Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Д. В. Куприянова, И. В. Лукьянова, Ю. А. Луцик

АРИФМЕТИЧЕСКИЕ И ЛОГИЧЕСКИЕ ОСНОВЫ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

*Рекомендовано УМО по образованию в области информатики
и радиоэлектроники в качестве пособия для специальности
1-40 02 01 «Вычислительные машины, системы и сети»*

Минск БГУИР 2020

УДК 004.31(075)
ББК 32.973я7
К92

Р е ц е н з е н т ы:
кафедра робототехнических систем
Белорусского национального технического университета
(протокол №5 от 27.12.2019);

заведующий лабораторией идентификации систем
государственного научного учреждения
«Объединенный институт проблем информатики
Национальной академии наук Беларуси»
доктор технических наук, профессор А. А. Дудкин

Куприянова, Д. В.
К92 Арифметические и логические основы вычислительной техники : пособие /Д. В. Куприянова, И. В. Лукьянова, Ю. А. Луцик. – Минск : БГУИР, 2020. − 70 с. : ил.
ISBN 978-985-543-573-1.

Приведены контрольные вопросы для проверки знаний по теме занятия, примеры выполненных заданий и задания для выполнения на практических занятиях.

УДК 004.31(075)
ББК 32.973я7

ISBN 978-985-543-573-1

Введение

В пособии к каждой изучаемой теме приводится список контрольных вопросов, владение которыми необходимо для успешного выполнения практических заданий по теме занятия. Перед выполнением заданий по теме студент может ознакомиться с выполненными примерами, аналогичными тем заданиям, которые он будет выполнять. В решении примеров приводятся пояснения и ссылки на литературу, где рассматривается теоретический материал по теме.

Данное пособие может быть использовано студентами как на занятиях, так и при подготовке к ним.

Тема 1. Системы счисления и операции с числами

Контрольные вопросы

1. Что такое система счисления (с/с)?
2. Какая система счисления называется позиционной?
3. Какие категории систем счисления можно выделить?
4. Какие существуют способы перевода чисел из одной системы счисления в другую?
5. Перечислите существующие критерии выбора системы счисления.

Арифметические операции над числами в различных системах счисления

Прежде чем приступать к выполнению практических заданий рекомендуется ознакомиться с материалом в [4, с. 10–22]. Следует помнить, что правила арифметики для всех позиционных систем счисления одинаковы.

Рассмотрим несколько примеров на сложение и вычитание чисел в различных системах счисления.

Пример 1. Выполнить сложение и вычитание чисел в двоичной системе счисления: A_2 = 10010111 и B_2 = 01001101.

$$\begin{array}{r} A_2 = 10010111 \\ \underline{B_2 = 01001101} \\ A_2 - B_2 = 01001010 \end{array} \qquad \begin{array}{r} A_2 = 10010111 \\ \underline{B_2 = 01001101} \\ A_2 + B_2 = 11100100 \end{array}$$

Пример 2. Выполнить сложение и вычитание чисел в восьмеричной системе счисления: A_8 = 5726 и B_8 = 1267.

$$\begin{array}{r} A_8 = 5726 \\ \underline{B_8 = 1267} \\ A_8 - B_8 = 4437 \end{array} \qquad \begin{array}{r} A_8 = 5726 \\ \underline{B_8 = 1267} \\ A_8 + B_8 = 7215 \end{array}$$

Пример 3. Выполнить сложение и вычитание чисел в шестнадцатеричной системе счисления: $A_{16} = A7F6$ и $B_{16} = 4D69$.

$$\begin{array}{r} A_{16} = A7F6 \\ \underline{B_{16} = 4D69} \\ A_{16} - B_{16} = 5A8D \end{array} \qquad \begin{array}{r} A_{16} = A7F6 \\ \underline{B_{16} = 4D69} \\ A_{16} + B_{16} = F55F \end{array}$$

Перевод чисел из одной системы счисления в другую

Необходимо отметить, что для перевода целых и дробных чисел используются разные способы. В примерах используем универсальные методы – метод деления на основание системы счисления, в которую выполняется перевод, для целой части числа и метод умножения на основание системы счисления, в которую выполняется перевод, для дробной части числа. Необходимо отметить, что все действия будут производиться в той системе счисления, из которой осуществляется перевод.

Пример 4. Выполнить перевод числа A_{10} = 713,793 из десятичной в двоичную систему счисления.

```
713  |2                                                   0,793
6     356 |2                                                  2
11    2    178 |2                                         1,586
10    15   16   89 |2                                         2
 13   14    18  8   44 |2                                 1,172
 12    16   18  9   44  22 |2                                 2
  1    16    0  8    0  22  11 |2                         0,344
        0       1        0  10  5 |2                          2
                            1   4  2 |2                   0,688
                                1  2  1                       2
                                   0                      1,376
                                                              2
                                                          0,752
                                                              2
                                                          1,504
                                                              2
                                                          1,008
                                                              2
                                                          0,016
```

Старшая цифра нового целого числа определяется последним результатом деления, а дальше записываются остатки от деления, начиная с последнего. В дробной части числа старшей цифрой новой дроби (после запятой) будет целая часть от первого умножения, далее производится умножение только дробной части числа, а цифры, которые получаются в результате последующих умножений в целой части результата, будут определять последующие цифры дроби (слева направо).

В результате получено двоичное число A_2 = 1011001001,110010110.

Пример 5. Выполнить перевод числа A_8 = 265 из восьмеричной системы счисления в десятичную, используя разложение в ряд.

$$265_8 = 2{\cdot}8^2+6{\cdot}8^1+5{\cdot}8^0 = 181_{10}.$$

Пример 6. Выполнить перевод числа A_2 = 10110101 из двоичной системы счисления в четверичную. Перевод будет выполняться по основанию кратному степени двойки. Исходное число разобьем на диады, после чего запишем четверичный эквивалент каждой диады. В результате получим четверичное число:

$\underline{10}\ \underline{11}\ \underline{01}\ \underline{01}_2 = 2311_4$.

Пример 7. Выполнить перевод числа A_4 = 2311 из четверичной системы счисления в восьмеричную, используя метод деления на основание системы счисления. Число делим на основание новой системы счисления в старой системе счисления ($8 = 20_4$).

```
2311  |20
20     112 |20
 31    100  2
 20     12
 111              2
 100          6
  11
       5
```

В результате деления мы получили три остатка от деления, теперь запишем результат перевода числа из одной системы счисления в другую: A_8 = 265.

Пример 8. Выполнить перевод числа A_{10} = 181 в шестнадцатеричную систему счисления. Для перевода числа использовать метод деления на основание системы счисления.

```
181  |16
16    11
 21
 16        B
  5
     5
```

В результате деления получилось два остатка от деления, результат перевода числа: $A_{16} = B5$.

