

$$C9,4_{16} = 12 \cdot 16^1 + 9 \cdot 16^0 + 4 \cdot 16^{-1} = 201,25$$

В ы ч и т а н и е

Пример 4. Вычтем единицу из чисел 10_2 , 10_8 и 10_{16}

Двоичная: $10_2 - 1_2$

$$\begin{array}{r} 1 \\ - 10 \\ \hline 1 \\ \underline{2-1=1} \end{array}$$

Восьмеричная: $10_8 - 1_8$

$$\begin{array}{r} 1 \\ - 10 \\ \hline 7 \\ \underline{8-1=7} \end{array}$$

Шестнадцатеричная: $10_{16} - 1_{16}$

$$\begin{array}{r} 1 \\ - 10 \\ \hline F \\ \underline{16-1=15=F_{16}} \end{array}$$

Пример 5. Вычтем единицу из чисел 100_2 , 100_8 и 100_{16} .

Двоичная: $100_2 - 1_2$

$$\begin{array}{r} 1 \\ - 100 \\ \hline 11 \\ \underline{10-1=9} \\ \underline{1-0=1} \end{array}$$

Восьмеричная: $100_8 - 1_8$

$$\begin{array}{r} 1 \\ - 100 \\ \hline 77 \\ \underline{78-1=77} \\ \underline{7-0=7} \end{array}$$

Шестнадцатеричная: $100_{16} - 1_{16}$

$$\begin{array}{r} 1 \\ - 100 \\ \hline FF \\ \underline{161-1=160=F_{16}} \\ \underline{1+1=2} \end{array}$$

Пример 6. Вычтем число $59,75$ из числа $201,25$.

Десятичная: $201,25_{10} - 59,75_{10}$

$$\begin{array}{r} 1 \quad 1 \\ - 201,25 \\ \hline 59,75 \\ \hline 141,50 \\ \hline \begin{array}{l} 5-5=0 \\ 10+2-7=5 \\ 10-9=1 \\ 9-5=4 \\ 2-1=1 \end{array} \end{array}$$

Двоичная: $11001001,01_2 - 111011,11_2$

$$\begin{array}{r} 1 \quad 1 \quad 1 \quad \text{Заемы} \\ - 11001001,01 \\ \hline 00111011,11 \\ \hline 10001101,10 \\ \hline \begin{array}{l} 1-0=1 \\ 0-0=0 \\ 1-1=0 \\ 1-1=0 \\ 2-1=1 \end{array} \quad \begin{array}{l} 101,10 \\ \hline 1-1=0 \\ 2-1=1 \\ 1-1=0 \\ 1-0=1 \end{array} \end{array}$$

Восьмеричная: $311,2_8 - 73,6_8$

$$\begin{array}{r} 1 \quad 1 \quad 1 \\ - 311,2 \\ \hline 73,6 \\ \hline 215,4 \\ \hline \begin{array}{l} 8+2-6=4 \\ 8-3=5 \\ 8-7=1 \end{array} \end{array}$$

Шестнадцатеричная: $C9,4_{16} - 3B,C_{16}$

$$\begin{array}{r} 1 \quad 1 \\ - C9,4 \\ \hline 3B,C \\ \hline 8D,8 \\ \hline \begin{array}{l} 16+4-12=8 \\ 16+8-11=13=D_{16} \\ 12-1-3=8 \end{array} \end{array}$$

Ответ: $201,25_{10} - 59,75_{10} = 141,5_{10} = 10001101,1_2 = 215,4_8 = 8D,8_{16}$.

Проверка. Преобразуем полученные разности к десятичному виду:

$$10001101,1_2 = 2^7 + 2^3 + 2^2 + 2^0 + 2^{-1} = 141,5;$$

$$215,4_8 = 2 \cdot 8^2 + 1 \cdot 8^1 + 5 \cdot 8^0 + 4 \cdot 8^{-1} = 141,5;$$

$$8D,8_{16} = 8 \cdot 16^1 + D \cdot 16^0 + 8 \cdot 16^{-1} = 141,5.$$

У м н о ж е н и е

Выполняя умножение многозначных чисел в различных позиционных системах счисления, можно использовать обычный алгоритм перемножения чисел в столбик, но при этом результаты перемножения и сложения однозначных чисел необходимо заимствовать из соответствующих рассматриваемой системе таблиц умножения и сложения.

