

Содержание

Введение.....	4
1 Перевод чисел, форматы.....	5
1.1 Алгоритм перевода числа.....	5
1.1.1 Перевод смешанного числа.....	5
1.1.2 Перевод чисел с использованием вспомогательных систем счисления.....	6
1.1.3 Представление чисел в формате с фиксированной запятой.....	6
1.1.4 Представление чисел в формате с плавающей запятой.....	7
1.2 Выполнение задания.....	8
1.2.1 Перевод чисел.....	8
1.2.2 Представление чисел с фиксированной запятой в форматах.....	9
1.2.3 Представление чисел с плавающей запятой в форматах.....	10
2 Сложение в различных кодах и формах представления.....	11
2.1 Теоретические сведения о сложении двоичных чисел.....	11
2.2 Выполнение сложения в различных кодах и формах представления.....	13
2.2.1 Сложение чисел с фиксированной запятой в обратном коде.....	13
2.2.2 Сложение чисел с фиксированной запятой в дополнительном коде.....	14
2.2.3 Сложение чисел с фиксированной запятой в модифицированном коде.....	14
2.2.4 Сложение чисел с плавающей запятой.....	15
3 Умножение различными алгоритмами и способами.....	17
3.1 Общие сведения об умножении двоичных чисел.....	17
3.2 Умножение чисел в прямом коде.....	19
3.3 Умножение чисел в дополнительном коде с простой коррекцией.....	20
3.4 Умножение чисел в дополнительном коде с автоматической коррекцией.....	23
3.5 Умножение чисел в прямом коде с ускорением второго порядка..	25
3.5.1 Умножение чисел четвертым способом.....	26
3.5.2 Умножение чисел третьим способом.....	27
3.6 Умножение в форме с плавающей запятой.....	28

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата	коде.....	14
					2.2.3 Сложение чисел с фиксированной запятой в модифицированном	
					коде.....	14
					2.2.4 Сложение чисел с плавающей запятой.....	15
					3 Умножение различными алгоритмами и способами.....	17
					3.1 Общие сведения об умножении двоичных чисел.....	17
					3.2 Умножение чисел в прямом коде.....	19
					3.3 Умножение чисел в дополнительном коде с простой коррекцией.	20
					3.4 Умножение чисел в дополнительном коде с автоматической	
					коррекцией.....	23
					3.5 Умножение чисел в прямом коде с ускорением второго порядка..	25
					3.5.1 Умножение чисел четвертым способом.....	26
					3.5.2 Умножение чисел третьим способом.....	27
					3.6 Умножение в форме с плавающей запятой.....	28

					ТПЖА 09.03.01.066		
Изм.	Лист	№ докум.	Подп.	Дата			
Разраб.	Рзаев А. Э.				Арифметические основы ЭВМ		
Пров.	Шихов М. М.						
					Лит.	Лист	Листов
						2	50
					Кафедра ЭВМ Группа ИВТ-11		

4	Деление различными способами и алгоритмами.....	31
4.1	Краткие сведения о способах деления и оценке погрешности.....	31
4.2	Деление чисел с восстановлением остатка.....	31
4.3	Деление чисел без восстановления остатка.....	34
4.4	Деление чисел в дополнительном коде.....	36
4.5	Деление чисел в формате с плавающей запятой.....	38
5	Сложение двоично-десятичных чисел.....	40
5.1	Код с естественными весами 8-4-2-1.....	40
5.2	Код с избытком три 8-4-2-1+3.....	42
5.3	Код Айкена 2-4-2-1.....	44
5.4	Пентадный код 3a+2.....	46
	Заключение.....	48
	Приложение А (справочное). Библиографический список.....	49
	Приложение Б (справочное). Список сокращений.....	50

Ине. №	Подп. и дата	Взам. инв.	Ине. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				
				Лист
				3

Введение

Информатика – это дисциплина, посвященная автоматизация процесса обработки информации. То есть информатика рождается в тот момент, когда для обработки информации применяется автоматика.

