Уральский федеральный университет
имени первого Президента России Б. Н. Ельцина
Высшая школа экономики и менеджмента

А. П. ШАМАНОВ

СИСТЕМЫ СЧИСЛЕНИЯ И ПРЕДСТАВЛЕНИЕ ЧИСЕЛ В ЭВМ

Учебное пособие

Министерство образования и науки Российской Федерации

Уральский федеральный университет
имени первого Президента России Б. Н. Ельцина

А. П. Шаманов

СИСТЕМЫ СЧИСЛЕНИЯ И ПРЕДСТАВЛЕНИЕ ЧИСЕЛ В ЭВМ

Учебное пособие

Рекомендовано методическим советом УрФУ для студентов,
обучающихся по направлениям подготовки
38.03.05 «Бизнес-информатика»,
09.03.03 «Прикладная информатика»,
38.03.01 «Экономика», 38.03.02 «Менеджмент»,
38.05.01 «Экономическая безопасность»,
38.05.02 «Таможенное дело»

Екатеринбург
Издательство Уральского университета
2016

УДК 511.11:004.4(075.8)
ББК 22.131я73 + 32.972я73
Ш19

Рецензенты:
заведующий сектором канд. физ.-мат. наук *Д. Г. Ермаков* (Институт математики и механики им. Н. Н. Красовского УрО РАН);
заведующий кафедрой «Информационные технологии и математическое моделирование» проф., д-р физ.-мат. наук *А. Н. Красовский* (Уральский государственный аграрный университет)

Шаманов, А. П.
Ш19 Системы счисления и представление чисел в ЭВМ : учебное пособие / А. П. Шаманов. — Екатеринбург : Изд-во Урал. ун-та, 2016. — 52 с.
ISBN 978-5-7996-1719-6

В данном пособии дано описание позиционных систем счисления, показаны правила выполнения арифметических операций, описаны методы перевода чисел из одной системы счисления в другую, показано представление как целых, так и дробных чисел. Помимо этого рассмотрены методы представления числовой информации в ЭВМ.

Пособие предназначено в качестве дополнительного источника для студентов практически всех специальностей, изучающих курсы дисциплин «Информатика» или «Архитектура ЭВМ».

Библиогр.: 10 назв. Табл. 10. Рис. 1.

УДК 511.11:004.4(075.8)
ББК 22.131я73 + 32.972я73

ISBN 978-5-7996-1719-6

© Уральский федеральный университет, 2016

ВВЕДЕНИЕ

Современный компьютер и другие устройства вычислительной техники основаны на использовании двоичной системы счисления. В двоичной системе счисления используется всего два символа — 0 и 1, а записи, сформированные из них, достаточно длинны и достаточно плохо воспринимаются человеком. Для их интерпретации используется шестнадцатеричная система, записи которой значительно короче и легче воспринимаются человеком. Перевод же из шестнадцатеричной системы в двоичную и наоборот весьма прост и нагляден, и при описании устройств вычислительной техники везде, где это возможно, используется шестнадцатеричная система счисления. Поэтому для правильного понимания работы вычислительных систем необходимо знание двоичной и шестнадцатеричной систем счисления. И уж тем более оно необходимо при написании программных продуктов.

1. СИСТЕМЫ СЧИСЛЕНИЯ

1.1. Понятие системы счисления

Представление целых положительных чисел с помощью письменных знаков (символов) называется нумерацией. Письменные знаки (символы), используемые при нумерации, называются цифрами. Необходимо четко делать различие между числом и символом (группой символов), которым пользуются для его письменного воспроизведения. Например, с одной стороны, для изображения числа «пять» могут использоваться цифра 5 (десятичная система нумерации), цифра V (римская система нумерации) или группа символов 101 (двоичная система нумерации). С другой стороны, группа символов 10 может обозначать число «десять» в десятичной системе или число 2 в двоичной системе. Иными словами, значение символа зависит от системы нумерации и его положения в записи, тогда как с числом всегда связана определенная количественная характеристика.

Совокупность правил записи чисел (способ соединения цифр для обозначения числа) называется *системой счисления*. Системы счисления подразделяются на *позиционные* и *непозиционные*.

Непозиционные системы счисления возникли раньше позиционных. Они характеризуются тем, что в них символы, обозначающие то или иное число, не меняют своего значения в зависимости от своего местоположения в записи этого числа. Классическим примером такой системы является римская система счисления. В ней для записи чисел используются буквы латинского алфавита. Значения основных цифр римской системы приведены ниже:
I — единица, *V* — пять, *X* — десять, *L* — пятьдесят, *C* — сто, *D* — пятьсот, *M* — тысяча.

Для получения количественного эквивалента числа в римской системе необходимо просто сложить количественные эквиваленты входящих в него цифр. Исключение составляет случай, когда младшая цифра стоит перед старшей — в такой ситауции количественный эквивалент младшей цифры берут со знаком «минус». Некоторые примеры чисел в римской системе счисления и их десятичные эквиваленты приведены в табл. 1.1.

Таблица 1.1

Некоторые числа в римской системе счисления

Число в римской системе	Значение в десятичной системе
III	1 + 1 + 1 = 3
IV	5–1 = 4
XII	10 + 1 + 1 = 12
XLV	–10 + 50 + 5 = 45
CDXVIII	–100 + 500 + 10 + 5 + 1 + 1 + 1 = 418
MMDXCVII	1000 + 1000 + 500 – 10 + 100 + 5 + 1 + 1 = 2597

Непозиционные системы счисления имеют два существенных недостатка:

- с увеличением изображаемых чисел требуется неограниченное число новых символов;

- процедура выполнения арифметических операций в таких системах счисления чрезвычайно сложна.

Поэтому в настоящее время непозиционные системы счисления практически не используются.

1.2. Позиционные системы счисления

Позиционные системы счисления характеризуются следующими понятиями:

- Для записи любого числа используется ограниченный набор символов. Число используемых символов называется *основанием* позиционной системы счисления.
- Устанавливается взаимно-однозначное соответствие между набором цифр и числами натурального ряда 0, 1, ..., $p - 1$, где p — основание системы счисления. Таким образом, численный эквивалент любой цифры меньше основания системы счисления.
- Место каждой цифры в числе называется *позицией* (отсюда, собственно, название таких систем — позиционные).
- Номер позиции цифры в числе называется *разрядом*. Нумерация разрядов начинается с нуля и выполняется справа налево. Разряд 0 называется младшим разрядом.
- Каждой цифре, в зависимости от ее позиции, ставится в соответствие количественный эквивалент, определяемый по формуле

$$\alpha_k = a_k p^k, \tag{1.1}$$

где α_k — количественный эквивалент цифры, находящейся в позиции k;

a_k — численный эквивалент цифры, находящейся в разряде k;

p — основание системы счисления;

k — номер позиции цифры (ее разряд).

- Само значение числа (его количественный эквивалент) определяется как сумма вычисленных по формуле (1.1) количественных эквивалентов всех цифр, входящих в запись числа.
- Для выполнения операции сложения каждой паре чисел, каждое из которых соотносится с какой-либо одной цифрой, ставится в соответствие число, являющееся результатом их сложения. Аналогично, для выполнения операции умножения каждой такой паре чисел ставится в соответствие число, являющееся результатом их умножения. Эти соответствия оформляются в виде таблицы сложения и таблицы умножения.

Таким образом, любое целое положительное число может быть представлено в виде

$$a_{n-1}a_{n-2}\ldots a_1a_0 = a_{n-1}p^{n-1} + a_{n-2}p^{n-2} + \ldots + a_1p^1 + a_0p^0, \qquad (1.2)$$

где a_i — цифра данной системы счисления ($0 \leqslant a_i < p$);
n — число разрядов при написании числа;
p — основание системы счисления (некоторое положительное целое число).

В качестве основания системы счисления может быть использовано любое натуральное число $p>1$. При заданном основании системы счисления p каждому натуральному числу соответствует единственное представление вида (1.2) и каждому представлению вида (1.2) соответствует единственное натуральное число. Естественно, при этом лидирующие нули не учитываются, например 000655 и 655 — это эквивалентные записи одного и того же числа.

Проиллюстрируем сказанное на привычной нам десятичной системе. Набор цифр для десятичной системы счисления: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}. Основание системы $p = 10$. Любое число в десятичной системе согласно формуле (1.2) представляется в виде

$$A_{10} = a_{n-1}a_{n-2}\ldots a_1a_0 = a_{n-1}\cdot 10^{n-1} + a_{n-2}\cdot 10^{n-2} + \ldots + a_1\cdot 10^1 + a_0\cdot 10^0,$$

где каждое a_i — одна из цифр множества {0,1,2,3,4,5,6,7,8,9}.