Практические задания

1. Вычислить сумму и разность чисел в указанной системе счисления:

а) A_2 = 1001,0010; B_2 = 0101,1011;
б) A_2 = 1010,10001; B_2 = 0101,01010;
в) A_2 = 10101,0101; B_2 = 01101,1001;
г) A_4 = 2232,13; B_4 = 1312,12;
д) A_4 = 2312,231; B_4 = 1232,313;
е) A_8 = 5421,72; B_8 = 1463,16;
ж) A_8 = 4576,17; B_8 = 2541,75;
з) $A_{16} = 94A,5FE$; $B_{16} = 5C7,B45$;
и) $A_{16} = A23F,E8$; $B_{16} = 3CA9,5B$;
к) A_8 = 4712,26; B_8 = 1561,72.

2. Перевести числа, заданные в десятичной системе счисления, в двоичную, восьмеричную и шестнадцатеричную системы счисления методом деле-

ния и умножения на основание системы счисления, в которую выполняется перевод. Проверить правильность выполненного перевода чисел, для этого перевести полученный результат обратно в десятичную систему счисления.

а) A_{10} = 413,718;	е) A_{10} = 802,299;
б) A_{10} = 595,397;	ж) A_{10} = 658,315;
в) A_{10} = 528,805;	з) A_{10} = 413,718;
г) A_{10} = 315,421;	и) A_{10} = 813,922;
д) A_{10} = 239,945;	к) A_{10} = 592,712.

Тема 2. Коды чисел

Контрольные вопросы

1. Какие виды кодов чисел существуют?
2. Как заменить операцию вычитания операцией сложения?
3. Что такое дополнение числа?
4. Как записать число из естественной формы записи в дополнительном коде?
5. Как записать число из естественной формы записи в обратном коде?

Сложение чисел с фиксированной запятой в дополнительном и обратном коде

Прежде чем приступать к выполнению практического задания, рекомендуется ознакомиться с материалом в [4, с. 22–25].

Пример 1. Представить заданные числа в прямом, обратном и дополнительном коде:

$A_2 = 10110110$	$A_{10} = 3759$	$A_8 = 5723$	$A_{16} = A75C$
$B_2 = -10110110$	$B_{10} = -3759$	$B_8 = -5723$	$A_{16} = -A75C$

Прямой код	Обратный код	Дополнительный код
$[A_2]_{пр} = 0.10110110$	$[A_2]_{обр} = 0.10110110$	$[A_2]_{доп} = 0.10110110$
$[B_2]_{пр} = 1.10110110$	$[B_2]_{обр} = 1.01001001$	$[B_2]_{доп} = 1.01001010$
$[A_{10}]_{пр} = 0.3759$	$[A_{10}]_{обр} = 0.3759$	$[A_{10}]_{доп} = 0.3759$
$[B_{10}]_{пр} = 9.3759$	$[B_{10}]_{обр} = 9.6240$	$[B_{10}]_{доп} = 9.6241$
$[A_8]_{пр} = 0.5723$	$[A_8]_{обр} = 0.5723$	$[A_8]_{доп} = 0.5723$
$[B_8]_{пр} = 7.5723$	$[B_8]_{обр} = 7.2054$	$[B_8]_{доп} = 7.2055$
$[A_{16}]_{пр} = 0.A75C$	$[A_{16}]_{пр} = 0.A75C$	$[A_{16}]_{пр} = 0.A75C$
$[B_{16}]_{пр} = F.A75C$	$[B_{16}]_{пр} = F.58A3$	$[B_{16}]_{пр} = F.58A4$

Пример 2. Выполнить сложение чисел $A_2 = 1001101$ и $B_2 = 0100011$ в дополнительном коде. Здесь и далее (в пределах темы 2) сложение выполнять при всех случаях сочетания знаков слагаемых:

а) $A > 0, B > 0$; б) $A > 0, B < 0$; в) $A < 0, B > 0$; г) $A < 0, B < 0$.

а)
$$\begin{array}{r} [A]_{доп} = 0.1001101 \\ \underline{[B]_{доп} = 0.0100011} \\ [A+B]_{доп} = 0.1110000 \end{array}$$

б)
$$\begin{array}{r} [A]_{доп} = 0.1001101 \\ \underline{[B]_{доп} = 1.1011101} \\ [A+B]_{доп} = 10.0101010 \end{array}$$

отбрасывается

в)
$$\begin{array}{r} [A]_{доп} = 1.0110011 \\ \underline{[B]_{доп} = 0.0100011} \\ [A+B]_{доп} = 1.1010110 \end{array}$$

г)
$$\begin{array}{r} [A]_{доп} = 1.0110011 \\ \underline{[B]_{доп} = 1.1011101} \\ [A+B]_{доп} = 11.0010000 \end{array}$$

отбрасывается

В примерах «б» и «г» выделенный серым цветом разряд (единица) выходит за пределы разрядной сетки и отбрасывается.

Пример 3. Выполнить сложение чисел $A_2 = 0{,}010010$ и $B_2 = 0{,}101101$ в обратном коде.

а)
$$\begin{array}{rr} [A]_{\text{обр}} = & 0{,}010010 \\ [B]_{\text{обр}} = & 0{,}101101 \\ \hline [A+B]_{\text{обр}} = & 0{,}111111 \end{array}$$

б)
$$\begin{array}{rr} [A]_{\text{обр}} = & 0{,}010010 \\ [B]_{\text{обр}} = & 1{,}010010 \\ \hline [A+B]_{\text{обр}} = & 1{,}100100 \end{array}$$

в)
$$\begin{array}{rr} [A]_{\text{обр}} = & 1{,}101101 \\ [B]_{\text{обр}} = & 0{,}101101 \\ \hline [A+B]_{\text{обр}} = & 10{,}011010 \\ & \longrightarrow 1 \\ \hline & 0{,}011011 \end{array}$$

г)
$$\begin{array}{rr} [A]_{\text{обр}} = & 1{,}101101 \\ [B]_{\text{обр}} = & 1{,}010010 \\ \hline [A+B]_{\text{обр}} = & 10{,}111111 \\ & \longrightarrow 1 \\ \hline & 1{,}000000 \end{array}$$

В примерах «б» и «г» разряд (единица) выходит за пределы разрядной сетки и в случае обратного кода выполняется циклический перенос с добавлением единицы в младший разряд.

Пример 4. Выполнить сложение чисел $A_2 = 101101$ и $B_2 = 110001$ в обратном модифицированном коде.