Умножение в двоичной системе

*	0	1
0	0	0
1	0	1

Умножение в восьмеричной системе

*	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	10	12	14	16
3	0	3	6	11	14	17	22	25
4	0	4	10	14	20	24	30	34
5	0	5	12	17	24	31	36	43
6	0	6	14	22	30	36	44	52
7	0	7	16	25	34	43	52	61

Ввиду чрезвычайной простоты таблицы умножения в двоичной системе, умножение сводится лишь к сдвигам множимого и сложениям.

Пример 7. Перемножим числа 5 и 6.

Десятичная: $5_{10} \cdot 6_{10}$

$$\begin{array}{r} \times 5 \\ 6 \\ \hline 30 \end{array}$$

Двоичная: $101_2 \cdot 110_2$

$$\begin{array}{r} \times 101 \\ 110 \\ \hline 101 \\ 1110 \\ \hline 11110 \end{array}$$

Восьмеричная: $5_8 \cdot 6_8$

$$\begin{array}{r} \times 5 \\ 6 \\ \hline 36 \end{array}$$

Ответ: $5 \cdot 6 = 30_{10} = 11110_2 = 36_8$.

Проверка. Преобразуем полученные произведения к десятичному виду:

$$11110_2 = 2^4 + 2^3 + 2^2 + 2^1 = 30;$$

$$36_8 = 3 \cdot 8^1 + 6 \cdot 8^0 = 30.$$

Пример 8. Перемножим числа 115 и 51.

Десятичная: $115_{10} \cdot 51_{10}$

$$\begin{array}{r} \times 115 \\ 51 \\ \hline 115 \\ 575 \\ \hline 5865 \end{array}$$

Двоичная: $1110011_2 \cdot 110011_2$

$$\begin{array}{r} \times 1110011 \\ 110011 \\ \hline 1110011 \\ 1110011 \\ \hline 1110011 \\ 1110011 \\ \hline 1011011101001 \end{array}$$

Восьмеричная: $163_8 \cdot 63_8$

$$\begin{array}{r} \times 163 \\ 63 \\ \hline 531 \\ 1262 \\ \hline 13351 \end{array}$$

Ответ: $115 \cdot 51 = 5865_{10} = 1011011101001_2 = 13351_8$.

Проверка. Преобразуем полученные произведения к десятичному виду:

$$1011011101001_2 = 2^{12} + 2^{10} + 2^9 + 2^7 + 2^6 + 2^5 + 2^3 + 2^0 = 5865;$$

$$13351_8 = 1 \cdot 8^4 + 3 \cdot 8^3 + 3 \cdot 8^2 + 5 \cdot 8^1 + 1 \cdot 8^0 = 5865.$$

Деление

Деление в любой позиционной системе счисления производится по тем же правилам, как и деление углом в десятичной системе. В двоичной системе деление выполняется особенно просто, ведь очередная цифра частного может быть только нулем или единицей.

Пример 9. Разделим число 30 на число 6.

Десятичная: $30_{10} : 6_{10}$

$$\begin{array}{r} 30 \overline{) 6} \\ \underline{30} \\ 0 \end{array}$$

Двоичная: $11110_2 : 110_2$

$$\begin{array}{r} 11110 \overline{) 110} \\ \underline{110} \\ 110 \\ \underline{110} \\ 0 \end{array}$$

Восьмеричная: $36_8 : 6_8$

$$\begin{array}{r} 36 \overline{) 6} \\ \underline{36} \\ 0 \end{array}$$

Ответ: $30 : 6 = 5_{10} = 101_2 = 5_8$.

Пример 10. Разделим число 5865 на число 115.

Десятичная: $5865_{10} : 115_{10}$

$$\begin{array}{r} 5865 \overline{) 115} \\ \underline{575} \\ 115 \\ \underline{115} \\ 0 \end{array}$$

Двоичная: $1011011101001_2 : 1110011_2$

$$\begin{array}{r} 1011011101001 \overline{) 1110011} \\ \underline{1110011} \\ 1000100 \\ \underline{1110011} \\ 10101100 \\ \underline{1110011} \\ 1110011 \\ \underline{1110011} \\ 0 \end{array}$$

Восьмеричная: $13351_8 : 163_8$

$$\begin{array}{r} 13351 \overline{) 163} \\ \underline{1262} \\ 531 \\ \underline{531} \\ 0 \end{array}$$

Ответ: $5865 : 115 = 51_{10} = 110011_2 = 63_8$.