Например,

$625 = 6\cdot 10^2 + 2\cdot 10^1 + 5\cdot 10^0 = 6\cdot 100 + 2\cdot 10 + 5$

или

$1309 = 1\cdot 10^3 + 3\cdot 10^2 + 0\cdot 10^1 + 9\cdot 10^0 = 1000 + 3\cdot 100 + 9.$

Таблицы умножения и сложения чисел и выполнение арифметических операций в десятичной системе известны с начальной школы и здесь не приводятся.

Далее мы будем рассматривать три системы счисления: двоичную, десятичную и шестнадцатеричную. Это вызвано следующим:

- компьютер работает только с двоичной информацией;
- человек производит вычисления, используя десятичную систему;
- двоичная информация плохо воспринимается человеком, для ее интерпретации удобнее использовать шестнадцатеричную систему.

1.3. Двоичная система счисления

Набор цифр для двоичной системы счисления: {0, 1}. Основание системы $p = 2$. Любое число в двоичной системе представляет собой последовательность нулей и единиц. Для того чтобы подчеркнуть, что это именно двоичная запись, в конце числа можно (но не обязательно) поставить нижний индекс 2 или символ *b* (от английского binary — «двоичный»). Последнее обозначение является обязательным при задании двоичных констант на языке Assembler. Например,

$5 = 101_2 = 101b$
или
$1025 = 10000000001_2 = 10000000001b.$

Согласно формуле (1.2) число в двоичной системе представляется в виде

$$A_2 = a_{n-1}a_{n-2} \ldots a_1a_0 = a_{n-1} \cdot 2^{n-1} + a_{n-2} \cdot 2^{n-2} + \ldots + a_1 \cdot 2^1 + a_0 \cdot 2^0, \quad (1.3)$$

где каждое a_i — одна из цифр 0 или 1.

Например,
$101_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 1 \cdot 4 + 1$
или
$10100_2 = 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 1 \cdot 16 + 1 \cdot 4 = 20.$

В формуле (1.3) разложение двоичного числа по степеням «двойки» выполнено в десятичной системе. То же самое разложение можно записать, используя только цифры двоичной системы

$$A_2 = a_{n-1} \ldots a_2a_1a_0 = (a_{n-1} \cdot 10^{n-1} + \ldots + a_2 \cdot 10^{10} + a_1 \cdot 10^1 + a_0 \cdot 10^0)_2. \quad (1.4)$$

Те же самые числа при использовании выражения (1.4) будут выглядеть следующим образом:
$101_2 = (1 \cdot 10^{10} + 0 \cdot 10^1 + 1 \cdot 10^0)_2$
и
$10100_2 = (1 \cdot 10^{100} + 0 \cdot 10^{11} + 1 \cdot 10^{10} + 0 \cdot 10^1 + 0 \cdot 10^0)_2.$

Таблицы умножения и сложения чисел и арифметические операции в двоичной системе счисления выполняются подобно тому, как это делается в десятичной системе, с той лишь разницей, что при этом используются свои таблицы умножения и сложения (табл. 1.2).

Таблица 1.2

Таблицы сложения и умножения двоичных чисел

Таблица сложения			Таблица умножения		
	0	**1**		**0**	**1**
0	0	1	**0**	0	0
1	1	10	**1**	0	1

Все арифметические операции могут выполняться столбиком.

Пример 1.1. Сложить числа 101 и 11011.

Решение:

```
        1 0 1 =   5
  +
    1 1 0 1 1 = 2 7
    - - - - -   - -
    1 0 0 0 0 = 3 2
```

Пример 1.2. Умножить числа 101 и 11011.

Решение:

```
          1 1 0 1 1 =   2 7
      ×
              1 0 1 =     5
        - - - - - -     - - -
          1 1 0 1 1     1 3 5
      1 1 0 1 1
    - - - - - - - -     - - -
    1 0 0 0 0 1 1 1 =   1 3 5
```

При работе с двоичными числами важную роль играют степени числа 2. Значения первых членов этого ряда, которые приведены в табл. 1.3, желательно помнить.

Таблица 1.3

Степени числа 2

n	2^n	n	2^n	n	2^n	n	2^n
0	1	6	64	12	4 096	18	262 144
1	2	7	128	13	8 192	19	524 288
2	4	8	256	14	16 384	20	1 048 576
3	8	9	512	15	32 768	21	2 097 152
4	16	10	1 024	16	65 536	22	4 194 304
5	32	11	2 048	17	131 072	23	8 388 608

Недостатком двоичной системы счисления является необходимость использования большого числа символов при записи даже сравнительно небольших чисел, что существенно затрудняет их восприятие человеком. Например, число 1567 записывается в двоичном виде как 11000011111, а число 8763 — как 10001000111011. Поэтому для интерпретации двоичной информации используется шестнадцатеричная система, запись чисел в которой значительно компактнее.

1.4. Шестнадцатеричная система счисления

Набор цифр для шестнадцатеричной системы счисления и их количественные эквиваленты приведены в табл. 1.4. Основание системы $p = 16$.

Таблица 1.4

Цифры шестнадцатеричной системы счисления и их количественные эквиваленты

Цифры	Количественный эквивалент в десятичной системе	Цифры	Количественный эквивалент в десятичной системе
0	0	8	8
1	1	9	9
2	2	A	10
3	3	B	11
4	4	C	12
5	5	D	13
6	6	E	14
7	7	F	15

Любое число в шестнадцатеричной системе представляет собой последовательность перечисленных символов. Для того чтобы подчеркнуть, что это именно шестнадцатеричная запись, в конце числа можно (но не обязательно) поставить нижний индекс 16 или символ h или H. Последнее обозначение является обязательным при задании шестнадцатеричных констант на языке Assembler, при этом к числу добавляется лидирующий нуль, чтобы различать числа и имена. Например,

$$161 = A1_{16} = 0A1h = 0A1H$$

или

$$1025 = 401_{16} = 401h.$$

Согласно формуле (1.2) число в шестнадцатеричной системе представляется в виде

$$A_{16} = a_{n-1}a_{n-2} \dots a_1a_0 = a_{n-1} \cdot (10_{16})^{n-1} + a_{n-2} \cdot (10_{16})^{n-2} + \\ + \dots + a_1 \cdot (10_{16})^1 + a_0 \cdot (10_{16})^0, \qquad (1.5)$$

где каждое a_i — одна из цифр {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, *A, B, C, D, E, F*}.

Например,
$10F_{16} = 1 \cdot (10_{16})^2 + 0 \cdot (10_{16})^1 + F \cdot (10_{16})^0.$

В формуле (1.5) разложение шестнадцатеричного числа по степеням «шестнадцати» выполнено в шестнадцатеричной системе. То же самое разложение можно выполнить в десятичной системе:

$A_{16} = a_{n-1}a_{n-2} \dots a_1a_0 = \alpha_{n-1} \cdot 16^{n-1} + \dots + \alpha_2 \cdot 16^2 + \alpha_1 \cdot 16^1 + \alpha_0 \cdot 16^0,$
где каждое α_i — количественный эквивалент соответствующей цифры a_i, записанный в десятичной системе.

Те же самые примеры при использовании выражения (1.4) будут выглядеть следующим образом:
$10F_{16} = 1 \cdot 16^2 + 0 \cdot 16^1 + 15 \cdot 16^0 = 256 + 15 = 271_{10}.$

Таблицы умножения и сложения чисел и арифметические операции в шестнадцатеричной системе счисления выполняются подобно тому, как это делается в десятичной системе, с той лишь разницей, что при этом используются свои таблицы умножения и сложения (табл. 1.5).