а)
$$\begin{array}{rr} [A]^{\text{м}}_{\text{обр}} = & 00.101101 \\ [B]^{\text{м}}_{\text{обр}} = & 00.110001 \\ \hline [A+B]^{\text{м}}_{\text{обр}} = & 01.011110 \end{array}$$
(переполнение)

б)
$$\begin{array}{rr} [A]^{\text{м}}_{\text{обр}} = & 00.101101 \\ [B]^{\text{м}}_{\text{обр}} = & 11.001110 \\ \hline [A+B]^{\text{м}}_{\text{обр}} = & 11.111011 \end{array}$$

в)
$$\begin{array}{rr} [A]^{\text{м}}_{\text{обр}} = & 11.010010 \\ [B]^{\text{м}}_{\text{обр}} = & 00.110001 \\ \hline [A+B]^{\text{м}}_{\text{обр}} = & 100.000011 \\ & \longrightarrow 1 \\ \hline & 00.000100 \end{array}$$

г)
$$\begin{array}{rr} [A]^{\text{м}}_{\text{обр}} = & 11.010010 \\ [B]^{\text{м}}_{\text{обр}} = & 11.001110 \\ \hline [A+B]^{\text{м}}_{\text{обр}} = & 110.100000 \end{array}$$
(переполнение)

В примерах «а» и «г» было зафиксировано переполнение разрядной сетки, признаком которого является различие значений знаковых разрядов (01 или 10).

Пример 5. Выполнить сложение чисел $A_2 = 101101$ и $B_2 = 111001$ в дополнительном модифицированном коде.

а)
$$\begin{array}{rr} [A]^{\text{м}}_{\text{доп}} = & 00.101101 \\ [B]^{\text{м}}_{\text{доп}} = & 00.111001 \\ \hline [A+B]^{\text{м}}_{\text{доп}} = & 01.101000 \end{array}$$
(переполнение)

б)
$$\begin{array}{rr} [A]^{\text{м}}_{\text{доп}} = & 00.101101 \\ [B]^{\text{м}}_{\text{доп}} = & 11.000111 \\ \hline [A+B]^{\text{м}}_{\text{доп}} = & 11.110100 \end{array}$$

в)
$$\begin{array}{rr} [A]^{\text{м}}_{\text{доп}} = & 11.010011 \\ [B]^{\text{м}}_{\text{доп}} = & 00.111001 \\ \hline [A+B]^{\text{м}}_{\text{доп}} = & 100.001100 \end{array}$$
отбрасывается

г)
$$\begin{array}{rr} [A]^{\text{м}}_{\text{доп}} = & 11.010011 \\ [B]^{\text{м}}_{\text{доп}} = & 11.000111 \\ \hline [A+B]^{\text{м}}_{\text{доп}} = & 110.011010 \end{array}$$
(переполнение)

В примерах «а» и «г» знаковые разряды различаются (01 или 10), это свидетельствует о наличии переполнения разрядной сетки.

Практическое задание

Вычислить сумму чисел в обратном и дополнительном модифицированном коде при всех случаях сочетания знаков слагаемых:

а) A = 11110011, B = 11001111;
б) A = 10110001, B = 10111101;
в) A = 11011111, B = 01110111;
г) A = 11101101, B = 00110010;
д) A = 10101111, B = 00011011;
е) A = 00001001, B = 11001111;
ж) A = 00000111, B = 10000111;
з) A = 01111101, B = 11111101;
и) A = 00000001, B = 01101100;
к) A = 00011110, B = 01111100.

Тема 3. Формы представления чисел. Числа с фиксированной и плавающей запятой. Сложение чисел с плавающей запятой

Контрольные вопросы

1. Приведите общую форму записи числа с фиксированной запятой.
2. Приведите общую форму записи числа с плавающей запятой.
3. Какое число называется нормализованным?
4. Какие варианты денормализации возможны при выполнении арифметических операций над числами с плавающей запятой?
5. Приведите пример классической разрядной сетки для чисел с плавающей запятой (мантисса, порядок).
6. Приведите пример разрядной сетки используемой в ЭВМ для чисел с плавающей запятой (мантисса, характеристика).
7. Приведите последовательность действий при сложении чисел с плавающей запятой.

Сложение чисел с плавающей запятой

Пример 1. $M_A = +0{,}01010$, $p_A = +101$;
$M_B = +0{,}10101$, $p_B = +010$.

$[M_A]_{доп} = 0{,}01010$.
$[M_B]_{доп} = 0{,}10101$.
Разность порядков $p = [p_A]_{доп} + [-p_B]_{доп} =$

$$\begin{array}{r} 0.101 \\ {}^{+}\ \underline{1.110} \\ 1\ 0.011 \end{array} > 0$$

Так как разность порядков $[p_A - p_B]_{доп} > 0$, то сдвигу подвергается мантисса M_B. При каждом сдвиге мантиссы на один разряд из положительной разности порядков производим последовательное вычитание единицы до тех пор, пока в результате не будет получен ноль (т. е. порядки чисел A и B не будут выравнены):

$[M_B]_{доп} = 0{,}10101$ — 0.011; $[-1]_{доп} = \underline{1.111}$

$[M_B]_{доп} = 0{,}\mathbf{0}10101$ — $0.010 \neq 0$

$[M_B]_{доп} = 0{,}\mathbf{00}10101$ — $[-1]_{доп} = \underline{1.111}$; $0.001 \neq 0$

$[M_B]_{доп} = 0{,}\mathbf{000}10101$ — $[-1]_{доп} = \underline{1.111}$; $0.000 = 0$

Порядки чисел A и B выравнены (мантисса M_B числа B сдвинута). Теперь можно выполнять сложение мантисс M_A и M_B.

$[M_A]_{доп} = 0{,}01010$

$[M_B]_{доп} = \underline{0{,}00010}\ 101$

$[M_{A+B}]_{доп} = 0{,}01100\ 101$ − произошла денормализация результата.

При этом порядок суммы равен $p_{A+B} = \max(p_A, p_B) = p_A = +101$, $[p_{A+B}]_{доп} = 0.101$.

Так как полученный результат денормализован (вправо), то необходимо выполнить операцию его нормализации (сдвиг $[M_{A+B}]_{доп}$ влево). При каждом сдвиге мантиссы порядок результата $[p_{A+B}]_{доп}$ уменьшается на единицу (т. к. мантисса увеличивается в два раза):

$[M_{A+B}]_{доп} = 0{,}01100\ 101$ $\qquad$ $[p_{A+B}]_{доп} = 0.101$

$0{,}11001\ 010$ $\qquad$ $\underline{1.111}$ $[-1]_{доп}$

0.100

После выполнения операции округления мантисса результата будет иметь вид $[M_{A+B}]_{доп} = 0{,}11001$.

Далее представим полученный результат в классической разрядной сетке:

$\underline{0}\ \underline{11001}\ \ \underline{0}\ \underline{100}$

Характеристика (смещенный порядок) $p' = 7 + 3 = 10$. Результат сложения в разрядной сетке ЭВМ имеет следующий вид (после нормализации мантиссы):

$[M_{A+B}]_{доп} = 0{,}11001\ 010$ (ненормализованная) $\qquad$ $[p_{A+B}]_{доп} = 0.100$

$1{,}10010\ 100$ (нормализованная) $\qquad$ $\underline{1.111}$ $[-1]_{доп}$

0.011 (3)

Полученный результат представим в разрядной сетке ЭВМ:

$\underline{0}\ \underline{1010}\ \ \underline{10010}$

Пример 2. $M_A = -0{,}11110$, $p_A = -0001$;

$M_B = -0{,}11111$, $p_B = +0010$.