Проверка. Преобразуем полученные частные к десятичному виду:

$$110011_2 = 2^5 + 2^4 + 2^1 + 2^0 = 51; \quad 63_8 = 6 \cdot 8^1 + 3 \cdot 8^0 = 51.$$

Пример 11. Разделим число 35 на число 14.

Десятичная: $35_{10} : 14_{10}$

$$\begin{array}{r} 35 \overline{) 14} \\ \underline{28} \\ 70 \\ \underline{70} \\ 0 \end{array}$$

Двоичная: $100011_2 : 1110_2$

$$\begin{array}{r} 100011 \overline{) 1110} \\ \underline{1110} \\ 1110 \\ \underline{1110} \\ 0 \end{array}$$

Восьмеричная: $43_8 : 16_8$

$$\begin{array}{r} 43 \overline{) 16} \\ \underline{34} \\ 70 \\ \underline{70} \\ 0 \end{array}$$

Ответ: $35 : 14 = 2,5_{10} = 10,1_2 = 2,4_8$.

Проверка. Преобразуем полученные частные к десятичному виду:

$$10,1_2 = 2^1 + 2^{-1} = 2,5;$$

$$2,4_8 = 2 \cdot 8^0 + 4 \cdot 8^{-1} = 2,5.$$

4.11. Как представляются в компьютере целые числа?

Целые числа могут представляться в компьютере со знаком или без знака.

Целые числа без знака

Обычно занимают в памяти компьютера один или два байта. В однобайтовом формате принимают значения от 00000000_2 до 11111111_2 . В двухбайтовом формате - от $00000000\ 00000000_2$ до $11111111\ 11111111_2$.

Диапазоны значений целых чисел без знака

Формат числа в байтах	Диапазон	
	Запись с порядком	Обычная запись
1	$0 \dots 2^8-1$	0 ... 255
2	$0 \dots 2^{16}-1$	0 ... 65535

Примеры:

а) число $72_{10} = 1001000_2$ в однобайтовом формате:

Номера разрядов	7 6 5 4 3 2 1 0
Биты числа	0 1 0 0 1 0 0 0

б) это же число в двухбайтовом формате:

Номера разрядов	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Биты числа	0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0

в) число 65535 в двухбайтовом формате:

Номера разрядов	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Биты числа	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Целые числа со знаком

Обычно занимают в памяти компьютера один, два или четыре байта, при этом самый левый (старший) разряд содержит информацию о знаке числа.

Диапазоны значений целых чисел со знаком

Формат числа в байтах	Диапазон	
	Запись с порядком	Обычная запись
1	$-2^7 \dots 2^7-1$	-128 ... 127
2	$-2^{15} \dots 2^{15}-1$	-32768 ... 32767
4	$-2^{31} \dots 2^{31}-1$	-2147483648 ... 2147483647

Рассмотрим особенности записи целых чисел со знаком на примере **однобайтового формата**, при котором для знака отводится один разряд, а для цифр абсолютной величины - семь разрядов.

В компьютерной технике применяются три формы записи (кодирования) целых чисел со знаком: прямой код, обратный код, дополнительный код.

Последние две формы применяются особенно широко, так как позволяют упростить конструкцию арифметико-логического устройства компьютера путем замены разнообразных арифметических операций операцией сложения.

Положительные числа в прямом, обратном и дополнительном кодах изображаются одинаково - двоичными кодами с цифрой 0 в знаковом разряде. Например:

Число $1_{10}=1_2$

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

Знак числа "+"

Число $127_{10}=1111111_2$

0	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

Знак числа "+"

Отрицательные числа в прямом, обратном и дополнительном кодах имеют разное изображение.

