Глава 1

Представление, измерение и преобразование информации

1.1. Системы счисления

Под **системой счисления** понимается способ представления любого числа с помощью некоторого алфавита символов, называемых **цифрами**. Системы счисления бывают позиционными и непозиционными.

В **позиционных системах счисления** значимость (вес) каждой цифры числа зависит от позиции, которую она занимает. Значение числа, состоящего из **n** цифр, может быть определено следующим образом:

$$(x_{n-1}\ x_{n-2}\ x_{n-3}\ x_{n-4} \ldots x_1 x_0) = x_{n-1} \cdot m^{n-1} + x_{n-2} \cdot m^{n-2} + \ldots + x_0 \cdot m^0,$$

где m — основание системы;

x_i — символ в i-й позиции, $0 \le x_i < m$; $0 \le i \le (n-1)$;

m^i — вес i-го знакоместа.

Для **десятичной системы счисления** $m = 10$, используемые символы: $0 \div 9$.

$$563_{10} = 5 \cdot 10^2 + 6 \cdot 10^1 + 3 \cdot 10^0$$

I	2	1	0
x_i	5	6	3
m^i	100	10	1
$x_i \cdot m^i$	500	60	3

Кроме десятичной системы широкое распространение получили позиционные системы счисления с основаниями 2, 8, 16, 60.

Из **непозиционных систем** самой распространенной является **римская.**

Электронные блоки компьютера могут обрабатывать информацию, представленную только в цифровой форме, причем обычно компьютеры работают **в двоичной системе счисления.** Основание системы: $m = 2$. Используемые символы: 1 и 0.

С точки зрения электроники значение единицы может быть представлено наличием напряжения, потенциала или тока, а ноль — отсутствием их.

Рассмотрим представление чисел в двоичной системе. Веса знакомест: $2^0 = 1$, $2^1 = 2$, $2^2 = 4$, $2^3 = 8$, $2^4 = 16$, $2^5 = 32$, $2^6 = 64$, $2^7 = 128$, $2^8 = 256$, $2^{10} = 1024$, $2^{16} = 65536$.

1.2. Перевод числа из десятичной системы в двоичную

Перевод числа из десятичной системы в двоичную осуществляется отдельно для целой и дробной частей числа по следующим алгоритмам:

а) целое десятичное число делится нацело на основание 2, затем на 2 делятся последовательно все частные от целочисленного деления, до тех пор пока частное не станет меньше основания. В результат заносится последнее частное и все остатки от деления, начиная с последнего (рис. 1.1). $227_{10} = 11100011_2$;

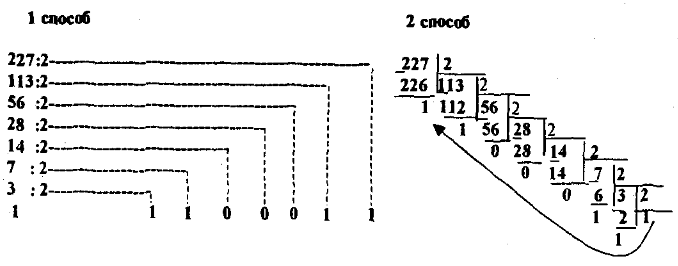

Рис. 1.1. Перевод числа из десятичной системы в двоичную

б) десятичная дробь последовательно умножается на основание 2, причем сразу после каждой операции умножения полученная целая часть записывается в результат и в дальнейшем умножении не участвует. Количество операций умножения зависит от требуемой точности, например, $0.64_{10} = 0.10100011_2$