Данный пример выполним в модифицированном дополнительном коде.

$[M_A]^{мод}_{доп} = 1{,}00010.$

$[M_B]^{мод}_{доп} = 1{,}00001.$

Разность порядков $p = [p_A]_{доп} + [-p_B]_{доп} = 1.1111$

$^{+}\underline{1.1110}$

$1\ 1.1101 < 0$

Так как разность порядков $[p_A - p_B]_{доп} < 0$, то сдвигу подвергается мантисса M_A. В этом случае при каждом сдвиге мантиссы числа A на один разряд к отрицательной разности порядков производим последовательное добавление единицы до тех пор, пока в результате не будет получен нуль (т. е. порядки чисел A и B не будут выравнены).

$\left[M_A\right]_{\text{доп}}^{\text{мод}} = 11{,}00010$ $\qquad$ 1.1101

$[+1]_{\text{доп}} = \underline{0.0001}$

$\left[M_A\right]_{\text{доп}}^{\text{мод}} = 11{,}\mathbf{1}0001\ 0$ $\qquad$ $1.1110 \neq 0$

$\left[M_A\right]_{\text{доп}}^{\text{мод}} = 11{,}\mathbf{11}000\ 10$ $\qquad$ $[+1]_{\text{доп}} = \underline{0.0001}$

$1.1111 \neq 0$

$\left[M_A\right]_{\text{доп}}^{\text{мод}} = 11{,}\mathbf{111}00\ 010$ $\qquad$ $[+1]_{\text{доп}} = \underline{0.0001}$

$0.0000 = 0$

Порядки чисел A и B выравнены, далее выполняем сложение мантисс M_A и M_B:

$\left[M_A\right]_{\text{доп}}^{\text{мод}} = 11{,}\mathbf{111}00\ 010$

$\left[M_B\right]_{\text{доп}}^{\text{мод}} = \underline{11{,}00001}$

$\left[M_{A+B}\right]_{\text{доп}}^{\text{мод}} = 110{,}11101\ 010$ − произошла денормализация (переполнение разрядной сетки) результата.

Как и в предыдущем примере, порядок суммы равен $p_{A+B} = \max(p_A, p_B) = p_B = +0010$, $[p_{A+B}]_{\text{доп}} = 0.0010$.

Как и ранее, выполним операцию нормализации $\left[M_{A+B}\right]_{\text{доп}}^{\text{мод}}$. В примере при каждом сдвиге (вправо) мантиссы результата порядок результата $[p_{A+B}]_{\text{доп}}$ увеличивается на единицу (т. к. мантисса результата уменьшается в два раза):

$\left[M_{A+B}\right]_{\text{доп}}^{\text{мод}} = 10{,}11101\ 010$ $\qquad$ $[p_{A+B}]_{\text{доп}} = 0.0010$

$11{,}01110\ 1010$ $\qquad$ $\underline{0.0001}$ $\quad [+1]_{\text{доп}}$

0.0011

После выполнения операции округления (например, по дополнению) мантисса результата будет иметь вид:

$\left[M_{A+B}\right]_{\text{доп}}^{\text{мод}} = 11{,}01111.$

Далее представим полученный результат в классической разрядной сетке:

$\underline{1}\ \underline{01111}\quad \underline{0}\ \underline{0011}$

Характеристика (смещенный порядок) $p' = 7 + 2 = 9$. Результат сложения в разрядной сетке ЭВМ имеет следующий вид (после нормализации мантиссы):

$$\left[M_{A+B}\right]^{\text{мод}}_{\text{доп}} = 11,01110\ \ 1010 \qquad\qquad [p_{A+B}]_{\text{доп}} = 0.0011$$

$$10,11101\ \ 0100 \qquad\qquad \underline{1.1111}\quad [-1]_{\text{доп}}$$

$$0.0010\quad (\ 2\ )$$

Полученный результат представим в разрядной сетке ЭВМ:

1 10001 00011

Пример 3. Выполнить преобразование числа с плавающей запятой *C*2290000 из формы представления его в ЭВМ в естественную форму записи.

Запишем число *C*2290000 (в памяти число занимает 4 байта, это тип числа с плавающей запятой одинарной точности) в двоичном виде:

```
 C   2   2   9   0   0   0   0
1 10000100 01010010000000000000000
1   2               3
```

Здесь 1 – знак числа, 2 – характеристика (132), 3 – мантисса (нормализованная).

Порядок числа $p = 132 – 127 = 5$.

Мантисса $\underbrace{1,01010}_{1}\underbrace{010000000000000000}_{2}$

Здесь 1 – целая, 2 – дробная части.

Выполним перевод числа из двоичной системы счисления в десятичную.

```
101010 |1010          0,010000000000000000
1010    100                           1010
   010  ↓             10,100000000000000000
    ↓   4                             1010
    2                101,000000000000000000

              0,25
```

Таким образом, мы получили, что число *C*2290000 в естественной (десятичной) форме записи будет равно −42,25.

Пример 4. Выполнить преобразование числа с плавающей запятой 429*A*8000 из формы представления его в ЭВМ в естественную форму записи.

Как и в предыдущем примере запишем число 429*A*8000 в двоичном виде:

```
 4   2   9   A   8   0   0   0
0 10000101 00110101000000000000000
```

Порядок числа p = 133 – 127 = 6.

Мантисса $\underbrace{1{,}00110}_{1}\underbrace{10100000000000000000}_{2}$

Здесь 1 – целая, 2 – дробная части.

Выполним перевод числа из двоичной системы счисления в десятичную.

```
1001101 |1010                    0,01000000000000000
 1010    111                                    1010
 10010    ↓                     10,10000000000000000
 01010    7                                     1010
  10001                        101,00000000000000000
   1010
   0111                  0,25
    ↓
    7
```

Таким образом, мы получили, что число *C*2290000 в естественной (десятичной) форме записи будет равно +77,25.

Практические задания

1. Выполнить сложение чисел с плавающей запятой и записать результат в разрядную сетку:

а) $M_A = -0{,}10010$, $p_A = -0001$; $M_B = -0{,}10101$, $p_B = -0100$;

б) $M_A = +0{,}11001$, $p_A = +0010$; $M_B = +0{,}11111$, $p_B = +0110$;

в) $M_A = +0{,}10110$, $p_A = -0001$; $M_B = -0{,}11001$, $p_B = +0010$;

г) $M_A = -0{,}11100$, $p_A = +0001$; $M_B = -0{,}10100$, $p_B = -0001$;

д) $M_A = -0{,}10110$, $p_A = -101$; $M_B = -0{,}11111$, $p_B = -001$;

е) $M_A = -0{,}11110$, $p_A = -010$; $M_B = -0{,}11101$, $p_B = +010$;

ж) $M_A = +0{,}11110$, $p_A = +101$; $M_B = -0{,}10100$, $p_B = +010$;

з) $M_A = -0{,}11010$, $p_A = -0001$; $M_B = -0{,}11011$, $p_B = +0001$;

и) $M_A = +0{,}11010$, $p_A = +0001$; $M_B = +0{,}10101$, $p_B = +0011$;

к) $M_A = -0{,}10011$, $p_A = +0010$; $M_B = +0{,}11101$, $p_B = -0010$.