Таблица 1.5

Таблица умножения шестнадцатеричных чисел

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	***A***	***B***	***C***	***D***	***E***	***F***
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	*A*	*B*	*C*	*D*	*E*	*F*
2	0	2	4	6	8	*A*	*C*	*E*	10	12	14	16	18	1*A*	1*C*	1*E*
3	0	3	6	9	*C*	*F*	12	15	18	1*B*	1*E*	21	24	27	2*A*	2*D*
4	0	4	8	*C*	10	14	18	1*C*	20	24	28	2*C*	30	34	38	3*C*
5	0	5	*A*	*F*	14	19	1*E*	23	28	2*D*	32	37	3*C*	41	46	4*B*
6	0	6	*C*	12	18	1*E*	24	2*A*	30	36	3*C*	42	48	4*E*	54	5*A*

Окончание табл. 1.5

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	***A***	***B***	***C***	***D***	***E***	***F***
7	0	7	*E*	15	1*C*	23	2*A*	31	38	3*F*	46	4*D*	54	5*B*	62	69
8	0	8	10	18	20	28	30	38	40	48	50	58	60	68	70	78
9	0	9	12	1*B*	24	2*D*	36	3*F*	48	51	5*A*	63	6*C*	75	7*E*	87
A	0	*A*	14	1*E*	28	32	3*C*	46	50	5*A*	64	6*E*	78	82	8*C*	96
B	0	*B*	16	21	2*C*	37	42	4*D*	58	63	6*E*	79	84	8*F*	9*A*	*A*5
C	0	*C*	18	24	30	3*C*	48	54	60	6*C*	78	84	90	9*C*	*A*8	*B*4
D	0	*D*	1*A*	27	34	41	4*E*	5*B*	68	75	82	8*F*	9*C*	*A*9	*B*6	*C*3
E	0	*E*	1*C*	2*A*	38	46	54	62	70	7*E*	8*C*	9*A*	*A*8	*B*6	*C*4	*D*2
F	0	*F*	1*E*	2*D*	3*C*	4*B*	5*A*	69	78	87	96	*A*5	*B*4	*C*3	*D*2	*E*1

Таблица 1.6

Таблица сложения шестнадцатеричных чисел

	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	1*A*
C	*C*	*D*	*E*	*F*	10	11	12	13	14	15	16	17	18	19	1*A*	1*B*
D	*D*	*E*	*F*	10	11	12	13	14	15	16	17	18	19	1*A*	1*B*	1*C*
E	*E*	*F*	10	11	12	13	14	15	16	17	18	19	1*A*	1*B*	1*C*	1*D*
F	*F*	10	11	12	13	14	15	16	17	18	19	1*A*	1*B*	1*C*	1*D*	1*E*

2. ВЫПОЛНЕНИЕ АРИФМЕТИЧЕСКИХ ОПЕРАЦИЙ

2.1. Сложение

Пусть требуется сложить два числа A и B, записанных в системе счисления с основанием p:

$$A = a_{n-1}a_{n-2}\dots a_1a_0 = a_{n-1}p^{n-1} + a_{n-2}p^{n-2} + \dots + a_1p^1 + a_0p^0$$

и

$$B = b_{n-1}b_{n-2}\dots b_1b_0 = b_{n-1}p^{n-1} + b_{n-2}p^{n-2} + \dots + b_1p^1 + b_0p^0,$$

где p — основание системы счисления;

a_i, b_i — цифры этой системы счисления.

Результатом сложения этой пары чисел будет число C:

$$C = c_nc_{n-1}\dots c_1c_0 = c_np^n + c_{n-1}p^{n-1} + \dots + c_1p^1 + c_0p^0. \qquad (2.1)$$

Сумму этой пары чисел

$$\begin{aligned} C = A + B = (a_{n-1}p^{n-1} + a_{n-2}p^{n-2} + \dots + a_1p^1 + a_0p^0) + \\ + (b_{n-1}p^{n-1} + b_{n-2}p^{n-2} + \dots + b_1p^1 + b_0p^0) = \qquad (2.2) \\ = (a_{n-1} + b_{n-1})p^{n-1} + (a_{n-2} + b_{n-2})p^{n-2} + \dots + (a_1 + b_1)p^1 + (a_0 + b_0)p^0 \end{aligned}$$

будем определять в следующем порядке.

1-е действие. Определяем цифру c_0. Для этого складываем цифры нулевого разряда слагаемых:

$$a_0 + b_0 = s_1 c_0 = s_1 p^1 + c_0 p^0,$$

где s_1 — старшая цифра суммы цифр a_0 и b_0 и соответственно значение переноса в первый разряд для выражения (2.1);
c_0 — младшая цифра суммы цифр a_0 и b_0 и соответственно цифра нулевого разряда для выражения (2.1).

При этом s_1 может быть равным 0 или 1. Если $s_1 = 0$, то мы говорим, что перенос в первый разряд отсутствует. Если $s_1 = 1$, то мы говорим, что есть перенос единицы в первый разряд. Выражение (2.2) для суммы приобретает вид

$$S = (a_{n-1} + b_{n-1})p^{n-1} + (a_{n-2} + b_{n-2})p^{n-2} + \ldots + (a_1 + b_1 + s_1)p^1 + c_0 p^0.$$

2-е действие. Определяем цифру c_1. Для этого складываем цифры первого разряда слагаемых и значение переноса из нулевого разряда:

$$a_1 + b_1 + s_1 = s_2 c_1 = s_2 p^1 + c_1 p^0,$$

где s_2 — старшая цифра суммы цифр a_0 и b_0 и соответственно значение переноса во второй разряд для выражения (2.1);
c_1 — младшая цифра суммы цифр a_1, b_1 и s_1 и соответственно цифра первого разряда для выражения (2.1).

Выражение для суммы приобретает вид

$$S = (a_{n-1} + b_{n-1})p^{n-1} + (a_{n-2} + b_{n-2})p^{n-2} + \ldots + (a_2 + b_2 + s_2)p^2 + c_1 p^1 + c_0 p^0.$$

Так будем продолжать до тех пор, пока не дойдем до разряда (n-1).

Действие с номером n. Определяем цифры c_{n-1} и c_n. Для этого складываем цифры разряда ($n-1$) слагаемых и значение переноса из разряда ($n-2$):

$$a_{n-1} + b_{n-1} + s_{n-1} = s_n c_{n-1} = s_n p^1 + c_{n-1} p^0,$$

где s_n — старшая цифра суммы цифр a_0 и b_0 и соответственно значение переноса в разряд с номером n для выражения (2.1);
c_{n-1} — младшая цифра суммы цифр a_{n-1}, b_{n-1} и s_{n-1} и соответственно цифра разряда для выражения (2.1).

Выражение для суммы приобретает вид

$$C = s_n p^n + c_{n-1} p^{n-1} + \ldots + c_2 p^2 + c_1 p^1 + c_0 p^0.$$

Примем $c_n = s_n$ и окончательно получим

$$C = c_n c_{n-1} \ldots c_1 c_0 = c_n p^n + c_{n-1} p^{n-1} + \ldots + c_1 p^1 + c_0 p^0.$$

Эту же последовательность операций можно оформить в виде сложения столбиком:

$a_{n-1} a_{n-2} \ldots a_1 a_0$ — первое слагаемое;
$b_{n-1} b_{n-2} \ldots b_1 b_0$ — второе слагаемое;
$s_n s_{n-1} \ldots s_2 s_1$ — строка переносов;
———————————————-
$c_n c_{n-1} c_{n-2} \ldots c_1 c_0$ — сумма.

При сложении в десятичной системе строку переносов не пишем, а запоминаем c_i, если оно не равно 0, для учета при сложении цифр следующего разряда — «s_i пишем, 1 в уме».

Пример 2.1. Сложение двоичных чисел.

Требуется сложить два числа 110011_2 и 100001_2.

Сложение будем выполнять в 6 действий (по числу разрядов). Результаты каждого действия для наглядности будем оформлять в виде таблицы. Исходная таблица имеет нижеследующий вид.

Номер разряда	6	5	4	3	2	1	0
Первое слагаемое	0	1	1	0	0	1	1
Второе слагаемое	0	1	0	0	0	0	1
Значение переноса							
Сумма							

1-е действие. Складываем цифры нулевого разряда:

1 + 1 = 10.

Нуль записываем в нулевой разряд суммы, 1 — в значение переноса для 1-го разряда. Таблица примет нижеследующий вид.

Номер разряда	6	5	4	3	2	1	0
Первое слагаемое	0	1	1	0	0	1	1
Второе слагаемое	0	1	0	0	0	0	1
Значение переноса						1	
Сумма							0

2-е действие. Складываем цифры 1-го разряда и значение переноса:

1 + 0 + 1 = 10.

Нуль записываем в 1-й разряд суммы, 1 — в значение переноса для 2-го разряда. Таблица примет нижеследующий вид.

Номер разряда	6	5	4	3	2	1	0
Первое слагаемое	0	1	1	0	0	1	1
Второе слагаемое	0	1	0	0	0	0	1
Значение переноса					1	1	
Сумма						0	0

3-е действие. Складываем цифры 2-го разряда и значение переноса:

0 + 0 + 1 = 01.

Единицу записываем во 2-й разряд суммы, 0 — в значение переноса для 3-го разряда. Таблица примет нижеследующий вид.