Основной задачей электронно-вычислительных машин (ЭВМ) является вычисление арифметических операций с максимально возможной скоростью и минимальной погрешностью. Эти требования противоречивы и во многих задачах требуется выбрать способ вычисления, обеспечивающий требуемую точность вычислений при приемлемой скорости. Поэтому важно знать и понимать основные способы осуществления арифметических операций на ЭВМ.

Целью данной курсовой работы является изучение алгоритмов сложения, умножения и деления, используемых в современных ЭВМ.

Для достижения данной цели необходимо решить следующие задачи:

- изучить базовые элементы для построения вычислительных устройств;
- реализовать устройства сложения, умножения и деления на их основе;
- показать процесс вычислений на заданных числах;
- произвести проверку результатов вычислений;
- оценить абсолютную и относительную погрешности вычислений.

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата						
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066					Лист
										4

1 Перевод чисел и форматы

Выполнить перевод чисел А и В из одной позиционной системы в другую с использованием промежуточных систем счисления (СС) и изобразить их в форматах современных ЭВМ.

1.1 Теоретические сведения

1.1.1 Перевод смешанного числа из одной СС в другую

Для перевода смешанного числа из одной СС в другую необходимо выполнить следующие действия:

1) Целую часть исходного числа необходимо последовательно делить на основание новой системы, остатки от деления записывать в обратном порядке.

2) Дробную часть исходного числа необходимо умножать на основание новой системы до тех пор, пока не будет получено необходимое количество цифр, целые части записывать в прямом порядке.

Перевод дробей – бесконечный процесс и может быть выполнен лишь приближенно. Чтобы сохранить точность исходной дроби, определяется количество цифр в изображении дроби по новому основанию q по формуле 1.

$$m_2 = \frac{m_1}{\log_p q} \cong \left\lceil \frac{m_1}{\log_p q} \right\rceil + 1, \quad (1)$$

где m_1 - количество цифр в исходной дроби с основанием p;

m_2 - количество цифр в дроби с новым основанием q.

Далее выполняется округление по последнему разряду, после чего этот последний разряд отбрасывается.

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата	ТПЖА 09.03.01.066					Лист
										5
Изм.	Лист	№ докум.	Подп.	Дата						

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

нужно:

- 1) Умножить число X на M и отбросить дробную часть
- 2) ДК получившегося целого числа занести в разрядную сетку

Формат с ФЗ подразумевает, что необходимо учесть абсолютную погрешность.

1.1.4 Представление чисел в формате с плавающей запятой

В формате с плавающей запятой (ПЗ) числа представляются как показано в формуле

$$A = m_A * 2^{P_A}, \quad (3)$$

где m_A – мантисса числа A , P_A – порядок числа A .

Мантисса обязательно должна быть нормализована. Правила нормализации: порядок числа подбирается так, что мантисса представляет дробное число, причем старший разряд дробной является значащей цифрой. При этом в машинном формате сохраняется конечное количество разрядов дробной части.

Формат с ПЗ работает с фиксированной относительной погрешностью.

В машинных форматах вместо порядка используется характеристика (смещенный порядок). В отличие от порядка характеристика - всегда положительное число. Чтобы получить характеристику C_A , необходимо к его порядку прибавить фиксированную константу – смещение (см. формулу 4).

$$C_A = P_A + \Delta \quad (4)$$

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				Лист
				7

Характеристика – ДК с инверсным знаком.