1. **Прямой код.** В знаковый разряд помещается цифра 1, а в разряды цифровой части числа — двоичный код его абсолютной величины. Например:

Прямой код числа - 1

1	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

Знак числа "-"

Прямой код числа - 127

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

Знак числа "-"

2. **Обратный код.** Получается инвертированием всех цифр двоичного кода абсолютной величины числа, включая разряд знака: нули заменяются единицами, а единицы — нулями. Например:

Число: -1
Код модуля числа: 0 0000001
Обратный код числа: 1 1111110

1	1	1	1	1	1	1	0
---	---	---	---	---	---	---	---

Число: -127
Код модуля числа: 0 1111111
Обратный код числа: 1 0000000

1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

3. **Дополнительный код.** Получается образованием обратного кода с последующим прибавлением единицы к его младшему разряду. Например:

Дополнительный код числа - 1

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

Дополнительный код числа - 127

1	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

Обычно отрицательные десятичные числа при вводе в машину автоматически преобразуются в обратный или дополнительный двоичный код и в таком виде хранятся, перемещаются и участвуют в операциях. При выводе таких чисел из машины происходит обратное преобразование в отрицательные десятичные числа.

4.12. Как компьютер выполняет арифметические действия над целыми числами?

Сложение и вычитание

В большинстве компьютеров операция вычитания не используется. Вместо нее производится сложение обратных или дополнительных кодов уменьшаемого и вычитаемого. Это позволяет существенно упростить конструкцию АЛУ.

Сложение обратных кодов. Здесь при сложении чисел А и В имеют место четыре основных и два особых случая:

1. А и В положительные. При суммировании складываются все разряды, включая разряд знака. Так как знаковые разряды положительных слагаемых равны нулю, разряд знака суммы тоже равен нулю. Например:

Десятичная запись

$$\begin{array}{r} + 3 \\ + 7 \\ \hline 10 \end{array}$$

Двоичные коды

$$\begin{array}{r} + 0000011 \\ + 0000111 \\ \hline 00001010 \end{array}$$

Получен правильный результат.

2. А положительное, В отрицательное и по абсолютной величине больше, чем А. Например:

Десятичная запись

$$\begin{array}{r} + 3 \\ + -10 \\ \hline -7 \end{array}$$

Двоичные коды

$$\begin{array}{r} + 0000011 \\ + 11110101 \\ \hline 11111000 \end{array}$$

Обратный код числа -10
Обратный код числа -7

Получен правильный результат в обратном коде. При переводе в прямой код биты цифровой части результата инвертируются: $10000111 = -7_{10}$.

3. А положительное, В отрицательное и по абсолютной величине меньше, чем А. Например:

Десятичная запись

$$\begin{array}{r} + 10 \\ + -3 \\ \hline 7 \end{array}$$

Двоичные коды

$$\begin{array}{r} + 0001010 \\ + 11111100 \\ \hline 0000110 \\ \rightarrow +1 \\ \hline 0000111 \end{array}$$

Обратный код числа -3

Компьютер исправляет полученный первоначально неправильный результат (6 вместо 7) переносом единицы из знакового разряда в младший разряд суммы.

4. А и В отрицательные. Например:

Десятичная запись

$$\begin{array}{r} + -3 \\ + -7 \\ \hline -10 \end{array}$$

Двоичные коды

$$\begin{array}{r} + 11111100 \\ + 11111000 \\ \hline 11110100 \\ \rightarrow +1 \\ \hline 11110101 \end{array}$$

Обратный код числа -3
Обратный код числа -7
Обратный код числа -10

Полученный первоначально неправильный результат (обратный код числа -11_{10} вместо обратного кода числа -10_{10}) компьютер исправляет переносом единицы из знакового разряда

в младший разряд суммы. При переводе результата в прямой код биты цифровой части числа инвертируются: $1\ 0001010 = -10_{10}$.

При сложении может возникнуть ситуация, когда старшие разряды результата операции не помещаются в отведенной для него области памяти. Такая ситуация называется **переполнением разрядной сетки формата числа**. Для обнаружения переполнения и оповещения о возникшей ошибке в компьютере используются специальные средства. Ниже приведены два возможных случая переполнения.

5. А и В положительные, сумма А+В больше, либо равна 2^{n-1} , где n — количество разрядов формата чисел (для однобайтового формата $n=8$, $2^{n-1} = 2^7 = 128$). Например:

Десятичная запись	Двоичные коды
$\begin{array}{r} +\ 65 \\ +\ 97 \\ \hline 162 \end{array}$	$\begin{array}{r} +\ 0\ 1000001 \\ +\ 0\ 1100001 \\ \hline 1\ 0100010 \end{array}$
	Переполнение

Семи разрядов цифровой части числового формата недостаточно для размещения восьмизначной суммы ($162_{10} = 10100010_2$), поэтому старший разряд суммы оказывается в знаковом разряде. Это вызывает несовпадение знака суммы и знаков слагаемых, что является свидетельством переполнения разрядной сетки.