Номер разряда	6	5	4	3	2	1	0
Первое слагаемое	0	1	1	0	0	1	1
Второе слагаемое	0	1	0	0	0	0	1
Значение переноса				0	1	1	
Сумма					1	0	0

4-е действие. Складываем цифры 3-го разряда и значение переноса:

$0 + 0 + 0 = 00.$

Ноль записываем в 3-й разряд суммы, 0 — в значение переноса для 4-го разряда. Таблица примет нижеследующий вид.

Номер разряда	6	5	4	3	2	1	0
Первое слагаемое	0	1	1	0	0	1	1
Второе слагаемое	0	1	0	0	0	0	1
Значение переноса			0	0	1	1	
Сумма				0	1	0	0

5-е действие. Складываем цифры 4-го разряда и значение переноса:

$1 + 0 + 0 = 01.$

Единицу записываем в 4-й разряд суммы, 0 — в значение переноса для 5-го разряда. Таблица примет нижеследующий вид.

Номер разряда	6	5	4	3	2	1	0
Первое слагаемое	0	1	1	0	0	1	1
Второе слагаемое	0	1	0	0	0	0	1
Значение переноса		0	0	0	1	1	
Сумма			1	0	1	0	0

6-е действие. Складываем цифры 5-го разряда и значение переноса:

$1+1+0 = 10.$

Ноль записываем в 5-й разряд суммы, 1 — в значение переноса для 6-го разряда. Поскольку у слагаемых 6-й разряд отсутствует, значение переноса (в данном случае 1) записываем в 6-й разряд суммы. Таблица примет нижеследующий вид.

Номер разряда	6	5	4	3	2	1	0
Первое слагаемое	0	1	1	0	0	1	1
Второе слагаемое	0	1	0	0	0	0	1
Значение переноса	1	0	0	0	1	1	
Сумма	1	0	1	0	1	0	0

Те же действия можно оформить в виде сложения столбиком:

```
   ·     · ·
   1 1 0 0 1 1
+
   1 0 0 0 0 1
   - - - - - -
 1 0 1 0 1 0 0
```

Точка над цифрами означает, что в данный разряд был совершен перенос единицы из младшего разряда.

Пример 2.2. Требуется сложить два числа $1FA27_{16}$ и $E3FA2_{16}$.

Сложение выполняем столбиком, так же как для десятичных чисел:

$$\dot{1}\dot{F}A27_{16}$$

$$E3EA2_{16}$$

$$1038C9_{16}$$

Точка над цифрами означает, что в данный разряд был совершен перенос единицы из младшего разряда.

То же самое сложение выполним подробно в 5 действий (по числу разрядов).

Исходная таблица имеет нижеследующий вид.

Номер разряда	5	4	3	2	1	0
Первое слагаемое		1	*F*	*A*	2	7
Второе слагаемое		*E*	3	*E*	*A*	2
Значение переноса						
Сумма						

1-е действие. Складываем цифры нулевого разряда. Согласно таблице сложения шестнадцатеричных чисел имеем

$$7_{16} + 2_{16} = 9_{16}.$$

9 пишем в нулевой разряд суммы, перенос отсутствует. В результате получим нижеследующую таблицу.

Номер разряда	5	4	3	2	1	0
Первое слагаемое		1	*F*	*A*	2	7
Второе слагаемое		*E*	3	*E*	*A*	2
Значение переноса						
Сумма						9

2-е действие. Складываем цифры первого разряда. Согласно таблице сложения шестнадцатеричных чисел имеем

$2_{16} + A_{16} = C_{16}$.

«*C*» пишем в первый разряд суммы, перенос отсутствует. В результате получим нижеследующую таблицу.

Номер разряда	5	4	3	2	1	0
Первое слагаемое		1	*F*	*A*	2	7
Второе слагаемое		*E*	3	*E*	*A*	2
Значение переноса						
Сумма					*C*	9

3-е действие. Складываем цифры второго разряда. Согласно таблице сложения имеем

$A_{16} + F_{16} = 18_{16}$.

8 пишем во второй разряд суммы, 1 пишем в значение переноса для третьего разряда, в результате чего получим нижеследующую таблицу.

Номер разряда	5	4	3	2	1	0
Первое слагаемое		1	*F*	*A*	2	7
Второе слагаемое		*E*	3	*E*	*A*	2
Значение переноса			1			
Сумма				8	*C*	9

4-е действие. Складываем цифры четвертого разряда и значение переноса. Согласно таблице сложения имеем

$$F_{16} + 3 + 1 = 13_{16}.$$

3 пишем в третий разряд суммы, 1 пишем в значение переноса для третьего разряда, в результате чего получим нижеследующую таблицу.

Номер разряда	5	4	3	2	1	0
Первое слагаемое		1	*F*	*A*	2	7
Второе слагаемое		*E*	3	*E*	*A*	2
Значение переноса		1	1			
Сумма			3	8	*C*	9

5-е действие. Складываем цифры пятого разряда и значение переноса. Согласно таблице сложения имеем

$$1 + E_{16} + 1 = 10_{16}.$$

0 пишем в четвертый разряд суммы, 1 пишем в значение переноса для пятого разряда. Поскольку у слагаемых 5-й разряд

отсутствует, значение переноса (в данном случае 1) записываем в пятый разряд суммы. Таблица примет нижеследующий вид.

Номер разряда	5	4	3	2	1	0
Первое слагаемое		1	*F*	*A*	2	7
Второе слагаемое		*E*	3	*E*	*A*	2
Значение переноса	1	1	1			
Сумма	1	0	3	8	*C*	9

Это окончательный результат.

2.2. Умножение

Пусть требуется умножить два числа A и B, записанных в системе счисления с основанием p:

$$A = a_{n-1}a_{n-2}\dots a_1a_0 = a_{n-1}p^{n-1} + a_{n-2}p^{n-2} + \dots + a_1p^1 + a_0p^0$$

и

$$B = b_{m-1}b_{m-2}\dots b_1b_0 = b_{m-1}p^{m-1} + b_{m-2}p^{-2} + \dots + b_1p^1 + b_0p^0,$$

где p — основание системы счисления;

a_i, b_i — цифры этой системы счисления;

n, m — число разрядов, использованных для представления первого и второго числа соответственно.

Результатом умножения этой пары чисел будет число C:

$$C = c_tc_{t-1}\dots c_1c_0 = c_tp^t + c_{t-1}p^{t-1} + \dots + c_1p^1 + c_0p^0,$$

где c_t — цифры системы счисления;

t — число разрядов, использованных для представления результата умножения.

Для того чтобы выразить представление числа C через числа A и B, запишем их произведение в следующем виде:

$$C = AB = A\,(b_{m-1}p^{m-1} + \ldots + b_1p^1 + b_0p^0) =$$

$$= Q_{m-1}p^{m-1} + Q_{m-2}p^{m-2} + \ldots + Q_1p^1 + Q_0p^0,$$

где Q_i — произведение первого сомножителя на цифру i-го разряда второго сомножителя (промежуточное произведение с номером i):

$$Q_i = Ab_i = q_{i,\,n}q_{i,\,n-1}\ldots\, q_{i,1}q_{i,0},$$

где $q_{i,\,k}$ — цифра k-го разряда записи числа Q_i.

Далее,

$Q_1p^1 = q_{1,\,n}q_{1,\,n-1}\ldots\, q_{1,1}q_{1,0}p^1 = q_{1,\,n}q_{1,\,n-1\ldots}\, q_{1,1}q_{1,0}0,$

$Q_2p^2 = q_{2,\,n}q_{2,\,n-1}\ldots\, q_{2,1}q_{2,0}p^2 = q_{2,\,n}q_{2,\,n-1\ldots}\, q_{2,1}q_{2,0}00,$

$Q_3p^3 = q_{3,\,n}q_{3,\,n-1\ldots}\, q_{3,1}q_{3,0}p^3 = q_{3,\,n}q_{3,\,n-1\ldots}\, q_{3,1}q_{3,0}000.$