1.2.1 Необходимо осуществить перевод чисел $A = 482.73$ и $B = 601.35$ из 10СС в 2СС через промежуточные системы счисления. Число A нужно перевести с использованием 8СС, а число B – 16СС, выполнить проверку. Результаты отображены на рисунках 1, 2.

$$\begin{array}{lcl} 0.73 * 8 & = & \mathbf{5.84} \\ 0.84 * 8 & = & \mathbf{6.72} \\ 0.72 * 8 & = & \mathbf{5.76} \end{array} \quad \downarrow$$
$$\begin{array}{r|l} 482 & 8 \\ \hline 480 & 60 \\ \hline 2 & 56 \\ & 4 \end{array}$$

Проверка:

$$16\text{CC} \rightarrow 10\text{CC}: \quad 1\text{E2.BA8}_{16} = 1*16^2 + 14*16^1 + 2*16^0 + 11*16^{-1} + 10*16^{-2} + 8*16^{-3} = 482.73_{10}$$

Рисунок 1 – Перевод числа А через восьмеричную систему счисления

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

$$B = 601.35_{10} = 1001011001.0101100_2$$

Смещенный порядок

Двоичная мантисса

Смещенный порядок									Двоичная матрица																						
1	1	0	0	0	1	0	0	1	1	1	1	1	0	0	0	1	0	1	0	1	1	1	0	1	0	1	0	0	0	0	0
1	1	0	0	0	0	1	1	0	0	0	1	1	1	1	0	0	0	1	0	1	0	1	1	1	0	1	0	1	0	0	0

Смещенный порядок

Двоичная мантисса

0	1	0	0	0	1	0	1	0	1	0	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	1	0	0	0	0	0
0	1	0	0	0	0	1	1	0	0	1	0	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	1	0	0	0	0

Рисунок 4 – Изображение чисел А и В в форме с ПЗ в 32-разрядной сетке ЭВМ

2 Сложение в различных кодах и формах представления

2.1 Теоретические сведения о сложении двоичных чисел

Отрицательные числа в ЦВМ представлены в специальных кодах: прямом, обратном и дополнительном.

Прямой код (ПК) представляет абсолютное значение числа с закодированным знаком: «+» – «0», «-» – «1».

Обратный код (ОК) положительного числа совпадает с его прямым кодом, а для отрицательного числа в знаковый разряд заносится «1», а в остальных разрядах цифры заменяются на взаимообратные (0 – 1, 1 – 0), т.е. формируется поразрядное дополнение числа до единицы.

Дополнительный код (ДК) положительного числа совпадает с его прямым кодом, а для отрицательного числа в знаковый разряд заносится «1», а в цифровой части числа цифры заменяются на взаимообратные и к полученному инверсному изображению прибавляется единица в младший разряд, т.е. код является дополнением до основания СС.

Таким образом, положительные числа во всех кодах одинаковы, а отрицательные - различны.

При алгебраическом сложении чисел в ОК со знаковым разрядом оперируют как с разрядом цифровой части числа, а при возникновении единицы переноса из знакового разряда ее прибавляют к младшему разряду числа.

При алгебраическом сложении чисел в ДК результат получают также, а ДК, а при возникновении единицы переноса из знакового разряда ее отбрасывают.

Модифицированные обратный и дополнительный коды (МОК и МДК) имеют для изображения знака два соседних разряда: «+» – «00», «-» – «11». Эти коды используются для обнаружения ситуации переполнения разрядной

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				Лист
				11

сетки (ПРС). ПРС возникает при сложении чисел с ФЗ одинакового знака, когда результат операции выходит за верхнюю границу диапазона представления чисел, это приводит к потере старших разрядов.

Формальным признаком ПРС при использовании МОК и МДК является появление запрещенных комбинаций в знаковых разрядах – «01» или «10».

Для исправления результата можно либо прекратить вычисления и выдать на пульт управления машин сигнал ПРС, либо пересмотреть масштаб результата, сдвинув его вправо на один разряд, а в освободившийся старший знаковый разряд поместить значение младшего знакового разряда.

Сложение чисел в форме с ПЗ. Выполняется в несколько этапов. Числа с ПЗ изображаются двумя частями – мантиссой и порядком:

$$A = \pm m_a * 2^{\pm p_a}; B = \pm m_b * 2^{\pm p_b}, \quad (7)$$

где m_a – мантисса числа а;

m_b – мантисса числа b;

p_a – порядок числа а;

p_b – порядок числа b.

