ТЕМА 2: Представление данных в ЭВМ

Содержание:

1. **Позиционные системы счисления**
2. Перевод чисел из одной системы счисления в другую
3. Представление отрицательных чисел
4. Двоичная арифметика
5. Форматы чисел

Основные понятия

Цифрами называется набор символов, с помощью которых записываются числа. Количество этих символов определяет **основание** системы счисления. Так, в десятичной системе имеется 10 цифр (0,1,2,3,4,5,6,7,8,9), а в двоичной – 2 цифры (0 и 1).

Система счисления называется **позиционной*,*** если в записи числа каждая цифра имеет свой вес.

Рассмотрим 4-х разрядное десятичное число – 7823. В нем: 7 тысяч; 8 сотен; 2 десятка и 3 единицы.

Величина числа определяется по формуле:

$$D = 7 \cdot 10^3+8 \cdot 10^2+2 \cdot 10^1+3 \cdot 10^0$$

или в общем виде для целого числа :

$$D = d_3 \cdot 10^3+d_2 \cdot 10^2+d_1 \cdot 10^1+d_0 \cdot 10^0$$

Каждая цифра d_i этого десятичного числа имеет вес, равный 10^i . Число 10 является основанием системы счисления.

В общем случае, в системе счисления с основанием b произвольное число, имеющее целую и дробную части записывается с помощью цифр d_i (их количество равно основанию системы) следующим образом:

$$d_{p-1}d_{p-2} \cdot\cdot\cdot d_1d_0.d_{-1}d_{-2} \cdot\cdot\cdot d_{-n},$$

где p – число цифр, расположенных слева , а n – число цифр, расположенных справа от точки, отделяющей целую часть числа от дробной.

Основные понятия ***продолжение***

Значение числа представляет собой сумму произведений отдельных цифр на основание системы счисления в соответствующей степени:

$$D = \sum_{i=-n}^{p-1} d_i \cdot b^i$$

В позиционной системе счисления крайняя левая цифра называется **цифрой старшего разряда**, а крайняя правая – **цифрой младшего разряда**.

В вычислительной технике используется двоичная система счисления. Цифры 0 и 1 называют битами. Каждый бит, стоящий в i-той позиции числа имеет вес 2^i .Например,

$$10001_2=1\cdot16+0\cdot8+0\cdot4+0\cdot2+1\cdot1=17_{10}$$

В двоичной системе используются понятия **бит старшего разряда** (старший бит) и **бит младшего разряда** (младший бит). Группу из трех рядом стоящих двоичных разрядов называют **триадой**, из четырех разрядов – **тетрадой**, а восьми разрядов – **байтом.**

Используемые системы счисления

Для пользователей ЭВМ кроме систем счисления 2 и 10 удобны восьмеричная (основание 8) и шестнадцатеричная (основание 16) системы счисления. Ниже в таблице приведено соответствие между числами указанных систем счисления.

Десятичные	Двоичные	Восьмеричные	Шестнадцатеричные
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
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

Содержание:

1. Позиционные системы счисления
2. **Перевод чисел из одной системы счисления в другую**
3. Представление отрицательных чисел
4. Двоичная арифметика
5. Форматы чисел

Перевод чисел между системами с основанием 2^n

Восьмеричная и шестнадцатеричная системы счисления удобны для представления двоичных чисел, т.к. их основания представляют собой степени числа 2.

Для преобразования двоичного числа в восьмеричное (шестнадцатеричное) нужно двоичное число, начиная от точки влево и вправо разбить на триады (тетрады) и каждую из них записать соответствующей восьмеричной (шестнадцатеричной) цифрой. При этом неполные группы слева и справа для удобства дополняются незначащими нулями.

$$11.1010011011_2 = 011\ .\ 101\ \ 001\ \ 101\ \ 100_2 = 3.5154_8$$

$$11.1010011011_2 = 0011\ .\ 1010\ \ 0110\ \ 1100_2 = 3.A6C_{16}$$

Для обратного преобразования 8→2 или 16→2 необходимо каждую восьмеричную или шестнадцатеричную цифру записать соответствующим эквивалентом из 3 или 4 бит.