$\ldots = \ldots.$

Таким образом, произведение двух чисел представимо в виде суммы промежуточных произведений первого сомножителя на количественный эквивалент каждой цифры второго сомножителя:

$$\begin{array}{cccccccccc}
 & & & & & a_{n-1} & a_{n-2} & \ldots & a_1 & a_0 \\
 & & & & \times & & & & & \\
 & & & & & b_{m-1} & b_{m-2} & \ldots & b_1 & b_0 \\
 & & & & \text{---} & \text{---} & \text{----} & \text{----} & \text{----} & \text{----} \\
 & & & & & q_{0,\,n} & q_{0,\,n-1} & \ldots & q_{0,1} & q_{0,0} \\
 & & & & q_{1,\,n} & q_{1,\,n-1} & \ldots & q_{1,1} & q_{1,0} & 0 \\
 & & & q_{3,\,n} & q_{2,\,n-1} & \ldots & q_{2,1} & q_{2,0} & 0 & 0 \\
 & & & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
 & q_{m-1,\,n} & q_{m-1,\,n-1} & \ldots & q_{m-1,1} & q_{m-1,0} & 0 & \ldots & 0 & 0 \\
\text{----} & \text{----} & \text{----} & \text{----} & \text{----} & \text{----} & \text{----} & \text{----} & \text{----} & \text{----} \\
c_t & c_{t-1} & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & c_1 & c_0
\end{array}$$

На практике нули, обусловленные умножением на основание системы счисления в какой-либо степени, не пишут, заменяя их сдвигом соответствующего числа влево на количество разрядов, равное степени основания. В результате запись операции умножения имеет вид

$$
\begin{array}{cccccccccc}
 & & & & & a_{n-1} & a_{n-2} & \dots & a_1 & a_0 \\
 & & & & \times & & & & & \\
 & & & & & b_{m-1} & b_{m-2} & \dots & b_1 & b_0 \\
 & & & & --- & --- & ---- & ---- & ---- & ---- \\
 & & & & & q_{0,\,n} & q_{0,\,n-1} & \dots & q_{0,1} & q_{0,0} \\
 & & & & q_{1,\,n} & q_{1,\,n-1} & \dots & q_{1,1} & q_{1,0} & \\
 & & & q_{3,\,n} & q_{2,\,n-1} & \dots & q_{2,1} & q_{2,0} & & \\
 & & & \dots & \dots & \dots & \dots & & & \\
 & q_{m-1,\,n} & q_{m-1,\,n-1} & \dots & q_{m-1,1} & q_{m-1,0} & & & & \\
---- & ---- & ---- & ---- & ---- & ---- & ---- & ---- & ---- & ---- \\
c_t & c_{t-1} & \dots & \dots & \dots & \dots & \dots & \dots & c_1 & c_0
\end{array}
$$

Пример 2.3. Умножение двоичных чисел.

Требуется умножить два числа 110011_2 и 100001_2. Выполняется достаточно просто, так как все промежуточные произведения либо равны первому сомножителю, либо равны нулю.

$$
\begin{array}{ccccccc l}
 & 1 & 1 & 0 & 0 & 1 & 1 & \text{–1-й сомножитель} \\
\times & & & & & & & \\
 & 1 & 0 & 0 & 0 & 0 & 1 & \text{— 2-й сомножитель} \\
 & - & - & - & - & - & - & \\
 & 1 & 1 & 0 & 0 & 1 & 1 & \text{— результат умножения 1-го числа на 1-ю цифру 2-го} \\
0 & 0 & 0 & 0 & 0 & 0 & & \text{— результат умножения 1-го числа на 2-ю цифру 2-го}
\end{array}
$$

```
      0 0 0 0 0 0     — результат умножения 1-го числа на 3-ю цифру 2-го
    0 0 0 0 0 0       — результат умножения 1-го числа на 4-ю цифру 2-го
  0 0 0 0 0 0         — результат умножения 1-го числа на 5-ю цифру 2-го
1 1 0 0 1 1           — результат умножения 1-го числа на 6-ю цифру 2-го
– – – – – – – – – – –
1 1 0 1 0 0 1 0 0 1 1 — произведение
```

Если удалить нулевые строчки и комментарии, то получится привычная запись умножения столбиком:

```
          1 1 0 0 1 1
        ×
          1 0 0 0 0 1
          – – – – – –
          1 1 0 0 1 1
1 1 0 0 1 1
– – – – – – – – – – –
1 1 0 1 0 0 1 0 0 1 1
```

3. ПЕРЕВОД ЧИСЕЛ ИЗ ОДНОЙ СИСТЕМЫ СЧИСЛЕНИЯ В ДРУГУЮ

3.1. Общие положения

Пусть имеется какое-либо число A. Оно может быть записано как в системе счисления с основанием p:

$$A = (a_{n-1}a_{n-2}\dots a_1a_0)_p = a_{n-1}p^{n-1} + a_{n-2}p^{n-2} + \dots + a_1p^1 + a_0p^0, \qquad (3.1)$$

так и в системе счисления с основанием q:

$$A = (b_{m-1}b_{m-2}\dots b_1b_0)_q = b_{m-1}q^{m-1} + b_{m-2}q^{m-2} + \dots + b_1q^1 + b_0. \qquad (3.2)$$

Для того чтобы перевести число из системы счисления с основанием p в систему счисления с основанием q, требуется по известным значениям цифр a_i выражения (3.1) определить значения цифр b_i выражения (3.2). При этом потребуется выполнить определенные вычисления, которые могут быть произведены как в системе счисления с основанием p, так и в системе счисления с основанием q. Эти вычисления выполняются по-разному — используются два способа перевода чисел из одной системы счисления в другую. Здесь мы ограничимся случаями перевода, при которых все расчеты выполняются в десятичной системе. Помимо этого рассмотрим перевод чисел из двоичной системы в шестнадцатеричную и наоборот, в которых вычисления вообще не требуются. Но в любом случае необходимо иметь таблицу соответствия цифр и основания исходной системы счисления их количественным эквивалентам, записанным цифрами системы счисления, в которую осуществляется переход.

Таблица 3.1

Цифры и основание шестнадцатеричной системы счисления и их количественные эквиваленты в двоичной и десятичной системах счисления

Цифры	Количественный эквивалент	
	Десятичная система	Двоичная система
0	0	0
1	1	1
2	2	10
3	3	11
4	4	100
5	5	101
6	6	110
7	7	111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111
10	16	10000

3.2. Перевод чисел в десятичную систему счисления

Пусть дано какое-либо число A, записанное в системе счисления с основанием p:

$$A = (a_{n-1}a_{n-2}\ldots a_1a_0)_p = a_{n-1}p^{n-1} + a_{n-2}p^{n-2} + \ldots + a_1p^1 + a_0p^0. \quad (3.3)$$

Для получения его записи в десятичной системе счисления достаточно в формуле (3.3) заменить все цифры a_{n-1}, a_{n-2}, ..., a_1, a_0 и основание системы p их количественными эквивалентами в системе счисления с основанием 10.

Пример 3.1. Требуется перевести число $1101\,1101_2$, записанное в двоичной системе, в десятичную. Расчеты выполнять в десятичной системе.

$$A = 1101\,1101_2 = 1\cdot2^0 + 0\cdot2^1 + 1\cdot2^2 + 1\cdot2^3 + 1\cdot2^4 + 0\cdot2^5 + 1\cdot2^6 + 1\cdot2^7 =$$
$$= 1 + 0 + 4 + 8 + 16 + 0 + 64 + 128 = 221_{10}.$$

Пример 3.2. Требуется перевести число $449F_{16}$, записанное в шестнадцатеричной системе, в десятичную. Расчеты выполнять в десятичной системе.

Количественный эквивалент основания шестнадцатеричной системы в десятичной системе записывается как $p = 16_{10}$, а в шестнадцатеричной системе — как $p = 10_{16.}$ Количественные эквиваленты цифр шестнадцатеричной системы приведены в табл. 3.1. На основании этого мы можем произвести следующие преобразования:

$$A = 449F_{16} = (F\cdot10^0 + 9\cdot10^1 + 4\cdot10^2 + 4\cdot10^3)_{16} = 15\cdot16^0 + 9\cdot16^1 + 4\cdot16^2 +$$
$$+ 4\cdot16^3 = 15 + 9\cdot16 + 4\cdot256 + 4\cdot4096 = 17567_{10}.$$

3.3. Перевод чисел из десятичной системы счисления

Пусть дано какое-либо число A, записанное в системе счисления с основанием 10:

$$A = (a_{n-1}a_{n-2}\dots a_1a_0)_{10} = a_{n-1}\cdot 10^{n-1} + a_{n-2}\cdot 10^{n-2} + \dots + a_1\cdot 10^1 + a_0\cdot 10^0.$$

Его требуется перевести в систему счисления с основанием p.

Расчет выполняется последовательным делением нацело с остатком исходного числа A на p_{10} — записанный в десятичной системе численный эквивалент основания системы, в которую мы переводим.