Чтобы сложить их, надо выполнить различные действия над мантиссами и порядками. Поэтому в машинах предусмотрены различные устройства для обработки мантисс и порядков. Мантиссы исходных операндов нормализованы.

1) Выравнивание порядков слагаемых: меньший порядок увеличивается до большего, при этом мантисса меньшего преобразуемого числа денормализуется. В машине выполняется вычитание порядков операндов. Знак и модуль разности порядков определяет, мантиссу какого из слагаемых надо сдвигать вправо и на сколько разрядов.

Ине. №	Подп. и дата	Взам. инв.	Ине. №	Подп. и дата	где m_a – мантисса числа а;	
					m_b – мантисса числа b;	
					p_a – порядок числа а;	
					p_b – порядок числа b.	
<p>Чтобы сложить их, надо выполнить различные действия над мантиссами и порядками. Поэтому в машинах предусмотрены различные устройства для обработки мантисс и порядков. Мантиссы исходных операндов нормализованы.</p> <p>1) Выравнивание порядков слагаемых: меньший порядок увеличивается до большего, при этом мантисса меньшего преобразуемого числа денормализуется. В машине выполняется вычитание порядков операндов. Знак и модуль разности порядков определяет, мантиссу какого из слагаемых надо сдвигать вправо и на сколько разрядов.</p>						
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066	Лист
						12

2) Сложение мантисс операндов по правилам сложения чисел с ФЗ.

3) Нормализация результата, если необходимо. При этом денормализация вправо, т.е. ситуация, когда в старшем разряде двоичной мантиссы «0», требует сдвига мантиссы влево и уменьшения порядка на соответствующее количество единиц. Денормализация влево означает временное ПРС мантиссы суммы, но в отличие от чисел с ФЗ, здесь возможна коррекция: сдвиг мантиссы на один разряд вправо и увеличение на «1» порядка суммы.

При больших величинах порядков возможно возникновение истинного ПРС числа с ПЗ, хотя вероятность этого невелика.

Смещенные порядки используют в большинстве современных ЭВМ для упрощения процесса выравнивания порядков и их сравнения.

При этом для представления порядка применяется специальный дополнительный код с инверсным кодированием знака: «+» – «1», «-» – «0». В результате порядки чисел увеличиваются (в ЕСЭВМ на $2^6 = 64$, в ПЭВМ на $2^7 = 128$), что приводит к смещению всех порядков по числовой оси в положительном направлении. Такие смещенные порядки называют характеристиками, и так как они все оказываются целыми положительными числами, то алгебраическое сложение можно производить без предварительного анализа знаков.

2.2 Выполнение сложения в различных кодах и формах представления

2.2.1 Сложить исходные операнды с ФЗ в ОК ($A > 0$, $B < 0$). Действие изображено на рисунке 5.

Ине. №	Подп. и дата	Взам. инв.	Ине. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				
				Лист
				13

$A = 482.73_{10} = 111100010.1011101_2$
 $B = -601.35_{10} = -1001011001.01011001_2$
 Масштаб: $M = 2^{10}$
 $A_{ПК} = 0.0111100010101110101110$
 $B_{ПК} = 1.1001011001010110011001$
Сложение:
 $A_{OK} = 0.0111100010101110101110$
 $B_{OK} = 1.0110100110101001100110$
 $(A+B)_{OK} = 1.1110001001011000010100$
 $(A+B)_{ПК} = 1.0001110110100111101011$
 $(A+B) = -0001110110.100111101011_2 = -118.62_{10}$
 Проверка: $482.73 - 601.35 = -118.62$

Рисунок 5 – Сложение с ФЗ в ОК

2.2.2 Сложить исходные операнды с ФЗ в ДК ($A < 0, B > 0$).

Действие изображено на рисунке 6.