2. Выполнить преобразование числа с плавающей запятой из формы представления его в ЭВМ в естественную форму записи.

а) *C*1240000; б) *C*2060000; в) 42*C*68000; г) 42870000; д) 42590000;
е) *C*2010000; ж) *C*2*B*18000; з) 42*CB*000; и) 429*F*0000; к) 42*CA*8000.

3. Выполнить преобразование числа с плавающей запятой из естественной формы записи в форму представления его в ЭВМ.

а) +22,25; б) +77,75; в) – 88,25; г) +96,5; д) –65,5;
е) +100,5; ж) –93,75; з) +15,625; и) –44,25; к) +88,25.

Тема 4. Машинные методы умножения чисел

Контрольные вопросы

1. Что такое частичное произведение и частичная сумма?
2. По какому признаку разделяются четыре алгоритма умножения на группы?
3. В каких алгоритмах умножения выполняется сдвиг частичных сумм?
4. В каких алгоритмах умножения выполняется сдвиг частичных произведений?
5. Какие основные элементы образуют структурную схему устройства умножения?

Умножение чисел в прямых кодах

Прежде чем приступать к выполнению практического задания рекомендуется ознакомиться с материалом в [4, с. 37–42].

Рассмотрим примеры умножения чисел алгоритмами умножения А, Б, В и Г.

Пример 1. Выполнить умножение двух чисел Мн = 0,10101 и Мт = 0,10111, используя алгоритм А.

Мн = 0,10101		Мт = 0,10111
		b_5 b_1
0,00000		
0,10101		$\Pi_1 = \text{Мн} \cdot b_1$
0,10101		$\sum_1$
0,01010	1	$\sum_1 \cdot 2^{-1}$
0,10101		$\Pi_2 = \text{Мн} \cdot b_2$
0,11111	1	$\sum_2$
0,01111	11	$\sum_2 \cdot 2^{-1}$
0,10101		$\Pi_3 = \text{Мн} \cdot b_3$
1,00100	11	$\sum_3$
0,10010	011	$\sum_3 \cdot 2^{-1}$
0,00000		$\Pi_4 = \text{Мн} \cdot b_4$
0,10010	011	$\sum_4$
0,01001	0011	$\sum_4 \cdot 2^{-1}$
0,10101		$\Pi_5 = \text{Мн} \cdot b_5$
0,11110	0011	$\sum_5$
0,01111	00011	$\sum_5 \cdot 2^{-1} = \text{Мн} \cdot \text{Мт}$

В результате получаем Мн · Мт = +0,0111100011.

Пример 2. Выполнить умножение двух чисел Мн = 0,10101 и Мт = 0,10111, используя алгоритм Б.

Мн = 0,10101 Мт = 0,10111

$b_5 \quad b_1$

0,0000000000	$\sum_0$
0,0000010101	$П_1 = Мн \cdot b_1 \cdot 2^0$
0,0000010101	$\sum_1$
0,0000101010	$П_2 = Мн \cdot b_2 \cdot 2^1$
0,0000111111	$\sum_2$
0,0001010100	$П_3 = Мн \cdot b_3 \cdot 2^2$
0,0010010011	$\sum_3$
0,0000000000	$П_4 = Мн \cdot b_4 \cdot 2^3 = 0, т. к. b_4 = 0$
0,0010010011	$\sum_4$
0,0101010000	$П_5 = Мн \cdot b_5 \cdot 2^4$
0,0111100011	$\sum_5$

В результате получаем Мн · Мт = +0,0111100011.

Пример 3. Выполнить умножение двух чисел Мн = 0,10101 и Мт = 0,10111, используя алгоритм В.

Мн = 0,10101 Мт = 0,10111

$b_5 \quad b_1$

0,0000000000	$\sum_0$	сдвиг
0,0000000000	$\sum_0 \cdot 2$	
0,0000010101	$П_1 = Мн \cdot b_5$	
0,0000010101	$\sum_1$	
0,0000101010	$\sum_1 \cdot 2$	
0,0000000000	$П_2 = Мн \cdot b_4$	
0,0000101010	$\sum_2$	
0,0001010100	$\sum_2 \cdot 2$	
0,0000010101	$П_3 = Мн \cdot b_3$	
0,0001101001	$\sum_3$	
0,0011010010	$\sum_3 \cdot 2$	
0,0000010101	$П_4 = Мн \cdot b_2$	
0,0011100111	$\sum_4$	
0,0111001110	$\sum_4 \cdot 2$	
0,0000010101	$П_5 = Мн \cdot b_1$	
0,0111100011	$\sum_5$	

В результате получаем Мн · Мт = +0,0111100011.

Пример 4. Выполнить умножение двух чисел Мн = 0,10101 и Мт = 010111, используя алгоритм Г.

Мн = 0,10101   Мт = 0,10111

b_5 b_1

0,0000000000	$\sum_0$
0,**0**101010000	$П_1 = Мн \cdot b_5 \cdot 2^{-1}$
0,0101010000	$\sum_1$
0,**00**00000000	$П_2 = Мн \cdot b_4 \cdot 2^{-2}$
0,0101010000	$\sum_2$
0,**000**1010100	$П_3 = Мн \cdot b_3 \cdot 2^{-3}$
0,0110100100	$\sum_3$
0,**0000**101010	$П_4 = Мн \cdot b_2 \cdot 2^{-4}$
0,0111001110	$\sum_4$
0,**00000**10101	$П_5 = Мн \cdot b_1 \cdot 2^{-5}$
0,0111100011	$\sum_5$

В результате получаем Мн · Мт = +0,0111100011.

В алгоритме Г производится сдвиг частичных произведений вправо. В этом случае в освобождающийся разряд (справа от запятой) заносится значение, совпадающее со знаковым разрядом.

Практическое задание

Выполнить умножение чисел, используя алгоритмы умножения А, Б, В и Г:

а) Мн = 0,10001, Мт = 0,10111;
б) Мн = 0,10101, Мт = 0,11001;
в) Мн = 0,00110, Мт = 0,11101;
г) Мн = 0,11111, Мт = 0,11011;
д) Мн = 0,01010, Мт = 0,00110;
е) Мн = 0,01110, Мт = 0,11111;
ж) Мн = 0,11011, Мт = 0,01111;
з) Мн = 0,11100, Мт = 0,11111;
и) Мн = 0,00100, Мт = 0,10000;
к) Мн = 0,01110, Мт = 0,01011.