Разделим число A на p_{10}. В результате получим число A_0 (результат деления) и число c_0 (остаток от деления). На основании этого получим

$$A = A_0p_{10} + c_0. \quad (3.4)$$

В свою очередь, разделим число A_0 на p_{10}. В результате получим A_1 (результат деления) и c_1 (остаток). На основании этого получим

$$A_0 = A_1p_{10} + c_1. \quad (3.5)$$

Подставив (3.5) в (3.4), получим

$$A = A_0p_{10} + b_0 = (A_1p_{10} + c_1)p_{10} + c_0 = A_1(p_{10})^2 + c_1p_{10} + c_0.$$

Далее разделим число A_1 на p_{10}. В результате получим A_2 (результат деления) и c_2 (остаток от деления). Так будем продолжать до тех пор, пока какое-то A_m не станет равным нулю. В результате получим разложение числа A по степеням числа p, которое является основанием требуемой системы счисления

$$A = c_{m-1}(p_{10})^{m-1} + c_{m-2}(p_{10})^{m-2} + \dots + c_1(p_{10})^1 + c_0. \quad (3.6)$$

Каждое число c_i, входящее в (3.6), больше 0 и меньше p и является количественным эквивалентом какой-либо цифры b_i из системы счисления с основанием p. Поэтому мы вправе написать

$$A = (b_{m-1}p^{m-1} + b_{m-2}p^{m-2} + \dots + b_1p^1 + b_0)_p = (b_{m-1}b_{m-2} \dots b_1b_0)_p,$$

что и требовалось выполнить.

Пример 3.3. Требуется перевести число 221 в двоичную систему.

Делим 221 на 2, получим 110 и остаток $b_0 = 1$.
Делим 110 на 2, получим 55 и остаток $b_1 = 0$.
Делим 55 на 2, получим 27 и остаток $b_2 = 1$.
Делим 27 на 2, получим 13 и остаток $b_3 = 1$.
Делим 13 на 2, получим 6 и остаток $b_4 = 1$.
Делим 6 на 2, получим 3 и остаток $b_5 = 0$.
Делим 3 на 2, получим 1 и остаток $b_6 = 1$.
Делим 1 на 2, получим 0 и остаток $b_7 = 1$.
На основании этого пишем

$$221 = b_7b_6b_5b_4b_3b_2b_1b_0 = 1101\,1101_2.$$

Приведенные вычисления можно оформить несколько иначе (столбиком) так, как это показано ниже.

221	$\mid\underline{2}$				
$\underline{220}$	110	$\mid\underline{2}$			
		$\mid 2$			
1	$\underline{110}$	55	$\mid\underline{2}$		
			$\mid 2$		
	0	$\underline{54}$	27	$\mid\underline{2}$	
				$\mid 2$	
		1	$\underline{26}$	13	$\mid\underline{2}$
					$\mid 2$

1	12	6	|2		
			|2		
	1	6	3	|2	
				|2	
		0	2	1	|2
					|2
			1	0	0
				1	

Записав остатки от деления на 2 (выделены жирным) в обратном порядке, так же получим

$221 = 1101\,1101_2$.

Пример 3.4. Требуется перевести число 17567 в шестнадцатеричную систему. Расчеты выполнять в десятичной системе.

Основание шестнадцатеричной системы $p = 16$, количественные эквиваленты цифр шестнадцатеричной системы приведены в табл. 3.1.

Делим 17567 на 16, получим 1097 и остаток $b_0 = 15$, который является количественным эквивалентом шестнадцатеричной цифры F.

Делим 1097 на 16, получим 68 и остаток $b_1 = 9$, который является количественным эквивалентом шестнадцатеричной цифры 9.

Делим 68 на 16, получим 4 и остаток $b_2 = 4$, который является количественным эквивалентом шестнадцатеричной цифры 4.

Делим 4 на 16, получим 0 и остаток $b_3 = 4$, который является количественным эквивалентом шестнадцатеричной цифры 4.

На основании этого пишем

$$117567 = b_3 b_2 b_1 b_0 = 449F_{16}.$$

То же самое, записанное упрощенным способом (столбиком):

17567	|16__			
16	1097	|16		
156	96_	68	|16	
_144	137	64	4	|16
127	_128	4	0	0
_112	9		4	
15				

Учитывая, что $15_{10} = F_{16}$, и записав остатки от деления на 16 (выделены жирным) в обратном порядке, так же получим

$117567 = 49F_{16}$.

3.4. Перевод чисел из шестнадцатеричной системы счисления в двоичную и наоборот

Перевод основан на взаимно-однозначном соответствии шестнадцатеричных цифр и двоичных тетрад (групп из четырех двоичных цифр). Такое соответствие приведено в табл. 3.2.

Таблица 3.2

Цифры шестнадцатеричной системы счисления и соответствующие им двоичные тетрады

Шестнадцатеричная цифра	Двоичная тетрада
0	0000
1	0001
2	0010
3	0011
4	0100

Окончание табл. 3.2

Шестнадцатеричная цифра	Двоичная тетрада
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

Для перевода чисел из шестнадцатеричной системы счисления в двоичную достаточно вместо шестнадцатеричной цифры вставить соответствующую ей двоичную тетраду.

Для перевода чисел из двоичной системы счисления в шестнадцатеричную необходимо двоичное число разбить на тетрады, выполняя разбиение справа налево. Если тетрада получается неполной, то ее необходимо дополнить нулями с левой стороны. Далее следует каждую тетраду заменить шестнадцатеричной цифрой согласно табл. 3.2.

Пример 3.5. Требуется перевести число 110010010110101_2 в шестнадцатеричную систему.

Разбиваем число на тетрады, к левой неполной тетраде дописываем слева нуль и заменяем тетрады шестнадцатеричными цифрами:

0110	0100	1011	0101
6	4	C	5

В результате имеем:

$110010010110101_2 = 64C5_{16}$.

Пример 3.6. Требуется перевести число $1F7A_{16}$ в двоичную систему.

Вместо каждой шестнадцатеричной цифры пишем соответствующую ей двоичную тетраду:

1	F	7	A
0001	1111	0111	1010

Лидирующие нули отбрасываем. В результате имеем:

$1F7A_{16} = 1111101111010_2$.

4. ВЕЩЕСТВЕННЫЕ ЧИСЛА В РАЗЛИЧНЫХ СИСТЕМАХ СЧИСЛЕНИЯ

4.1. Представление вещественных чисел в различных системах счисления

Представление целых чисел в позиционных системах счисления (см. подглаву 1.1) естественным образом расширяется для любых вещественных, т. е. имеющих дробную часть чисел. А именно, любое вещественное число может быть единственным способом представлено в виде

$$A = a_{n-1}a_{n-2}\dots a_1\, a_0, a_{-1}a_{-2}\dots a_{-m} = a_{n-1}p^{n-1} + a_{n-2}p^{n-2} + \dots + a_1p^1 + a_0p^0 +$$

$$+ a_{-1}p^{-1} + a_{-2}p^{-2} + \dots + a_{m-1}p^{m-1} + a_{-m}p^{-m}, \qquad (4.1)$$

где p — основание системы счисления;

n — число разрядов в целой части числа;

m — число разрядов в дробной части числа;

a_i, a_k — цифры данной системы счисления ($0< a_i, a_k<p$).

Например,

$193{,}74 = 1\cdot 10^2 + 9\cdot 10^1 + 3\cdot 10^0 + 7\cdot 10^{-1} + 4\cdot 10^{-2}$.

При заданном основании системы счисления каждому вещественному числу соответствует единственное разложение вида (4.1) и, наоборот, каждому представлению вида (4.1) соответствует единственное вещественное число. Сказанное верно с двумя оговорками. Во-первых, лидирующие нули не учитываются, т. е. 000567,1 = 567,1. Во-вторых, числа, представляемые бесконечной периодической дробью с цифрой (p-1) в периоде, эквивалентны 1:

$$(0{,}99999999999999...)_{10} = 1{,}0_{10}$$

или

$$(0{,}111111111111...)_2 = 1{,}0.$$

4.2. Перевод вещественных чисел из одной системы счисления в другую

В данном случае ограничимся рассмотрением перевода чисел из десятичной системы счисления в двоичную и наоборот. При переводе вещественных чисел обычно отдельно переводят целую и дробную части. Перевод целой части рассмотрен в главе 3. Поэтому здесь ограничимся рассмотрением перевода дробных частей.

Перевод из двоичной системы счисления в десятичную выполняется непосредственно по формуле (4.1).