$0.64 \cdot 2$

$1.28 \cdot 2$

$0.56 \cdot 2$

$1.12 \cdot 2$

$0.24 \cdot 2$
$0.48 \cdot 2$
$0.96 \cdot 2$
$1.92 \cdot 2$
$1.84 \cdot 2$

1.3. Перевод числа из двоичной системы в десятичную

Перевод числа из двоичной системы в десятичную можно осуществлять для целой и дробной частей числа по одному алгоритму путем вычисления суммы произведений цифры двоичного числа на вес ее знакоместа:

$$11100011_2 = 1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + \\ + 1 \cdot 2^0 = 128 + 64 + 32 + 2 + 1 = 227_{10}$$

$$0{,}10100011_2 = 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} + 0 \cdot 2^{-4} + 0 \cdot 2^{-5} + 0 \cdot 2^{-6} + \\ + 1 \cdot 2^{-7} + 1 \cdot 2^{-8} = 0.5 + 0.125 + 0.0078 + 0.0039 = 0.6367_{10}$$

1.4. Представление в компьютере отрицательных чисел

Следует иметь в виду, что в памяти ПЭВМ двоичные числа хранятся в регистрах, состоящих из 8 ячеек, т.е. минимальное двоичное число, которое можно разместить в памяти, должно быть восьмиразрядным. При этом в незаполненных ячейках регистра (в старших разрядах) записываются нули.

В отличие от десятичной системы в двоичной системе счисления отсутствуют специальные символы, обозначающие знак числа: положительный (+) или отрицательный (–), поэтому для представления двоичных отрицательных чисел используются следующие две формы.

Форма значения со знаком — старший (левый) разряд метится как знаковый и содержит информацию только о знаке числа:

1 — число отрицательное;

0 — число положительное.

Остальные разряды отводятся под абсолютную величину числа.

$5_{10} = 0000\ 0101_2$ $-5_{10} = 1000\ 0101_2$.

Форма обратного дополнительного кода, перевод в которую производится по следующему алгоритму:

1) инвертировать все разряды числа, кроме знакового разряда;

2) прибавить единицу к полученному коду;

3) восстановить единицу в знаковом разряде.

Преобразование числа

$-5_{10} = 1000\,0101 \rightarrow 111\,1010 + 1 \rightarrow 111\ 1011 \rightarrow 1111\,1011.$

Устройство компьютера выполняется таким образом, чтобы отрицательные числа были представлены в дополнительном коде, поскольку это дает существенную экономию времени при выполнении с ними арифметических операций.

Основные свойства дополнительных кодов:

1. Дополнительный код положительного числа — само число.
2. Преобразование дополнительного кода по приведенному алгоритму перевода приводит к первоначальному виду числа в знаковой форме.

1.5. Правила выполнения арифметических операций в двоичной системе

Сложение. Операция сложения выполняется так же, как и в десятичной системе. Переполнение разряда приводит к появлению единицы в следующем разряде:

$$0+0=0,\quad 0+1=1,\quad 1+1=10;\qquad \begin{array}{r} 11110011 \\ +\ \ 111011 \\ \hline 100101110 \end{array}$$

Вычитание. Поскольку большинство современных компьютеров располагает только одним аппаратным сумматором, с помощью которого реализуются все арифметические операции, вычитание сводится к сложению с отрицательным числом:

$15 - 8 = 15 + (-8).$

Правила вычитания в двоичной системе. Алгоритм операции вычитания путем сложения дополнительных кодов:

1) преобразовать отрицательное число из формы со знаком в дополнительный код;

2) выполнить операцию двоичного сложения над всеми разрядами, включая знаковый, игнорируя единицу переноса из самого высокого разряда;

3) при равенстве единице знакового разряда суммы, что означает получение отрицательного результата в форме дополнительного кода, необходимо перевести результат в знаковую форму, используя второе свойство дополнений.

13-15=13+(-15) 1)$-15_{10}=10001111 \rightarrow 1110000+1 \rightarrow 1110001 \rightarrow 11110001$

2)
$$\begin{array}{r} 00001101 \\ +\ 11110001 \\ \hline 11111110 \end{array}$$

3) $1111\ 1110 \rightarrow 000\ 0001+1 \rightarrow 1000\ 0010=2_{10}$

Таким образом, при выполнении операций сложения и вычитания арифметико-логическому устройству процессора приходится выполнять поразрядное сложение с переносом, инвертирование и проверку на знак двоичных чисел.