$A = -482.73_{10} = -111100010.1011101_2$
 $B = 601.35_{10} = 1001011001.01011001_2$
 Масштаб: $M = 2^{10}$
 $A_{ПК} = 1.0111100010101110101110$
 $B_{ПК} = 0.1001011001010110011001$
Сложение:
 $A_{ДК} = 1.1000011101010001010010$
 $B_{ДК} = 0.1001011001010110011001$
 $(A+B)_{ДК} = 0.0001110110100111101011$
 $(A+B)_{ПК} = 0.0001110110100111101011$
 $(A+B) = 0001110110.100111101011_2 = 118.62_{10}$
 Проверка: $601.35 - 482.73 = 118.62$

Рисунок 6 – Сложение с ФЗ в ДК

2.2.3 В третьем пункте необходимо сложить исходные операнды с ФЗ в модифицированном коде. $A < 0, B < 0$. В качестве модифицированного

Инв. №	Подп. и дата	Инв. №	Взам. инв.	Подп. и дата	ТПЖА 09.03.01.066	Лист
						14
	Изм.	Лист	№ докум.	Подп.		Дата

кода выберем МДК. Данное действие изображено на рисунке 7.

$A = -482.73_{10} = -111100010.1011101_2$
 $B = -601.35_{10} = -1001011001.01011001_2$
 Масштаб: $M = 2^{10}$
 $A_{ПК} = 1.0111100010101110101110$
 $B_{ПК} = 1.1001011001010110011001$
Сложение:
 $A_{МДК} = 11.1000011101010001010010$
 $B_{МДК} = \underline{11.0110100110101001100111}$
 $(A+B)_{МДК} = 10.1111000011111010111001$
 10 - запрещенная комбинация, выбираем другой масштаб слагаемых:
 $M = 2^{11}$
 $A_{МДК} = 11.11000011101010001010010$
 $B_{МДК} = \underline{11.10110100110101001100111}$
 $(A+B)_{МДК} = 11.01111000011111010111001$
 $(A+B)_{ПК} = 1.10000111100000101000111$
 $(A+B) = -10000111100.000101000111_2 = -1084.08_{10}$
 Проверка: $-482.73 - 601.35 = -1084.08$

Рисунок 7 – Сложение с ФЗ в МДК

2.2.4 Необходимо сложить исходные операнды с ПЗ, причем $A > 0$, $B > 0$. Представить операнды в разрядной сетке ЭВМ, выровнять порядки, нормализовать мантиссу результата при необходимости. Необходимые действия изображены на рисунке 8.

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				Лист
				15

Под мантииссы со знаком отведем 23 разряда, под порядки со знаком 5 разрядов.

$$A_{ПК} = 111100010.101110101110$$

$$B_{ПК} = 1001011001.010110011001$$

	Мантисса																							Порядок						
A	0	1	1	1	1	0	0	0	1	0	1	0	1	1	1	0	1	0	1	1	1	0	0	0	0	0	1	0	0	1
B	0	1	0	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	1	1	0	0	1	0	0	1	0	0	1	0

Выполним выравнивание порядков, для чего выполним их вычитание в ДК.

$$P_A = 1001$$

$$P_A^{ПК} = P_A^{ДК} = 0.1001$$

$$P_B = 1010$$

$$-P_B^{ПК} = 1.1010$$

$$-P_B^{ДК} = \underline{1.0110}$$

$$(P_A - P_B)^{ДК} = 1.1111$$

$(P_A - P_B)_{ПК} = 1.0001$; $(P_A - P_B) = -1 \rightarrow$ мантииссу числа A надо сдвинуть на 1 разряд вправо (денормализовать) и увеличить порядок на 1:

A	0	0	1	1	1	1	0	0	0	1	0	1	0	1	1	1	0	1	0	1	1	1	0	0	0	1	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Складываем мантииссы в ДК:

$$m_A^{ПК} = m_A^{ДК} = 0.01111000101011101011100$$

$$m_B^{ПК} = m_B^{ДК} = \underline{0.10010110010101100110010}$$

$$(m_A + m_B)^{ПК} = (m_A + m_B)^{ДК} = 1.00001111000001010001110$$

Временное ПРС суммы мантиисс.