Перевод числа из десятичной системы в двоичную

1. ***Для целой части числа***:

Последовательно делим ***целую часть*** на основание новой системы счисления, записывая получившиеся остатки. Цифры остатков объединяем в обратном порядке (первый остаток – младший разряд, последний остаток – старший разряд).

$$300_{10}=100101100_2$$

Частное от деления на 2	Остатки
300	0 (мл.бит)
150	0
75	1
37	1
18	0
9	1
4	0
2	0
1	1 (ст.бит)
0 признак окончания	

Перевод числа из десятичной системы в двоичную

2. Для дробной части числа:

Последовательно умножаем ***дробную часть*** числа на основание новой системы счисления, каждый раз отделяя целые части произведений. Перевод заканчивается при достижении требуемой точности.

Целые части произведений	
(ст.бит) 0.	5408 × 2
1.	0816 × 2
0.	1632 × 2
0.	3264 × 2
0.	6528 × 2
1.	3056 × 2
0.	6112 × 2
1.	2224 × 2
0.	4448 × 2
0.	8896 × 2
1.	7792 × 2
1.	5584 × 2
(мл.бит) 1.	1168

$$\mathbf{0{,}5408_{10} = 0.\ 1000\ \ 1010\ \ 0111_2}$$

Перевод числа из двоичной системы в десятичную

Для перевода двоичного числа в десятичное число нужно воспользоваться формулой:

$$D = \sum_{i=-n}^{p-1} d_i \cdot b^i$$

Преобразование можно выполнить отдельно для целой части числа и дробной части, а затем результаты сложить.

Целая часть:

$\mathbf{100101100_2 = 1\cdot 2^8+0\cdot 2^7+0\cdot 2^6+1\cdot 2^5+0\cdot 2^4+1\cdot 2^3+1\cdot 2^2+0\cdot 2^1+0\cdot 2^0 = 256+32+8+4 = 300_{10}}$

Дробная часть:

$\mathbf{0.100010100111_2=}$

$\mathbf{1\cdot 2^{-1}+0\cdot 2^{-2}+0\cdot 2^{-3}+0\cdot 2^{-4}+1\cdot 2^{-5}+0\cdot 2^{-6}+1\cdot 2^{-7}+0\cdot 2^{-8}+0\cdot 2^{-9}+1\cdot 2^{-10}+1\cdot 2^{-11}+1\cdot 2^{-12}=}$

=0.5+0+0+0+0.031+0+0.0078125+0+0+0.0009765625+0.00048828125+0.000244140625=

=0.54077148

Результат:

$$\mathbf{100101100.100010100111_2 = 300.54077148_{10}}$$

Двоично-десятичные числа

Особое место занимает двоично-десятичная система представления чисел.

В ней десятичные цифры от 0 до 9 представляются 4-х разрядными двоичными комбинациями от 0000 до 1001; комбинации от 1010 до 1111 не используются.

Эта система нашла широкое применение во многих устройствах ввода-вывода десятичных данных. Многие ЭВМ имеют полный набор команд для операций над двоично-десятичными числами.

Содержание:

Отрицательные числа в прямом коде

Наиболее широко применяемым способом представления десятичных чисел является прямой код. Числа при этом состоят из величины и знака. +98, -100. Нуль можно представить двумя способами +0 и –0 (оба значения совпадают).

В двоичной системе для представления знака вводится дополнительный знаковый разряд: "0" это "+"; "1" это "–".

$00101011_2=+43_{10}$ $\quad\quad$ $10101011_2=–43_{10}$

Система представления в прямом коде содержит одинаковое количество положительных и отрицательных чисел, причем ноль может быть представлен двумя способами.

Правила сложения двух чисел в прямом коде:

♦ если знаки чисел совпадают, то их величины складываются, а результату присваивается знак слагаемых;

♦ если знаки разные, то из большей величины вычитается меньшая, а результату присваивается знак большей величины.

Таким образом, необходимо производить проверку знаков слагаемых.

Отрицательные числа в дополнительном коде

В ЭВМ отрицательные числа обычно представляются в виде ***дополнений***.

Чаще всего используется представление чисел дополнением до основания системы счисления. При двоичной системе это ***дополнение до двух***.

По определению с***умма n-разрядного двоичного числа и его дополнения равна 2ⁿ.*** Поэтому для нахождения дополнения числа D нужно найти разность 2^n-D.