Тема 5. Умножение с хранением переносов

Контрольные вопросы по теме

1. В чем основное отличие умножения с хранением переносов от четырех рассмотренных ранее обычных методов умножения?

2. В чем состоит преимущество умножения с хранением переносов от рассмотренных ранее методов умножения?

3. Какие изменения необходимо внести в схему, чтобы она реализовывала умножение с хранением переносов?

4. Какой алгоритм умножения (А, …, Г) используется при умножении с хранением переносов? Ответ обоснуйте.

5. Обязательно ли на последнем шаге выполнять умножение с учетом переносов?

Умножение с хранением переносов

Прежде чем приступать к выполнению практического задания, рекомендуется ознакомиться с материалом в [4, с. 42–43].

Рассмотрим пример умножения двух чисел с хранением переносов. Применяется алгоритм А.

Мн = 0,11101 Мт = 0,10111

b_5 b_1

0,00000		$\sum_0$
0,00000		перенос
0,11101		$П_1 = Мн \cdot b_1$
0,11101		$\sum_1$
0,00000		перенос
0,00110	1	$\sum_1 \cdot 2^{-1}$
0,11101		$П_2 = Мн \cdot b_2$
0,11011	1	$\sum_2$
0,00100		перенос
0,01101	11	$\sum_2 \cdot 2^{-1}$
0,11101		$П_3 = Мн \cdot b_3$
0,10100	11	$\sum_3$
0,01101		перенос
0,01010	011	$\sum_3 \cdot 2^{-1}$
0,00000		$П_4 = Мн \cdot b_4$
0,00111	011	$\sum_4$
0,01000		перенос
0,00011	1011	$\sum_4 \cdot 2^{-1}$
0,11101		$П_5 = Мн \cdot b_5$
1,01000	1011	$\sum_5$
0,10100	01011	$\sum_5 \cdot 2^{-1}$

В результате получаем $Мн \cdot Мт = +0{,}1010001011$.

При использовании этого метода, в отличие от рассмотренных в предыдущей теме, межразрядный перенос при формировании частичной суммы не выполняется, а сохраняется в специальном выделенном для этого регистре. В каждом такте при формировании очередной частичной суммы выполняется суммирование трех значений: предыдущей частичной суммы, сохраненного переноса и очередного частичного произведения. В последнем такте сложение выполняется уже с учетом межразрядного переноса.

Необходимо отметить, что данный метод может применяться только для алгоритма А (это связано с направлением сдвига частичных сумм).

Практическое задание

Перемножить числа с хранением переносов:

а) Мн = 0,11011, Мт = 0,10111;
б) Мн = 0,01101, Мт = 0,10101;
в) Мн = 0,10111, Мт = 0,01011;
г) Мн = 0,11101, Мт = 0,11011;
д) Мн = 0,11010, Мт = 0,11101;
е) Мн = 0,11001, Мт = 0,11111;
ж) Мн = 0,11101, Мт = 0,10111;
з) Мн = 0,01110, Мт = 0,01101;
и) Мн = 0,11011, Мт = 0,11111;
к) Мн = 0,11010, Мт = 0,11011.

Тема 6. Умножение чисел на два разряда множителя одновременно в прямом коде

Контрольные вопросы

1. Какое основное требование должно выполняться при умножении на два разряда множителя одновременно?
2. Какая из пар множителя подвергается преобразованию? Почему?
3. Можно ли умножать на 4, 8, … разрядов множителя? Если да, то какие преимущества и проблемы будут при этом?
4. Какие проблемы могут возникнуть при умножении по алгоритму В и Г?
5. Почему при преобразовании пары вначале надо сделать перенос в нее (из младшей пары), а затем ее преобразовать?

Умножение чисел на два разряда в прямом коде

Прежде чем приступать к выполнению практического задания рекомендуется ознакомиться с материалом в [4, с. 43−46].

Рассмотрим пример умножения чисел на два разряда множителя одновременно в прямом коде (алгоритм А): Мн = 0101, Мт = 0111101101.

Для преобразования множителя необходимо разбить его на пары. Разбиение на пары для целых чисел следует начинать с младших разрядов.

Вначале выполним преобразование множителя, начиная с младшей пары:

$$\text{Мт} = 0111101101;$$

$$\text{Мт}^{\text{п}} = 1000\bar{1}0\bar{1}01.$$

Из преобразованного множителя видно, что в нем содержатся пары: 00, 01, $0\bar{1}$ и 10. В соответствии с этим возможны следующие частичные произведения:

$$[\text{Мн}]_{\text{доп}} = 0.0101; \quad [-\text{Мн}]_{\text{доп}} = 1.1011; \quad [2\text{Мн}]_{\text{доп}} = 0.1010$$

Умножение выполняется по алгоритму А.

0.0000		$\sum_0$
0.0101		$\Pi_1 = [\text{Мн}]_{\text{доп}}$
0.0101		$\sum_1$
0.0001	01	$\sum_1 \cdot 2^{-2}$
1.1011		$\Pi_2 = [-\text{Мн}]_{\text{доп}}\ (0\bar{1})$
1.1100	01	$\sum_2$
1.1111	0001	$\sum_2 \cdot 2^{-2}$
1.1011		$\Pi_3 = [-\text{Мн}]_{\text{доп}}\ (0\bar{1})$
1.1010	0001	$\sum_3$

11.1110	100001	$\sum_3 \cdot 2^{-2}$
11.1111	10100001	$\sum_4 \cdot 2^{-2}$
00.1010		$П_5 = [2Мн]_{доп}$
00.1001	10100001	$\sum_5$
00.0010	0110100001	$\sum_5 \cdot 2^{-2}$

В результате получим Мн · Мт = +00.00100110100001.

Практическое задание

Перемножить следующие числа, используя метод умножения на два разряда множителя одновременно, для всех алгоритмов умножения (А, Б, В и Г):

а) Мн = 01011, Мт = 110011110;
б) Мн = 0110, Мт = 011101010;
в) Мн = 01100, Мт = 111000111;
г) Мн = 01101, Мт = 001001001;
д) Мн = 0110, Мт = 101010110;
е) Мн = 01110, Мт = 110010011;
ж) Мн = 01111, Мт = 101101101;
з) Мн = 0011, Мт = 010010001;
и) Мн = 0100, Мт = 000111000;
к) Мн = 0010, Мт = 001111100.

Тема 7. Умножение чисел в дополнительном коде

Контрольные вопросы

1. Какие алгоритмы машинного умножения можно использовать для умножения чисел в дополнительном коде?
2. Какие поправки необходимо выполнить при умножении целых чисел в дополнительном коде?
3. Какие поправки необходимо выполнить при умножении дробных чисел при умножении чисел в дополнительном коде?
4. При каком знакосочетании сомножителей поправка на требуется и почему?
5. Когда вводится поправка при умножении чисел (в начале, в конце или на любом этапе умножения)?