Необходимо сдвинуть вправо мантииссу на один разряд, с одновременным увеличением порядка на 1.

Денормализация вправо: $(m_A + m_B)_{ДК} = 0.100001111000001010001110$

A+B	0	1	0	0	0	0	1	1	1	0	0	0	0	0	1	0	1	0	0	0	1	1	1	0	0	1	0	1	1
-----	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

$$\text{Проверка: } 10000111100.0001010001110 = 1084.079833_{10} \approx 1084.08$$

$$482.73 + 601.35 = 1084.08$$

Рисунок 8 – Сложение с ПЗ в ДК

Име. №	Подп. и дата	Име. №	Подп. и дата	Взам. име.	Име. №	Подп. и дата						ТПЖА 09.03.01.066		Лист			
														16			
							Изм.	Лист	№ докум.	Подп.	Дата						

3 Умножение различными алгоритмами и способами

3.1 Общие сведения об умножении двоичных чисел

Процесс умножения чисел в двоичной системе счисления прост, так как разрядами множителя могут быть либо «0», либо «1», и, следовательно, частичным произведением в каждом такте цикла умножения будет либо «0», либо множимое. Поэтому в цикле умножения двоичных чисел три элементарных операции:

- анализ цифры очередного разряда множителя
- суммирование множимого с накопленной суммой частичных произведений, если цифра множителя «1»
- сдвиги в каждом такте умножения

Умножение можно выполнять как с младших, так и со старших разрядов множителя, со сдвигом, как частичной суммы, так и множимого в процессе умножения. Этим объясняется существование четырех способов умножения чисел.

Следует обратить внимание на то, что множитель сдвигается во всех способах умножения, так как в каждом такте анализируется очередной разряд: при умножении с младших разрядов сдвиг вправо (в сторону младших разрядов), при умножении со старших разрядов множитель сдвигается влево. И еще одна особенность, позволяющая легко запомнить способы умножения: сумма частичных произведений обычно сдвигается в ту же сторону, что и множитель, а множимое сдвигается навстречу множителю, т.е. в противоположную сторону.

I способ – умножение с младших разрядов множителя со сдвигом суммы частичных произведений вправо.

Устройства, которые хранят операнды, регистры, имеют следующую

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата					Лист 17
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066				

разрядность:

- регистры множителя и множимого – n -разрядные
- регистр частичных произведений – $2n$ -разрядный

Суммирование множимого следует выполнять в старшие n разрядов регистра суммы частичных произведений. Причем разрядность его можно уменьшить вдвое, до n -разрядов, помещая при сдвиге младшие разряды суммы на место освобождающихся разрядов регистра множителя.

Особенность I способа умножения состоит в том, что имеется возможность временного переполнения разрядной сетки в регистре суммы частичных произведений, которое ликвидируется при очередном сдвиге вправо.

II способ – умножение с младших разрядов множителя со сдвигом множимого влево.

Этот способ требует n -разрядного регистра множителя и двух $2n$ -разрядных регистров множимого и суммы частичных произведений. Причем, первоначально множимое помещается в младшие разряды регистра, а затем в каждом такте сдвигается на один разряд влево.

III способ – умножение со старших разрядов множителя со сдвигом суммы частичных произведений влево.

Этот способ требует два n -разрядных регистра множителя и множимого и одного $2n$ -разрядных регистра суммы частичных произведений. На схеме видно, что суммирование множимого следует выполнять в младшие n разрядов регистра суммы частичных произведений.

Особенность III способа умножения состоит в том, что в последнем такте не следует выполнять сдвиг в регистре сумм частичных произведений.

IV способ – умножение со старших разрядов множителя со сдвигом множимого вправо.