В тех случаях, когда необходимо произвести арифметические действия над числами больше 127, они размещаются уже не в одном, а в двух и более регистрах.

Умножение. Если наряду с перечисленными операциями выполнить операции сдвига, то с помощью сумматора можно выполнить и умножение, которое сводится к серии повторных сложений. Если цифра в нулевой позиции множителя равна 1, то множимое переписывается под соответствующими разрядами, умножение на последующие единицы приводят к сдвигу слагаемого влево на одну позицию. Если цифра множителя равна 0, то следующее слагаемое смещается на две позиции влево.

$$15_{10} \cdot 13_{10}=195_{10}=11000011_2=1 \cdot 2^7+1 \cdot 2^6+1 \cdot 2^1+1 \cdot 2^0=195_{10}$$

$$\begin{array}{r} 00001111 \\ \times\ 00001101 \\ \hline 00001111 \\ +\ 00001111 \\ +\ 00001111 \\ \hline 00011000011 \end{array}$$

Деление. При выполнении операции деления несколько раз производится операция вычитания. Поэтому предварительно следует найти дополнительный код делителя. Деление выполняется путем повторного вычитания и сдвига. Для примера выполним деление числа 195 на

15 или в двоичной системе 11000011_2 на 1111_2. Дополнительный код числа 1111 → 11110001. Поскольку по правилам деления каждое промежуточное делимое должно быть больше делителя, выбираем в качестве первого делимого число 11000, т.е. первые пять разрядов и добавляем слева три нуля, дополняя делимое до 8 разрядов. Затем производим сложение его с дополнительным кодом делимого и заносим в результат единицу. Если следующее делимое после сноса очередной цифры будет меньше делителя, то в результат заносится нуль и в делимое сносится еще одна цифра из исходного делимого.

```
 00011000011 |1111
+11110001    |1101
  00010010
 +11110001
    00001111
   +11110001
    00000000
```

Делимое 111 на третьем шаге после сложения и сноски очередного разряда меньше делителя, поэтому записываем в результат 0 и сносим еще один разряд из оставшихся в делимом. После третьего шага результат сложения равен 0, деление закончено.

Ответ: $00001101_2 = 13_{10}$

1.6. Использование восьмеричной и шестнадцатеричной систем счисления

Двоичная система счисления неудобна для использования человеком, поэтому программисты используют восьмеричную (основание 8, используемые символы $0 \div 7$) и шестнадцатеричную (основание 16, используемые символы $0 \div 9$, $A \div F$) системы (табл. 1.1).

Таблица 1.1

Позиционные системы счисления

Десятичная	Двоичная	Восьмеричная	Шестнадцатеричная
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7

Десятичная	Двоичная	Восьмеричная	Шестнадцатеричная
8	1000	10	8
9	1001	11	9
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

Каждая тройка двоичных разрядов соответствует одной восьмеричной цифре, а каждая четверка — шестнадцатеричной. Отсюда следует простота преобразований из двоичной системы в восьмеричную и шестнадцатеричную системы счисления.

Например:

$$11010011_2 = 1101\ 0011_2 = D3_{16}$$
$$11010011_2 = 011\ 010\ 011_2 = 323_8.$$

Если исходное количество бит не кратно 3 или 4, добавляются нули слева.

Обратное преобразование аналогично:

$$B9_{16} = 1011\ 1001_2$$
$$270_8 = 10\ 111\ 000_2.$$

Перевод из десятичной системы в m-ричную систему счисления производится аналогично переводу в двоичную систему путем целочисленного деления десятичного числа на основание системы m до тех пор, пока частное не станет меньше основания. Так, перевод в 16-ричную систему осуществляется следующим образом:

$$\begin{array}{llll} 347 & \underline{|\,16} & & \\ \underline{336} & 21 & \underline{|\,16} & \\ 11 & \underline{16} & 1 & \\ & 5 & & \end{array} \qquad 347_{10}=15B_{16}$$

Перевод из m-ричной системы в десятичную систему производится путем сложения произведений соответствующего десятичного эквивалента символа числа в m-ричной системе на вес i-го знакоместа.