6. А и В отрицательные, сумма абсолютных величин А и В больше, либо равна 2^{n-1} . Например:

Десятичная запись	Двоичные коды
$\begin{array}{r} +\ -63 \\ +\ -95 \\ \hline 158 \end{array}$	$\begin{array}{r} +\ 1\ 1000000 \\ +\ 1\ 0100000 \\ \hline 0\ 1100000 \end{array}$
	Обратный код числа -63 Обратный код числа -95
	Переполнение
	→ +1

Здесь знак суммы тоже не совпадает со знаками слагаемых, что свидетельствует о переполнении разрядной сетки.

Сложение дополнительных кодов. Здесь также имеют место рассмотренные выше шесть случаев:

1. А и В положительные. Здесь нет отличий от случая 1, рассмотренного для обратного кода.

2. А положительное, В отрицательное и по абсолютной величине больше, чем А. Например:

Десятичная запись	Двоичные коды
$\begin{array}{r} +\ 3 \\ +\ -10 \\ \hline -7 \end{array}$	$\begin{array}{r} +\ 0\ 0000011 \\ +\ 1\ 1110110 \\ \hline 1\ 1110011 \end{array}$
	Дополнительный код числа -10 Дополнительный код числа -7

Получен правильный результат в дополнительном коде. При переводе в прямой код биты цифровой части результата инвертируются и к младшему разряду прибавляется единица: $1\ 0000110 + 1 = 1\ 0000111 = -7_{10}$.

3. А положительное, В отрицательное и по абсолютной величине меньше, чем А. Например:

Десятичная запись	Двоичные коды
$\begin{array}{r} +\ 10 \\ +\ -3 \\ \hline 7 \end{array}$	$\begin{array}{r} +\ 0\ 0001010 \\ +\ 1\ 1111101 \\ \hline 0\ 0000111 \end{array}$
	Дополнительный код числа -3
	→ перенос отбрасывается

Получен правильный результат. Единицу переноса из знакового разряда компьютер отбрасывает.

4. А и В отрицательные. Например:

Десятичная запись

$$\begin{array}{r} + \quad -3 \\ + \quad -7 \\ \hline -10 \end{array}$$

Двоичные коды

$$\begin{array}{r} + \quad 1 \ 111101 \\ + \quad 1 \ 111001 \\ \hline 1 \ 1110110 \end{array}$$

Дополнительный код числа -3
Дополнительный код числа -7
Дополнительный код числа -10

→ перенос отбрасывается

Получен правильный результат в дополнительном коде. Единицу переноса из знакового разряда компьютер отбрасывает.

Случаи переполнения для дополнительных кодов рассматриваются по аналогии со случаями 5 и 6 для обратных кодов.

Сравнение рассмотренных форм кодирования целых чисел со знаком показывает:

- на преобразование отрицательного числа в обратный код компьютер затрачивает меньше времени, чем на преобразование в дополнительный код, так как последнее состоит из двух шагов — образования обратного кода и прибавления единицы к его младшему разряду;
- время выполнения сложения для дополнительных кодов чисел меньше, чем для их обратных кодов, потому что в таком сложении нет переноса единицы из знакового разряда в младший разряд результата.

Умножение и деление

Во многих компьютерах **умножение** производится как последовательность сложений и сдвигов. Для этого в АЛУ имеется **регистр**, называемый **накапливающим сумматором**, который до начала выполнения операции **содержит число ноль**. В процессе выполнения операции в нем поочередно размещаются **множимое** и **результаты промежуточных сложений**, а по завершении операции — **окончательный результат**.

Другой регистр АЛУ, участвующий в выполнении этой операции, **вначале содержит множитель**. Затем по мере выполнения сложений содержащееся в нем **число уменьшается**, пока не достигнет нулевого значения.

Для иллюстрации умножим 110011_2 на 101101_2 .