Пример.
Пусть числа задаются семью значащими разрядами (0 – 7) и восьмым знаковым (+/-):

Номера битов	7	6	5	4	3	2	1	0
	0	1	1	0	0	1	0	1
Знак и веса битов	+/-	64	32	16	8	4	2	1

В данном случае имеем число 01100101_2=$+65_{16}$. Образуем его дополнение. Для этого вычислим разность 2^8 –D:

$$\begin{array}{r} 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1 \\ \hline 1\ 0\ 0\ 1\ 1\ 0\ 1\ 1 \end{array}$$

Непосредственно вычитать сложно, т.к. приходится делать заем из старшего разряда.

Отрицательные числа в дополнительном коде

Для упрощения вычислений преобразуем исходное выражение:

$$\mathbf{2^n - D = [(2^n - 1) - D] + 1.}$$

Если число 2^n записывается в виде единицы с n нулями (100000000), то число 2^n-1 будет записано с помощью n единиц (11111111).

Из такого числа вычитать легче, т.к. ни в одном из разрядов, независимо от величины вычитаемого, заем делать не надо. Разность (2^n-1)-D представляет собой ни что иное, как ***инверсию*** числа D.

ПРАВИЛО*: для получения дополнения до двух некоторого числа D необходимо образовать его* ***инверсию****, а затем к инверсии прибавить* ***единицу****.*

Достоинством представления отрицательных чисел дополнениями является то, что ***при сложении чисел*** не нужно проверять их знак (как в случае использования прямого кода). *Всегда выполняется двоичное сложение* и, если результат не выходит за границы разрядной сетки рассматриваемой системы*, то результат всегда получается верным.* ***Вычитание двоичных чисел заменяется сложением уменьшаемого с дополнением вычитаемого*****.**

Отрицательные числа в дополнительном коде

Особенностью представления чисел дополнением до двух является то, что нуль может быть представлен единственным образом (отсутствует «–0»).

Количество отрицательных чисел на единицу больше, чем положительных. В таблице приведены десятичные числа и их представление в дополнительном коде.

+127	01111111
…..	…..
+5	00000101
+4	00000100
+3	00000011
+2	00000010
+1	00000001
+0	00000000
–1	11111111
–2	11111110
–3	11111101
–4	11111100
–5	11111011
……	……
–128	10000000

Содержание:

1. Позиционные системы счисления
2. Перевод чисел из одной системы счисления в другую
3. Представление отрицательных чисел
4. **Двоичная арифметика**
5. Форматы чисел

Арифметические операции

Сложение

```
   0011
+
   0100
  ------
   0111
```

Вычитание

```
   0100
+
   1001
  ------
   1101
```

Умножение

```
       1011
   ×
       1101
     ------
       1011
      0000
     1011
    1011
   ---------
   10001111
```

Деление

```
10001111 | 1101
1101     |------
-----    | 1011
010011
  1101
 -----
  1101
  1101
  ----
     0
```

Логические операции

Логические операции выполняются над отдельными битами двоичного числа без формирования переносов в старший разряд в соответствии со следующими таблицами истинности:

Входы		**Выход**					
A	**B**	**И**	**И-НЕ**	**ИЛИ**	**ИЛИ-НЕ**	**XOR**	**XOR-НЕ**
0	**0**	**0**	**1**	**0**	**1**	**1**	**0**
0	**1**	**0**	**1**	**1**	**0**	**0**	**1**
1	**0**	**0**	**1**	**1**	**0**	**0**	**1**
1	**1**	**1**	**0**	**1**	**0**	**1**	**0**

Логическая операция "И" используется для выделения отдельных битов в байте. Например, если нужно определить состояние битов 0 и 4 следует в качестве маски использовать комбинацию: 00010001
При выполнении операции И, в соответствии с таблицей, в анализируемом байте сохранятся биты 0 и 4 , а все остальные будут обнулены.

Логическая операция "ИЛИ" (1) используется для принудительной установки необходимых битов в единичное состояние.

Арифметические сдвиги

Арифметическим сдвигом влево (вправо) двоичного числа называется перемещение всех его разрядов на одну позицию влево (вправо).