Пример перевода из 16-ричной системы счисления в десятичную систему:

$$15B_{16} = 1 \cdot 16^2 + 5 \cdot 16^1 + 11 \cdot 16^0 = 256 + 80 + 11 = 347_{10}.$$

1.7. Единицы измерения количества информации

Первоначально слово «информация» означало сведения об окружающем мире и протекающих в нем процессах, что предполагает наличие смысла, значимости сообщения. Смысл и значимость — понятия человеческие, субъективные. Информацию перед использованием (обработкой, хранением, передачей) необходимо закодировать. Кодирование производится с помощью специальных алфавитов. В отличие от общепринятых алфавитов (национальных, азбуки Морзе, рельефного шрифта Брайля), используемых человеком, при работе ЭВМ применяется двоичный алфавит.

Кодирование информации, при котором используются два символа 1 и 0, называется *двоичным кодированием*. Минимальный объем информации, который может быть передан с помощью этой кодировки, т.е. цифры 1 или 0, называется *битом* (от английского BInary digiT — двоичная цифра).

Как правило, устройства ЭВМ работают не с отдельными битами, а с группой битов сразу. Последовательность, составленная из восьми битов, составляет один байт.

Для измерения количества информации используются также более крупные единицы:

1 Килобайт = 1024 байта (2^{10} байта)
1 Мегабайт = 1024 Кбайта (2^{20} байта)
1 Гигабайт = 1024 Мбайта (2^{30} байта)
1 Терабайт = 1024 Гбайта (2^{40} байта)
1 Петабайт = 1024 Тбайта (2^{50} байта)
1 Экзабайт = 1024 Пбайта (2^{60} байта).

1.8. Представление числовой информации

1. Целые числа со знаком.

Тип	*Число бит*	*Диапазон значений*
Короткий	16	$-32768 \ldots +32767$
Средний	32	$-2 \cdot 10^9 \ldots +2 \cdot 10^9$
Длинный	64	$-9 \cdot 10^{18} \ldots +9 \cdot 10^{18}$

2. Действительные числа, представленные в формате с плавающей точкой.

Любое вещественное число N может быть представлено в виде $N = \pm A \cdot m^{\pm p}$, где A — мантисса, m — основание системы счисления, p — характеристика (или порядок) числа. Если после запятой в мантиссе стоит не нуль, то число называется нормализованным справа. Нормализованное число одинарной точности (до семи значащих цифр) размещается в памяти в 32 битах следующим образом:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15	14	...	2	1	0

8 бит-смещенный порядок *23 бита-мантисса*

1 бит-знаковый

Поскольку в нормализованной мантиссе первая цифра всегда равна 1, ее переводят в целую часть числа, а запись мантиссы в память начинается со второй цифры. Первая единица автоматически восстанавливается при преобразовании числа или в процессе вычисления.

Порядок числа с плавающей запятой изменяется в диапазоне от –127 до +128. Для того чтобы порядок был всегда положительным, его принимают увеличенным на 127.

$$P_{\text{смещенный}} = P + 2^7 = 1.$$

Такой способ представления порядка называют смещенным. Рассмотрим примеры:

1) Число $-0{,}0625_{10} = -0{,}0001_2 = -1{,}0 * 2^{-4}$ разместится в памяти компьютера следующим образом: 10111101 10000000 00000000 00000000.

Первый бит = 1, это означает, что число отрицательное. $P_{\text{смещ}} = -4 + 127 = 123_{10} = 01111011$, мантисса состоит из нулей.

2) Число $25_{10} = 11001 = 1{,}1001 * 2^{4}$ разместится в памяти компьютера следующим образом: 01000001 11001000 00000000 00000000.

Первый бит = 0, значит число положительное. $P_{\text{смещ}} = -4 + 127 = 131_{10} = 10000011_2$, в мантиссе после отбрасывания целой части остается 1001.