Накапливающий сумматор

$$\begin{array}{r} + \quad 0000000000 \\ + \quad 110011 \\ \hline + \quad 110011 \\ + \quad 110011 \\ \hline + \quad 11111111 \\ + \quad 110011 \\ \hline + \quad 1010010111 \\ + \quad 110011 \\ \hline 100011110111 \end{array}$$

Множитель

$$\begin{array}{r} 101101 \\ 101100 \\ 101000 \\ 100000 \\ 000000 \end{array}$$

Сдвиг на две позиции влево
Сдвиг на одну позицию влево
Сдвиг на две позиции влево

Деление для компьютера является трудной операцией. Обычно оно реализуется путем многократного прибавления к делимому дополнительного кода делителя.

4.13. Как представляются в компьютере вещественные числа?

Система вещественных чисел в математических вычислениях предполагается **непрерывной и бесконечной**, т.е. не имеющей ограничений на диапазон и точность представления чисел. Однако в компьютерах числа хранятся в регистрах и ячейках памяти с

ограниченным количеством разрядов. В следствие этого **система вещественных чисел, представимых в машине, является дискретной (прерывной) и конечной.**

При написании вещественных чисел в программах вместо привычной запятой принято ставить точку. Для отображения вещественных чисел, которые могут быть как очень маленькими, так и очень большими, используется форма записи чисел с **порядком основания системы счисления**. Например, десятичное число 1.25 в этой форме можно представить так:

$$1.25 \cdot 10^0 = 0.125 \cdot 10^1 = 0.0125 \cdot 10^2 = \dots$$

или так:

$$12.5 \cdot 10^{-1} = 125.0 \cdot 10^{-2} = 1250.0 \cdot 10^{-3} = \dots$$

Любое число N в системе счисления с основанием q можно записать в виде $N = M \cdot q^p$, где M — множитель, содержащий все цифры числа (мантисса), а p — целое число, называемое порядком. Такой способ записи чисел называется представлением числа с плавающей точкой.

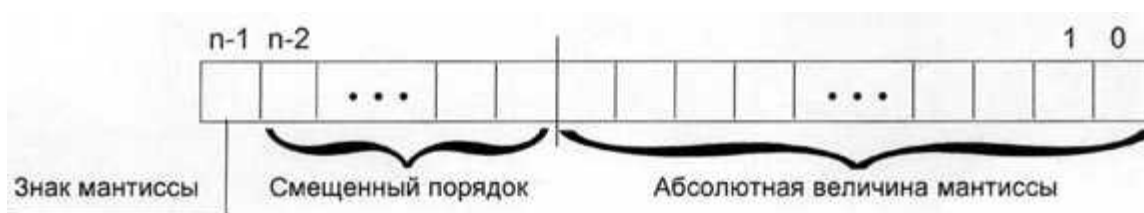
Если "плавающая" точка расположена в мантиссе перед первой значащей цифрой, то при фиксированном количестве разрядов, отведённых под мантиссу, обеспечивается запись максимального количества значащих цифр числа, то есть максимальная точность представления числа в машине. Из этого следует:

Мантисса должна быть правильной дробью, у которой первая цифра после точки (запятой в обычной записи) отлична от нуля: $0.1_2 \leq |M| < 1$. Если это требование выполнено, то число называется нормализованным

Мантиссу и порядок q -ичного числа принято записывать в системе с основанием q , а само основание — в десятичной системе. Примеры нормализованного представления:

Десятичная система	Двоичная система
$753.15 = 0.75315 \cdot 10^3$;	$-101.01 = -0.10101 \cdot 2^{11}$ (порядок $11_2 = 3_{10}$)
$-0.000034 = -0.34 \cdot 10^{-4}$;	$0.000011 = 0.11 \cdot 2^{-100}$ (порядок $-100_2 = -4_{10}$).

Вещественные числа в компьютерах различных типов записываются по-разному, тем не менее, **все компьютеры поддерживают несколько международных стандартных форматов, различающихся по точности, но имеющих одинаковую структуру следующего вида:**



Здесь порядок n -разрядного нормализованного числа задается в так называемой **смещенной форме**: если для задания порядка выделено k разрядов, то к истинному значению порядка, **представленного в дополнительном коде**, прибавляют смещение, равное $(2^{k-1} - 1)$. Например, порядок, принимающий значения в диапазоне от -128 до $+127$, представляется смещенным порядком, значения которого меняются от 0 до 255 .

Использование смещенной формы позволяет производить операции над порядками, как над беззнаковыми числами, что упрощает операции сравнения, сложения и вычитания порядков, а также упрощает операцию сравнения самих нормализованных чисел.