Этот способ требует одного n -разрядного регистра множителя и двух

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата						
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066					Лист
										18

2n-разрядных регистров множимого и суммы частичных произведений. Причем первоначально множимое помещается в старшие разряды регистра, а затем в каждом такте сдвигается на один разряд вправо.

Особенность IV способа умножения состоит в том, что перед началом цикла умножения следует множимое сдвинуть на один разряд вправо.

Все приведенные выше четыре способа используют как в алгоритмах умножения в прямом коде, так и в алгоритмах умножения в дополнительном коде.

3.2 Умножение чисел в прямом коде

Алгоритм умножения двоичных чисел в ПК:

- 1) Определить знак произведения путем сложения по модулю два знаковых разрядов сомножителей.
- 2) Перемножить модули сомножителей одним из четырех способов.
- 3) Присвоить полученному произведению знак из п.1.

Операнды: $C = 52$, $D = 56$. В данном задании число C – положительное, число D – отрицательное. Необходимо представить числа в форме с ФЗ в ПК и перемножить их первым способом, проверить результат операции. Данное действие представлено на рисунке 9.

Име. №	Подп. и дата	Име. №	Взам. инв.	Подп. и дата	Име. №	Лист
	Изм.	Лист	№ докум.	Подп.		

ТПЖА 09.03.01.066

19

Масштаб: $M = 2^7$

$S_{ПК} = 0.0110100$ $D_{ПК} = 1.0111000$

Знак произведения: $0 \oplus 1 = 1$;

Перемножение модулей – 1 способ:

Множитель →	Сумматор →	Пояснения
0.0110100	0.0000000000000000	Сдвиги
0.0011010	0.0000000000000000	Сдвиги
0.0001101	0.0000000000000000 0.0111000000000000 0.0111000000000000	Сложение Сдвиги
0.0000110	0.0011100000000000	Сдвиги
0.0000011	0.0001110000000000 0.0111000000000000 0.1000110000000000	Сложение Сдвиги
0.0000001	0.0100011000000000 0.0111000000000000 0.1011011000000000	Сложение Сдвиги
0.0000000	0.0101101100000000	Сдвиги
0.0000000	0.0010110110000000	Результат

$(C * D)_{ПК} = 1.00101101100000$ ($M = 2^{14}$)

Проверка: $C * D = -101101100000 = -2912_{10}$

Рисунок 9 – Умножение в ПК первым способом

3.3 Умножение чисел в дополнительном коде с простой коррекцией

Алгоритм умножения двоичных чисел в форме с ФЗ в ДК с простой коррекцией:

- 1) Определить знак произведения путем сложения по модулю два знаковых разрядов сомножителей.
- 2) Перемножить модули сомножителей, представленных в ДК, одним из четырех способов - получить псевдопроизведение.
- 3) Если хотя бы один из сомножителей отрицателен, выполнить

Ине. №	Подп. и дата	Взам. ине.	Ине. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				Лист
				20

коррекцию по следующим правилам:

- если один сомножитель отрицателен, к псевдопроизведению прибавляется дополнительный код от модуля положительного сомножителя;
- если оба сомножителя отрицательны, к псевдопроизведению прибавляются дополнительные коды от модулей дополнительных кодов обоих сомножителей, т.е. их прямые коды.

4) Присвоить модулю произведения знак из п.1 данного алгоритма.

В данном задании оба числа C и D – отрицательны. Необходимо представить числа в форме с ФЗ в ДК и перемножить их третьим способом с простой коррекцией, проверить результат операции. Данное действие представлено на рисунке 10.