Нормализованное число двойной точности размещается в памяти в 80 битах, причем под мантиссу отводится 55 бит.

Таким образом, количество бит информации в числе определяется количеством знакомест, необходимых для представления этого числа в двоичной системе.

1.9. Представление текстовой информации

При вводе документов, текстов программ и другой информации вводимые символы (буквы, цифры, знаки) кодируются определенными комбинациями из восьми нулей и единиц и наоборот — при выводе их для чтения человеком (на монитор или принтер) по коду символа строится изображение символа.

При двоичном кодировании текстовой информации каждому символу назначается код — последовательность из фиксированного количества нулей и единиц со взаимно однозначным соответствием. Используя 1 двоичную цифру (один бит) можно закодировать всего 2 символа. Двухбитовых комбинаций может быть 4 → 00; 01, 10, 11, т. е. 2^2, с помощью трех битов можно получить восемь различных сочетаний нулей и единиц (2^3). Аналогичным образом можно подсчитать, что число битов, необходимое для кодирования 32 различных символов, равно 5 (2^5). Этот код использовался в работе телеграфа в 20-е годы прошлого столетия, вместо знаков препинания ставились ТЧК и ЗПТ. Используя 7 бит, можно закодировать 128 символов (двоичный семибитовый код обмена информацией КОИ-7), а с помощью 24 бит — 16777216 различных символов или состояний.

Оптимальное количество символов, которые используются при наборе различных текстов, равно примерно 200 (буквы латинские и русские, заглавные и строчные, знаки препинания, цифры, математические знаки, элементы псевдографики). В двоичной системе такое количество символов может быть закодировано последовательностью из 8 бит (2^8=256), т.е. 1 байтом.

Кодировка IBM (ASCII коды American Standard Coding for Information Interchange) состоит из двух частей: нижняя является общепринятой во всем мире (десятичные коды 0-127).

Фрагмент кодировки ASCII:

Код двоичный	*Символ*	*Десятичный код*
0010 0000	пробел	32
0010 1011	+	43
0011 0000	1	49
0011 0000	0	48
0011 1001	9	57
0011 1010	двоеточие :	58
0100 1101	М лат	77

Первые 32 кода зарезервированы для различных управляющих символов, таких как возврат каретки, табуляция, отмена операции и т.п.

Вторая — «верхняя половина» представляет собой расширенные ASCII коды, в ней находятся национальные алфавиты и специальные символы. В России вторая половина подчиняется 4 разным стандартам: КОИ-8 (Код обмена информацией восьмизначный или кодовая страница 866, полученная путем замены греческих букв и некоторых элементов псевдографики из таблиц ASCII кодов на буквы русского алфавита), кодировка WINDOWS 1251, ISO, модифицированная альтернативная кодировка ГОСТ. В последней прописные буквы от А до Я имеют десятичные коды 128 — 159, строчные буквы от а до п имеют десятичные коды 160 — 175, от р до я имеют коды 224 — 241.

Помимо восьмиразрядной системы кодирования символьной (текстовой) информации разработана система шестнадцатиразрядного кодирования символов, которая получила название универсальной, UNICODE. Такая система позволяет закодировать $2^{16} = 65\,536$ различных символов, в том числе практически все алфавиты языков нашей планеты.

Расчет объема текстовой информации сводится к вычислению произведения количества символов в тексте на число разрядов двоичного кода, необходимого для кодирования одного символа.

1.10. Кодирование цветовой и графической информации

Последовательностями нулей и единиц можно закодировать и графическую информацию.

Различают три вида компьютерной графики: растровую, векторную и фрактальную.

Рассмотрим наиболее часто используемую при разработке электронных (мультимедийных) и полиграфических изданий растровую графику. Основным элементом растрового изображения является точка, или пиксель.

Для кодирования любого изображения нужно разбить его на точки и цвет каждой точки закодировать. Например, черно-белую картинку можно закодировать, используя два бита: 11 — белый цвет, 10 — светло-серый, 01 — темно-серый и 00 — черный цвет.