Пример 4.1. Требуется перевести число $0{,}101001_2$ в десятичную систему.

$$0{,}101001_2 = 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} + 0 \cdot 2^{-4} + 0 \cdot 2^{-5} + 1 \cdot 2^{-6} =$$

$$= 0{,}5 + 0{,}125 + 0{,}015625 = 0{,}640625.$$

Перевод из десятичной системы счисления в двоичную выполняется методом последовательного умножения на 2, т. е. на основание системы счисления, в которую выполняется перевод. Перевод осуществляется в следующем порядке.

Дробная часть числа умножается на 2. Если результат умножения меньше 1, то в дробную часть двоичного представления записывается 0. Если полученное произведение больше 1, то в дробную часть двоичного представления записывается 1.

Дробная часть полученного произведения снова умножается на 2. Если результат умножения меньше 1, то в дробную часть двоичного представления записывается 0. Если полученное произведение больше 1, то в дробную часть двоичного представления записывается 1.

Процесс продолжается до тех пор, пока дробная часть не станет равной нулю или будет достигнута требуемая точность (в принципе процесс может продолжаться бесконечно).

Пример 4.2. Требуется перевести число 0,406 в двоичную систему счисления. Результаты последовательного умножения сведем в приведенную ниже таблицу.

Результаты умножения	Двоичное представление числа
$0{,}406 \cdot 2 = 0{,}812$	0,0
$0{,}812 \cdot 2 = 1{,}624$	0,01
$0{,}624 \cdot 2 = 1{,}248$	0,011
$0{,}248 \cdot 2 = 0{,}496$	0,0110
$0{,}496 \cdot 2 = 0{,}992$	0,01100
$0{,}992 \cdot 2 = 1{,}984$	0,011001
$0{,}984 \cdot 2 = 1{,}968$	0,0110011
$0{,}968 \cdot 2 = 1{,}936$	0,01100111

Процесс в принципе можно было продолжать дальше. В данном случае двоичная дробь будет бесконечной.

4.3. Экспоненциальная форма представления вещественных чисел

Экспоненциальная запись — представление действительных чисел в виде мантиссы и порядка. Удобна при представлении очень больших и очень малых чисел, а также для унификации их написания. Для представления вещественного числа используется следующая формула:

$$A = (\pm M) \cdot N^{\pm p}, \tag{4.2}$$

где M — мантисса числа A;

N — основание системы счисления;

p — порядок числа.

Используя формулу (4.2), одно и тоже число можно записать несколькими способами. Например,

$101{,}325 = 0{,}101325 \cdot 10^3$,

$101{,}325 = 1{,}01325 \cdot 10^2$,

$101{,}325 = 0{,}0101325 \cdot 10^4$,

$101{,}325 = 1013{,}25 \cdot 10^{-1}$ и т. д.

Ввиду того что положение точки, разделяющей целую и дробные части, зависит от показателя степени, такие числа частот называют числами с плавающей точкой (запятой). Для внесения определенности вводится понятие нормализованной записи числа в экспоненциальной форме. В инженерной практике (в том числе в информатике) мантисса в нормализованном виде должна удовлетворять соотношению

$$\frac{1}{N} \leqslant M < 1,$$

т. е. мантисса не имеет целой части, а первая цифра справа от точки, разделяющей дробную и целую части, не равна нулю.

Экспоненциальная форма представления вещественных чисел служит основой для их представления в ЭВМ.

5. ПРЕДСТАВЛЕНИЕ ЦЕЛЫХ ЧИСЕЛ В ЭВМ

Целые числа в компьютере представляются в двоичной системе счисления. Но форма их представления в компьютере отличается от формы их представления в обычной арифметике. Это обусловлено тем, что в обычной арифметике для записи числа мы можем использовать теоретически бесконечное число разрядов. В компьютере же мы можем для представления каждого числа использовать лишь фиксированное число разрядов (бит) и, стало быть, использовать ограниченный диапазон чисел. Число разрядов, используемое для представления числа, называется его форматом. Чем больше формат, тем больше диапазон представляемых чисел. Но вместе с тем с увеличением формата увеличивается объем памяти, требуемый для записи одного числа. Поэтому для записи целых используется несколько форматов:

- однобайтовый (8 разрядов или байт);
- двухбайтовый (16 разрядов или слово или 2 байта);
- четырехбайтовый (двойное слово или 4 байта).

Помимо этих четырех основных форматов могут использоваться и другие (64 бита, например).

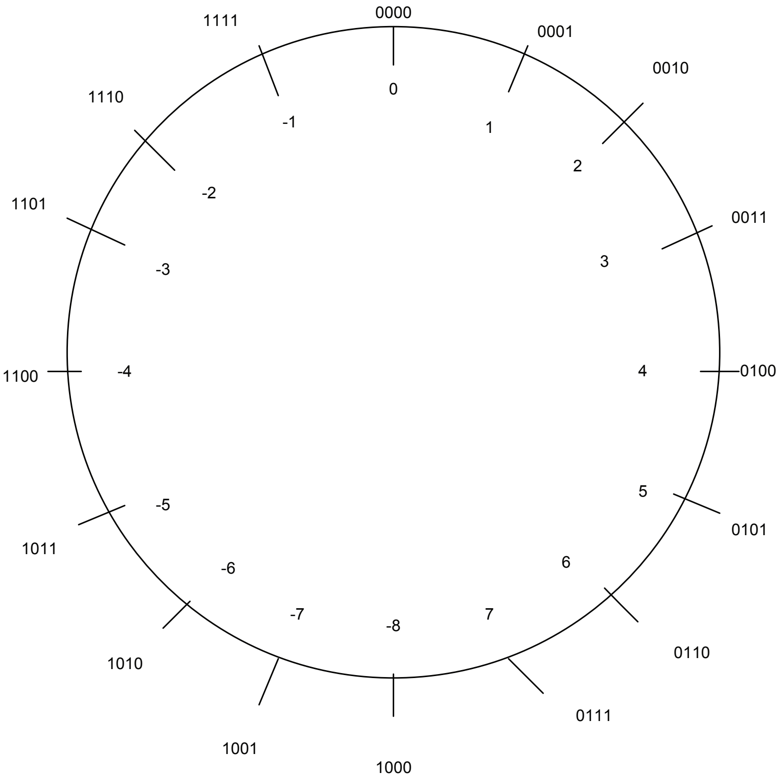
Для целых чисел существуют два представления: беззнаковое (только для неотрицательных целых чисел) и со знаком. Очевидно, что отрицательные числа можно представлять только в знаковом виде. В беззнаковом формате все разряды предназначены для записи значения числа, при представлении чисел со знаком старший (самый левый) бит предназначен для знака. Если старший бит числа равен нулю, то это — положительное число. Если старший бит числа равен единице, то это — отрицательное число. Положительное число представляется его прямым кодом, отрицательное — дополнительным кодом.

Прямой код — старший бит кода равен нулю, остальные биты представляют двоичное представление числа.

Дополнительный код получается из прямого путем инверсии (замена нулей единицами, а единиц нулями) с последующим добавлением единицы.

Использование дополнительного кода для представления отрицательных чисел может показаться надуманным, однако это не так. Предположим (для краткости), что у нас имеется четырехразрядный формат. Возьмем окружность и разобьем ее на 16 дуг (см. рисунок на с. 43) [9]. Расположим по окружности все возможные комбинации нулей и единиц в четырех разрядах (тетрады), как это показано на рисунке. Пронумеруем тетрады по часовой стрелке положительными числами от 0 до 15 и против часовой от 0 до –15 (на рисунке показаны только цифры от 0 до 7 и от 0 до –8). Тогда каждой тетраде будет соответствовать как положительное, так и отрицательное число. Например, тетраде 1111 соответствует + 15 и –1. Тетрада, соответствующая положительному числу, является его прямым кодом, а тетрада, соответствующая отрицательному числу, является его обратным кодом. Если мы используем беззнаковый формат, то все тетрады представляют положительные числа, и $1111_2 = 15$. Если же мы используем знаковый формат, то для положительных чисел используем правую полуокружность, а для отрицательных — левую, так как показано на рисунке. В этом случае $1111_2 = -1$.

Аналогичные рассуждения можно провести и для 8-разрядного формата, и для любого другого.



Пояснение представления знаковых форматов

Пример 5.1. Прямой код числа 1 в однобайтовом формате представляется как 0000 0001, его инверсия 1111 1110, его дополнительный код равен 1111 1110 + 1 = 1111 1111. Таким образом, —1 представляется как 1111 1111.

Пример 5.2. Число 5 в однобайтовом беззнаковом формате представляется как $5_{10} = 0000\,0101_2$, в однобайтовом знаковом формате оно будет представляться также $5_{10} = 0000\,0101_2$.