При сдвиге влево старший разряд числа теряется, а в младший разряд записывается 0.

При сдвиге вправо младший разряд числа теряется, а в старший разряд записывается 0.

Пример сдвига влево на 1 разряд:

0	0	1	0	1	1	0	1		$2D_{16}$	
□ Сдвиг влево										
0	1	0	1	1	0	1	0		$5A_{16}$	

При сдвиге числа влево на один разряд число увеличивается в 2 раза.

При сдвиге числа вправо на один разряд число уменьшается в 2 раза.

Содержание:

Числа с фиксированной точкой

Системы представления, в которых положение двоичной точки фиксировано, называются системами представления с фиксированной точкой.

Диапазон чисел при таком представлении определяется числом разрядов, отведенных для целой части.

+/-	Целая часть								Дробная часть						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Место подразумеваемой точки ^															

Числа с плавающей точкой

Значительно больший диапазон чисел при фиксированном числе разрядов обеспечивается в случае представления с плавающей точкой. Слово разбивается на три поля: мантиссу, порядок и знак мантиссы.

31	30	…	23	22	…	0
Знак мантиссы	Порядок (P) или характеристика (X)			Мантисса (M)		

Величина числа в таком формате определяется как $M \cdot 2^P$. Мантисса записывается в нормализованном виде – она меньше единицы и первая цифра отлична от нуля.

Чтобы избежать действий с отрицательными порядками, все порядки увеличены на 64, т.е. вместо порядка введена характеристика:

$$X=P+64$$

Характеристика изменяется в диапазоне от 0 до 127, что соответствует изменению порядка от –64 до +63.

Кодирование символов

ЭВМ может оперировать не только с числами, но и с текстом. Текст представляет собой строки символов определенного набора. Каждый символ представляется комбинацией битов в соответствии с таблицей кодировки, называемой таблицей ASCII – кодов.

В этом коде каждый символ представляется 8-битовой последовательностью двоичных цифр (содержит 256 различных символов).

Важной особенностью кода ASCII является то, что комбинации двоичных цифр, соответствующие буквам и цифрам, образуют числовые последовательности, несущие смысловую нагрузку, так что сортировку строк текста можно производить с помощью машинных команд, предназначенных для сравнения числовых значений.

Кроме ASCII кода широко применяется UNICOD, в котором код символа записывается 16-разрядным двоичным числом. Такой набор содержит до $2^{16} = 65536$ символов.

Таблица ASCII кодов

Кодовая страница **RUSSIAN 866**

Код	0	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
0	NUL	DLE	SP	0	@	P	`	p	А	Р	а	░	└	╨	р	Ё
1	SOH	DC1	!	1	A	Q	a	q	Б	С	б	▒	┴	╤	с	ё
2	STX	DC2	“	2	B	R	b	r	В	Т	в	▓	┬	╥	т	Є
3	ETX	DC3	#	3	C	S	c	s	Г	У	г	│	├	╙	у	є
4	EOT	DC4	$	4	D	T	d	t	Д	Ф	д	┤	─	╘	ф	Ї
5	ENQ	NAK	%	5	E	U	e	u	Е	Х	е	╡	┼	╒	х	ї
6	ACK	SYN	&	6	F	V	f	v	Ж	Ц	ж	╢	╞	╓	ц	Ў
7	BEL	ETB	´	7	G	W	g	w	З	Ч	з	╖	╟	╫	ч	ў
8	BS	CAN	(	8	H	X	h	x	И	Ш	и	╕	╚	╪	ш	°
9	HT	EM	)	9	I	Y	i	y	Й	Щ	й	╣	╔	┘	щ	•
A	LF	SUB	*	:	J	Z	j	z	К	Ъ	к	║	╩	┌	ъ	·
B	VT	ESC	+	;	K	[	k	{	Л	Ы	л	╗	╦	█	ы	√
C	FF	FS	,	<	L	\	l	|	М	Ь	м	╝	╠	▄	ь	№
D	CR	GS	-	=	M	]	m	}	Н	Э	н	╜	═	▌	э	¤
E	SO	RS	.	>	N	^	n	~	О	Ю	о	╛	╬	▐	ю	■
F	SI	US	/	?	O	_	o	DEL	П	Я	п	┐	╧	▀	я	