Умножение чисел в дополнительном коде

Прежде чем приступать к выполнению практического задания рекомендуется ознакомиться с материалом в [4, с. 46−51].

Умножение чисел в дополнительном коде отличается от умножения в прямом коде тем, что в некоторых случаях в конце операции умножения необходимо выполнить добавление поправки. Величина поправки и ее вывод приведены в [4].

Пример 1. Выполнить умножение чисел в дополнительном коде (алгоритм В). Если Мн < 0, Мт > 0, то поправка может и не вводится [4], но в этом случае частичное произведение берется вместе со знаком.

$-\text{Мн} = 0{,}10101$ $\qquad +\text{Мт} = 0{,}10111$

$[\text{Мн}]_{\text{доп}} = 1{,}01011$ $\qquad b_5 \quad b_1$

0,0000000000	$\sum_0$
1,1111101011	$П_1 = b_5 \cdot \text{Мн} \cdot 2^1$
1,1111101011	$\sum_1$
1,1111010110	$\sum_1 \cdot 2^1$
0,0000000000	$П_2 = b_4 \cdot \text{Мн} = 0$
1,1111010110	$\sum_2$
1,1110101100	$\sum_2 \cdot 2^1$
1,1111101011	$П_3 = b_3 \cdot \text{Мн}$
1,1110010111	$\sum_3$
1,1100101110	$\sum_3 \cdot 2^1$
1,1111101011	$П_4 = b_2 \cdot \text{Мн}$
1,1100011001	$\sum_4$
1,1000110010	$\sum_4 \cdot 2^1$
1,1111101011	$П_5 = b_1 \cdot \text{Мн}$
1,1000011101	$\sum_5$

В результате получим Мн · Мт = −0,1000011101.

Пример 2. Выполнить умножение чисел в дополнительном коде (алгоритм Г): Мн = −0,0110, Мт = −0,0011.

$[Мн]_д = 1{,}1010$
$[Мт]_д = 1{,}1101$
$\Delta = [-Мн]_{доп} = 0{,}0110$

0,0000 0000	$\sum_0^ч$
1,1101 0000	$П_1 = Мн \cdot 2^{-1}$
1,1101 0000	$\sum_1^ч$
1,1110 1000	$П_2 = Мн \cdot 2^{-2}$
1,1011 1000	$\sum_2^ч$
1,1111 1010	$П_4 = Мн \cdot 2^{-3}$
1,1011 0010	$\sum_4^ч$
0,0110	$\Delta = [-Мн]_{доп}$
0,0001 0010	$[Мн \cdot Мт]_{доп}$

В результате получим Мн · Мт = +0,00010010.

Практическое задание

Перемножить числа в дополнительном коде для следующих случаев сочетания знаков сомножителей: Мн > 0, Мт < 0; Мн < 0, Мт > 0; Мн < 0, Мт < 0.

а) Мн = 0,11011, Мт = 0,00110;
б) Мн = 0,01011, Мт = 0,10010;
в) Мн = 0,11101, Мт = 0,00101;
г) Мн = 0,11010, Мт = 0,11110;
д) Мн = 0,01001, Мт = 0,01010;
е) Мн = 0,00011, Мт = 0,00110;
ж) Мн = 0,10101, Мт = 0,10000;
з) Мн = 0,11000, Мт = 0,00111;
и) Мн = 0,11011, Мт = 0,11001;
к) Мн = 0,00111, Мт = 0,10101.

Тема 8. Умножение чисел на два разряда в дополнительном коде

Контрольные вопросы

1. Какие алгоритмы умножения допускают некорректное преобразование множителя при умножении в прямых кодах на два разряда множителя?
2. В чем заключается основное отличие в преобразовании пар множителя в дополнительном коде от преобразования в прямом коде?
3. Вводятся ли поправки при умножении в дополнительном коде на два разряда множителя? Поясните.
4. Что происходит с переносом (если он возникает) из старшей пары множителя?
5. Почему при преобразовании очередной пары вначале надо сделать ее преобразование, а затем добавить в нее перенос?

Умножение чисел на два разряда в дополнительном коде

Прежде чем приступать к выполнению практического задания, необходимо ознакомиться с материалом в [4, с. 51−54].

Рассмотрим пример умножения чисел на два разряда множителя одновременно (алгоритм Г): Мн = −0101, Мт = −1011111, $[Мт]_{доп}$ = $\underbrace{1}\underbrace{01}\underbrace{00}\underbrace{00}\underbrace{1}$.

Выполним преобразование множителя в дополнительном коде. Данный метод умножения отличается от аналогичного метода умножения в прямом коде тем, что преобразованию подвергается не только пара 11, а также пара 10. Обоснование этому приведено в [4].

$$[Мт]^{п}_{доп} = 0\bar{1}\bar{1}00001.$$

В результате преобразования множителя получены пары 00, 01, $0\bar{1}$ и $\bar{1}0$. Подготовим соответствующие этим парам множимые:

[−Мн] = 0.0101;
[Мн] = 1.1011;
[−2Мн] = 0.1010.

В рассматриваемом примере выделенные разряды показывают сдвиг частичного произведения.

0,0000	$\sum_0$
0,**00**0101	$П_1 = [-Мн] \cdot 2^{-2}$
0,000101	$\sum_1$
0,**0000**1010	$П_2 = [-2Мн] \cdot 2^{-4}$
0,00011110	$\sum_2 = \sum_3$
1,**11111111**1011	$П_4 = [Мн] \cdot 2^{-8}$
0,000111011011	$\sum_4 = [Мн \cdot Мт]_{доп}$

В результате получим Мн · Мт = +0,000111011011.

Практическое задание

Перемножить числа, используя метод умножения на два разряда множителя одновременно в дополнительном коде для следующих случаев сочетания знаков сомножителей: Мн > 0, Мт < 0; Мн < 0, Мт > 0; Мн < 0, Мт < 0.

а) Мн = 1010, Мт = 1011110;
б) Мн = 0111, Мт = 0101010;
в) Мн = 1100, Мт = 1000111;
г) Мн = 1101, Мт = 0010011;
д) Мн = 1011, Мт = 1010110;
е) Мн = 1110, Мт = 1110011;
ж) Мн = 0001, Мт = 1001101;
з) Мн = 1011, Мт = 0110001;
и) Мн = 0110, Мт = 0001000;
к) Мн = 1010, Мт = 0011100.

Тема 9. Машинные методы деления чисел

Контрольные вопросы

1. Какие существуют подходы к выполнению операции деления?
2. Какова последовательность действий в алгоритме деления с восстановлением остатка?
3. Какова последовательность действий в алгоритме деления без восстановления остатка?
4. Какова последовательность действий в алгоритме деления в дополнительных кодах?
5. Приведите схему устройства деления?

Машинные методы деления чисел

Прежде чем приступать к выполнению практического задания рекомендуется ознакомиться с материалом в [4, с. 56−60].