Число –5 в однобайтовом беззнаковом формате представить нельзя. Для представления числа –5 в однобайтовом знаковом формате необходимо определить дополнительный код числа 5, который и будет представлять число –5.

0000 0101 — прямой код числа 5;

1111 1010 — инверсия кода числа 5;

1111 1010 + 0000 0001 = 1111 1011 — дополнительный кода числа 5.

Откуда получим $-5_{10} = 1111\,1011_2$.

Наличие нескольких форматов позволяет представлять одно и то же число по-разному. Например, число 1 можно представить как 0000 0001 или как 0000 0000 0000 0001 (см. таблицу ниже). Процессор ЭВМ арифметические операции с числами разных форматов не выполняет. Для возможности выполнения арифметической операции с числами, записанными в разных форматах, нужно предварительно, преобразовать их к одному формату. При переходе от более короткого формата к более длинному для положительных чисел слева дописываются нули, для отрицательных дописываются единицы. С одной стороны, если запись 1100 1111 является кодом беззнакового числа, то его расширением до двухбайтового формата будет 0000 0000 11001111. Если же запись 1100 1111 является кодом числа со знаком, то его расширением до двухбайтового формата будет 1111 1111 1100 1111. С другой стороны, один и тот же код может представлять разные числа. Например, 1111 1011 может обозначать –5 либо 251 в зависимости от того, как мы его рассматриваем — как число со знаком или число без знака. Арифметические операции с числами, одно из которых закодировано как число со знаком, а другое без, выполнять нельзя. Их нужно предварительно преобразовать к одному типу.

Диапазоны представления целых чисел в различных форматах

Формат	Целое без знака	Целое со знаком
Байт	0...255	–128... + 127
Слово	0...65535	–32768...32767
Двойное слово	0...4294967295	–2147483648...2147483647

Порядок байтов. При записи двоичных чисел их разряды (биты) нумеруются справа налево. При разбиении числа на байты последние также нумеруются справа налево:

$$B_3B_2B_1B_0.$$

Однако при записи в оперативную память байты нумеруются слева направо:

$$B_0B_1B_2B_3.$$

Говорят, что числа, хранящиеся в оперативной памяти, имеют обратный порядок байтов. Нумерация битов внутри байта при этом остается прежней.

6. ПРЕДСТАВЛЕНИЕ ВЕЩЕСТВЕННЫХ ЧИСЕЛ В ЭВМ

Для представления вещественного числа используется следующая формула [10]:

$$A = (-1)^s \cdot 2^p m,$$

где m — нормализованная мантисса числа A;

p — порядок числа;

s — показатель знака числа.

Используется три формата представления кода числа — короткий, длинный и расширенный. Состоит из трех полей:

s	***q***	*mantissa*

где s — поле знака, занимающее 1 бит, $s=0$ — число положительное, $s=1$ — число отрицательное;

q — поле характеристики, размер которого зависит от формата и определяет значение показателя степени p;

mantissa — поле мантиссы, размер которого зависит от формата и определяет значение мантиссы.

Короткий формат

Размер записи 32 бита.

Поле s занимает 31-й бит.

Поле q занимает биты с 23-го по 30-й, размер поля 8 бит. Поле предназначено для записи характеристики числа, которая в данном формате связана с показателем соотношением $q = p + 127$.

Диапазон значений характеристики в данном формате от 0 до 255, соответственно диапазон значений показателя от –127 до 128.

Поле mantissa занимает биты с 0-го по 22-й, размер поля 23 бит. Поле предназначено для записи мантиссы. Ввиду того что первый бит нормализованной мантиссы всегда равен 1, его не запоминают, а при выполнении операций процессор добавляет его автоматически. Таким образом, реальная мантисса числа формируется как

$$m = 0{,}1<\text{mantissa}>,$$

т. е. реально под мантиссу отводится 24 бита — 23 в поле мантиссы плюс 1 бит по умолчанию.

Аналогично представляются числа в длинном и расширенном форматах, с оговоркой, что в расширенном формате единицы по умолчанию нет. Характеристика форматов вещественных чисел приведена ниже в таблице [7].

Форматы вещественных чисел

Формат	Короткий	Длинный	Расширенный
Длина числа (бит)	32	64	80
Мантисса М	24	53	64
Диапазон значений	10–38...10 + 38	10–308...10 + 308	10–4932 ...10 + + 4932
Длина поля характеристики (бит)	8	11	15
Диапазон значений характеристики q	0...255	0...2047	0...32767
Диапазон порядков p	–127... + 128	–1022... + 1023	–6382... + 16383

Пример. Перевести машинный код действительного числа, записанный в коротком формате.

Код в двоичном виде: 1**100 0000 1**110 101111110 010111101010.

Показатель знака числа $s = 1$ (нормальный шрифт).

Характеристика $q = 1000\,0001_2 = 129$ (в представлении кода — выделено жирным шрифтом).

Показатель степени $p = q - 127 = 2$.

Код мантиссы mantissa = 110 1011 1110 0101 1110 1010.

Мантисса m = 0.1<mantissa> = 0.1110 1011 1110 0101 1110 1010.

Число в двоичном виде в экспоненциальной форме

$$A_1 = (-1)^s \cdot 2^p m = (-1)^1 \cdot 2^1 \cdot 0{,}1110\,1011\,1110\,0101\,1110\,1010.$$

Порядок байтов. Вещественные числа, хранящиеся в оперативной памяти, так же как и целые числа, имеют обратный порядок байт (см. главу 5).

ЗАКЛЮЧЕНИЕ

В данном пособии дано описание позиционных систем счисления, показаны правила выполнения арифметических операций, описаны методы перевода чисел из одной системы счисления в другую, показано представление как целых, так и дробных чисел. Помимо этого рассмотрены методы представления числовой информации в ЭВМ. За рамками пособия остались способы кодирования и представления в ЭВМ символьной, графической и другой информации. Однако принципы ее кодирования базируются на основе использования двоичной и шестнадцатеричной систем счисления. Поэтому можно считать, что материал пособия дает хороший фундамент для освоения методов кодирования любой информации в любых типах компьютеров.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Фомин С. В. Системы счисления / С. В. Фомин. — М. : Наука, 1987.

2. Гашков С. Б. Системы счисления и их применение / С. Б. Гашков. — М. : МЦНМО, 2004.

3. Яглом И. Системы счисления / И. Яглом // Квант. — 1970. — № 6.

4. Микушин А. В. Цифровые устройства и микропроцессоры / А. В. Микушин А. М. Сажнев, В. И. Сединин. — СПб. : БХВ-Петербург, 2010.

5. Угрюмов Е. П. Цифровая схемотехника / Е. П. Угрюмов. — СПб. : БХВ-Петербург, 2004.

6. Шило В. Л. Популярные цифровые микросхемы / В. Л. Шило. — М. : Радио и связь, 1987.

7. Юров В. И. Assembler / В. И. Юров. — СПб. : Питер, 2003.

8. Танебаум Э. Архитектура компьютера / Э. Танебаум. — СПб. : Питер, 2010.

9. Хамахер К. Организация ЭВМ / К. Хамахер, З. Вранешич, С. Заки. — СПб. : Питер, 2003.

10. IEEE Standard for Binary Floating-Point Arithmetic / The Institute of Electrical and Electronics, Inc. — N. Y. (USA), 1985.

ОГЛАВЛЕНИЕ

Учебное издание

Шаманов Анатолий Павлович

СИСТЕМЫ СЧИСЛЕНИЯ И ПРЕДСТАВЛЕНИЕ ЧИСЕЛ В ЭВМ

Редактор *И. В. Коршунова*
Верстка *Е. В. Ровнушкиной*

Подписано в печать 26.04.2016. Формат 60×84 1/16.
Плоская печать. Усл. печ. л. 3,02. Гарнитура *Newton*.
Уч.-изд. л. 2,37. Тираж 100 экз. Заказ 130.

Издательство Уральского университета
Редакционно-издательский отдел ИПЦ УрФУ
620049, Екатеринбург, ул. С. Ковалевской, 5
Тел.: 8 (343) 375-48-25, 375-46-85, 374-19-41
E-mail: rio@urfu.ru

Отпечатано в Издательско-полиграфическом центре УрФУ
620075, Екатеринбург, ул. Тургенева, 4
Тел.: 8 (343) 350-56-64, 350-90-13
Факс: 8 (343) 358-93-06
E-mail: press-urfu@mail.ru

ЛИДЕР УНИВЕРСИТЕТСКОГО КНИГОИЗДАНИЯ
УНИВЕРСИТЕТСКАЯ КНИГА
2015

9 785799 617196