Чем больше разрядов отводится под запись мантиссы, тем выше точность представления числа. Чем больше разрядов занимает порядок, тем шире диапазон от наименьшего отличного от нуля числа до наибольшего числа, представимого в машине при заданном формате.

Стандартные форматы представления вещественных чисел:

1) **одинарный** — 32-разрядное нормализованное число со знаком, 8-разрядным смещенным порядком и 24-разрядной мантиссой (старший бит мантиссы, всегда равный 1, не хранится в памяти, и размер поля, выделенного для хранения мантиссы, составляет только 23 разряда).

2) **двойной** — 64-разрядное нормализованное число со знаком, 11-разрядным смещенным порядком и 53-разрядной мантиссой (старший бит мантиссы не хранится, размер поля, выделенного для хранения мантиссы, составляет 52 разряда).

3) **расширенный** — 80-разрядное число со знаком, 15-разрядным смещенным порядком и 64-разрядной мантиссой. Позволяет хранить ненормализованные числа.

Следует отметить, что вещественный формат с m -разрядной мантиссой позволяет абсолютно точно представлять m -разрядные целые числа, т. е. **любое двоичное целое число, содержащее не более m разрядов, может быть без искажений преобразовано в вещественный формат.**

4.14. Как компьютер выполняет арифметические действия над нормализованными числами?

К началу выполнения арифметического действия операнды операции помещаются в соответствующие регистры АЛУ.

Сложение и вычитание

При сложении и вычитании сначала производится подготовительная операция, называемая **выравниванием порядков**.

В процессе выравнивания порядков мантисса числа с меньшим порядком сдвигается в своем регистре вправо на количество разрядов, равное разности порядков операндов. После каждого сдвига порядок увеличивается на единицу.

В результате выравнивания порядков одноименные разряды чисел оказываются расположенными в соответствующих разрядах обоих регистров, после чего **мантиссы складываются или вычитаются**. В случае необходимости полученный результат нормализуется путем сдвига мантиссы результата влево. После каждого сдвига влево порядок результата уменьшается на единицу.

Пример 1. Сложить двоичные нормализованные числа $0.10111 \cdot 2^{-1}$ и $0.11011 \cdot 2^{10}$. Разность порядков слагаемых здесь равна трем, поэтому перед сложением мантисса первого числа сдвигается на три разряда вправо:

$$\begin{array}{r}
 + \quad 0.00010111 \cdot 2^{10} \\
 \quad 0.11011 \quad \cdot 2^{10} \\
 \hline
 0.11101111 \cdot 2^{10}
 \end{array}$$

Пример 2. Выполнить вычитание двоичных нормализованных чисел $0.10101 \cdot 2^{10}$ и $0.11101 \cdot 2^1$. Разность порядков уменьшаемого и вычитаемого здесь равна единице, поэтому перед вычитанием мантисса второго числа сдвигается на один разряд вправо:

$$\begin{array}{r}
 - \quad 0.10101 \quad \cdot 2^{10} \\
 \quad 0.011101 \cdot 2^{10} \\
 \hline
 0.001101 \cdot 2^{10}
 \end{array}$$

Результат получился **не нормализованным**, поэтому его **мантисса сдвигается влево на два разряда** с соответствующим уменьшением порядка на две единицы: $0.1101 \cdot 2^0$.

Умножение

При умножении двух нормализованных чисел их порядки складываются, а мантиссы перемножаются.

Пример 3. Выполнить умножение двоичных нормализованных чисел:

$$(0.11101 \cdot 2^{101}) \cdot (0.1001 \cdot 2^{11}) = (0.11101 \cdot 0.1001) \cdot 2^{(101+11)} = 0.100000101 \cdot 2^{1000}.$$

Деление

При делении двух нормализованных чисел из порядка делимого вычитается порядок делителя, а мантисса делимого делится на мантиссу делителя. Затем в случае необходимости полученный результат нормализуется.

Пример 4. Выполнить деление двоичных нормализованных чисел:

$$0.1111 \cdot 2^{100} : 0.101 \cdot 2^{11} = (0.1111 : 0.101) \cdot 2^{(100-11)} = 1.1 \cdot 2^1 = 0.11 \cdot 2^{10}.$$

Использование представления чисел с плавающей точкой существенно усложняет схему арифметико-логического устройства.