Пример 1. Выполнить деление чисел в прямом коде с восстановлением остатка.

Дм = 0,10101 Дт = 0,11011

$[-Дт]_{доп}$ = 1,00101

		0,10101	Дм (A_0)
	+	1,00101	$[-Дт]_{доп}$
$A_1 < 0$		**1**,11010	$A_1 < 0$
	+	0,11011	+ Дт (восстановление A_0)
		0,10101	A_0
		1,01010	$A_0 \cdot 2$
	+	1,00101	$[-Дт]_{доп}$
$A_2 > 0$		**0**,01111	A_2
		0,11110	$A_2 \cdot 2$
	+	1,00101	$[-Дт]_{доп}$
$A_3 > 0$		**0**,00011	A_3
		0,00110	$A_3 \cdot 2$
	+	1,00101	$[-Дт]_{доп}$
$A_4 < 0$		**1**,01011	A_4
Чт = 0, 1 1 0 0. . .	+	0,11011	+ Дт (восстановление A_0)
		0,00110	$A_3 \cdot 2$
		0,01100	$(A_3 \cdot 2) \cdot 2$
	+	1,00101	$[-Дт]_{доп}$
$A_5 < 0$		1,10001	A_5

Выделенные жирным шрифтом знаковые разряды остатков определяют очередную цифру частного и надо или не надо восстанавливать предыдущий остаток. Если знаковый разряд остатка равен «1» (остаток отрицателен), то в очередной разряд частного заносится «0» и выполняется восстановление предыдущего остатка. Если же знаковый разряд равен «0», то в частное пишется «1» и восстанавливать остаток не надо.

В результате выполнения деления получим Чт = 0,1100.

Пример 2. Выполнить деление чисел в прямом коде без восстановления остатка.

Дм = 0,10101 Дт = 0,11011

$$\begin{array}{lrl}
 & 0{,}10101 & \text{Дм } (A_0)\\
 & {}^{+}\ \underline{1{,}00101} & [-\text{Дт}]_{\text{доп}}\\
A_1 < 0 & \mathbf{1}{,}11010 & A_1 < 0\\
 & 1{,}10100 & A_0 \cdot 2\\
 & {}^{+}\ \underline{0{,}11011} & [\text{Дт}]_{\text{доп}}\\
A_2 > 0 & \mathbf{0}{,}01111 & A_2\\
 & 0{,}11110 & A_2 \cdot 2\\
 & {}^{+}\ \underline{1{,}00101} & [-\text{Дт}]_{\text{доп}}\\
A_3 > 0 & \mathbf{0}{,}00011 & A_3\\
 & 0{,}00110 & A_3 \cdot 2\\
 & {}^{+}\ \underline{1{,}00101} & [-\text{Дт}]_{\text{доп}}\\
A_4 < 0 & \mathbf{1}{,}01011 & A_4\\
\text{Чт} = 0,\ 1\ 1\ 0\ 0 \ldots & 0{,}10110 & A_3 \cdot 2\\
 & {}^{+}\ \underline{0{,}11011} & [\text{Дт}]_{\text{доп}}\\
A_5 < 0 & 1{,}10001 & A_5
\end{array}$$

В этом случае цифра частного определяется так же, как и в предыдущем примере, кроме того, знак остатка определяет, с каким знаком добавлять делитель к остатку после его сдвига. Если знак остатка положительный, он сдвигается влево и делитель добавляется к нему со знаком минус; если же знак остатка отрицательный, он сдвигается влево и делитель добавляется к нему со знаком плюс.

В результате получим Чт = 0,11000.

Пример 3. Выполнить деление чисел в дополнительном коде.

В этом случае цифра частного определяется из сравнения знаков получаемого остатка и делителя. При совпадении знаков в очередной разряд частного заносится «1», иначе «0». Из этого же сравнения знаков определяется и то действие (добавлять или вычитать делитель), которое будет выполнено на следующем шаге деления. То есть если знаки остатка и делителя не совпадают, то к сдвинутому остатку добавляется делитель, иначе (при совпадении знаков) к сдвинутому остатку добавляется делитель с противоположным знаком.

Дм = −0,10101 $[\text{Дм}]_{\text{доп}}$ = 1,01011

Дт = −0,11011 $[\text{Дт}]_{\text{доп}}$ = 1,00101

$[-\text{Дт}]_{\text{доп}}$ = 0,11011

$$\begin{array}{lrl}
 & 1{,}01011 & \text{Дм } (A_0) \\
 & {}^{+}\ \underline{0{,}11011} & [-\text{Дт}]_{\text{доп}} \\
\text{ЗнДт} \ne \text{Зн}A_1 & \mathbf{0}{,}00110 & A_1 < 0 \\
 & 0{,}01100 & A_0 \cdot 2 \\
 & {}^{+}\ \underline{1{,}00101} & [-\text{Дт}]_{\text{доп}} \\
\text{ЗнДт} = \text{Зн}A_2 & \mathbf{1}{,}10001 & A_2 \\
 & 1{,}00010 & A_2 \cdot 2 \\
 & {}^{+}\ \underline{0{,}11011} & [-\text{Дт}]_{\text{доп}} \\
\text{ЗнДт} = \text{Зн}A_3 & \mathbf{1}{,}11101 & A_3 \\
 & 1{,}11010 & A_3 \cdot 2 \\
 & {}^{+}\ \underline{0{,}11011} & [-\text{Дт}]_{\text{доп}} \\
\text{ЗнДт} \ne \text{Зн}A_4 & \mathbf{0}{,}10101 & A_4 \\
\text{Чт} = 0,\ 1\ 1\ 0\ 0.\,.\,. & 1{,}01010 & A_3 \cdot 2 \\
 & {}^{+}\ \underline{1{,}00101} & [-\text{Дт}]_{\text{доп}} \\
\text{ЗнДт} \ne \text{Зн}A_5 & 0{,}01111 & A_5
\end{array}$$

Далее вычисляется знак полученного частного:

ЗнЧт = ЗнДм ⊕ ЗнДт = 1 ⊕ 1 = 0.

В результате получим Чт = +0,110001.

Практическое задание

Выполнить деление чисел в прямом коде с восстановлением остатков, без восстановления остатков и в дополнительном коде для следующих случаев сочетания знаков: Дм > 0, Дт < 0; Дм < 0, Дт > 0; Дм < 0, Дт < 0.

а) Дм = 0,11010, Дт = 0,11110;
б) Дм = 0,00111, Дт = 0,01010;
в) Дм = 0,10100, Дт = 0,11001;
г) Дм = 0,01101, Дт = 0,10011;
д) Дм = 0,10011, Дт = 0,11101;
е) Дм = 0,01110, Дт = 0,10011;
ж) Дм = 0,10001, Дт = 0,11001;
з) Дм = 0,01011, Дт = 0,10101;
и) Дм = 0,10110, Дт = 0,11010;
к) Дм = 0,01010, Дт = 0,11100.