Для кодировки 256 различных цветов требуется 8 бит. Однако этого недостаточно для кодирования полноцветных изображений живой природы. Человеческий глаз может различать десятки миллионов цветовых оттенков. В современных компьютерах для кодирования цвета одной точки используется 3 байта.

Каждый цвет представляет собой комбинацию трех основных цветов: красного, зеленого и синего. Первый байт определяет интенсивность красной составляющей, второй — зеленой, третий — синей.

Белый цвет кодируется полными тремя байтами (255, 255, 255 или в двоичной системе 11111111, 11111111, 11111111). Черный цвет — отсутствие всех цветов — (0,0,0). Красный цвет может быть темным — (120,0,0) или ярко-красным (255,0,0). Такая система кодирования цветной графической информации называется системой RGB (Red, Green, Blue) и обеспечивает однозначное определение 16,5 млн. различных цветов и оттенков (2^{24}). Качество графического изображения зависит от количества точек (пикселей) на единице площади. Этот параметр называется *разрешением* и измеряется в точках на дюйм — *dpi*.

Расчет объема графической информации сводится к вычислению произведения количества точек на изображении на количество разрядов, необходимых для кодирования цвета одной точки.

Например, для цветной картинки, составленной из 256 цветов в графическом режиме монитора 640 x 480, требуется объем видеопамяти, равный:

$8 \cdot 640 \cdot 480 = 2457600$ бит $= 307200$ байт $= 300$ Кбайт.

Задания для самостоятельной работы

1. Преобразовать десятичные числа в восьмеричные и шестнадцатеричные: 35; 1024; 1135.

2. Перевести в восьмеричную и шестнадцатеричную системы счисления следующие двоичные числа:

а) 1111010100010000010011110010100О;
б)100010101010110011001100000000111.

3. Используя двоичное счисление, произвести сложение двух чисел: а) 75 + 44; б) 158 + 36; в) 144 + 56. Проверить результат вычислений путем перевода его в десятичную систему.

4. Используя двоичное счисление, произвести вычитание путем сложения дополнений до двух : а) 75 – 44; б) –15 – 36; в) 14 – 56. Проверить результат вычислений путем перевода его в десятичную систему.

5. Используя двоичное счисление, произвести деление : а) 75 : 5; б) 54 : 6; в) 56 : 14. Проверить результат вычислений путем перевода его в десятичную систему.

6. Рассчитать объем памяти, необходимый для хранения следующих чисел: а) 35_{10}; б) 1024_{10}; в)1135_8; г) $10AF_{16}$.

7. Рассчитать объем памяти, необходимый для хранения следующих чисел: а) 12,123456789; б) 1456123,23 с одинарной и двойной точностью.

8. Подсчитать количество информации, содержащейся в записи полного адреса вашего учебного заведения, при использовании различных кодировок.

9. Вычислить объем памяти, который займет при двоичном кодировании цветная картинка:

а) размером 2×4 см, при использовании 256 цветовых оттенков;

б) размером 5×6 см, при использовании 15 000 цветовых оттенков.

Учесть, что в каждом квадратном сантиметре содержится 24×24 точки.

10. Какой объем адресуемой оперативной памяти имеют ОЗУ с 16-битовой адресной организацией?

Контрольные вопросы

1. Что такое позиционная система счисления?
2. В чем состоит отличие позиционной системы от непозиционной? Приведите примеры.
3. Назовите общее правило перевода чисел из любой системы счисления в десятичную систему.
4. Расскажите правила перевода чисел из десятичной системы счисления в любую другую систему.
5. Какие операции с двоичными числами может выполнять процессор вычислительного устройства?
6. Какие существуют формы представления отрицательного числа в двоичной системе счисления?
7. Как представляются целые и действительные числа в ЭВМ? Приведите примеры.
8. Какой способ представления порядка числа с плавающей запятой называется смещенным?
9. Как представляются символьные данные в памяти ЭВМ?
10. Что такое управляющие символы и как они кодируются?
11. Какие данные хранятся в файлах, содержащих растровые изображения?