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата						
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066					Лист
										21

Масштаб: $M = 2^7$

$C_{ДК} = 1.1001100$ $D_{ДК} = 1.1001000$

Знак произведения: $1 \oplus 1 = 0$;

Перемножение модулей – 3 способ:

← Множитель	← Сумматор	Пояснения
0.1001100	$\begin{array}{r} 0.00000000000000 \\ 0.00000001001000 \\ 0.00000001001000 \end{array}$	Сложение Сдвиги
0.0011000	0.00000010010000	Сдвиги
0.0110000	0.00000100100000	Сдвиги
0.1100000	$\begin{array}{r} 0.00001001000000 \\ 0.00000001001000 \\ 0.00001010001000 \end{array}$	Сложение Сдвиги
0.1000000	$\begin{array}{r} 0.00010100010000 \\ 0.00000001001000 \\ 0.00010101011000 \end{array}$	Сложение Сдвиги
0.0000000	0.00101010110000	Сдвиги
0.0000000	0.01010101100000	Нет последнего сдвига

Псевдопроизведение: 0.010101011000000

Коррекция:

$$\begin{array}{r} 0.01010101100000 \\ + \\ 0.01101000000000 \\ \hline 0.10111101100000 \\ + \\ 0.01110000000000 \\ \hline 1.00101101100000 \end{array}$$

$M = 2^{14}$

Проверка: $C * D = 0.00101101100000 * M = 2912_{10}$

Рисунок 10 – Умножение в ДК с простой коррекцией

Ине. №	Подп. и дата	Взам. инв.	Ине. №	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066	Лист
						22

3.4 Умножение чисел в дополнительном коде с автоматической коррекцией

Теорема: чтобы получить число, представленное в ДК, достаточно просуммировать вклады переходов между группами нулей и единиц кода

При таком подходе становится возможным выполнение умножения непосредственно в ДК, анализируя в i -м такте умножения пары разрядов множителя. При этом к регистру сумм частных произведений (СЧП) множимое либо прибавляется, либо вычитается.

В процессе умножения анализируются две смежные цифры множителя: та, на которую выполняется умножение в данном такте – m_1 , и соседняя младшая цифра – m_2 . В двоичном множителе этой паре соответствуют четыре возможных набора – «00», «01», «10», «11», каждый из которых требует выполнения следующих действий:

- "00" и "11" не требуют никаких действий;
- "01" - прибавить множимое;
- "10" - вычесть множимое.

В каждом такте умножения выполняются сдвиги на один разряд. При этом могут использоваться все способы умножения с некоторыми особенностями:

1) в I и II способе не следует выполнять последний сдвиг суммы частичных произведений;

2) в IV способе не выполняется первый сдвиг множимого. Это объясняется тем, что в этих тактах реализуется умножение не на цифровой, а на знаковый разряд числа.

Кроме того, при выполнении алгоритма умножения с автоматической коррекцией следует помнить о правилах сдвига отрицательных чисел в ДК: при сдвиге влево освобождающиеся младшие разряды заполняются нулями,

Име. №	Подп. и дата	Име. №	Подп. и дата			
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066	Лист
						23

при сдвиге вправо освобождающиеся старшие разряды заполняются единицами, т.е. реализуется арифметический сдвиг числа.

В данном задании число С – отрицательно, число D – положительно. Необходимо представить числа в форме с ФЗ в ДК и перемножить их первым способом с автоматической коррекцией, проверить результат операции. Данное действие представлено на рисунке 11.

Масштаб: $M = 2^7$

$C_{\text{ДК}} = 1.1001100$ $D_{\text{ДК}} = 0.0111000$

Перемножение чисел – 2 способ с автокоррекцией:

множитель →	← множимое	Сумматор	Пояснения
1.1001100 <u>0</u>	0.00000000111000	0.00000000000000	Сдвиги
1.1100110 <u>0</u>	0.00000001110000	0.00000000000000	Сдвиги
1.111001 <u>1</u> 0	0.00000011100000	0.00000000000000 <u>1.11111100100000</u> 1.11111100100000	Вычитание Сдвиги
1.111100 <u>1</u> 1	0.00000111000000	1.11111100100000	Сдвиги
1.1111100 <u>1</u>	0.00001110000000	1.11111100100000 <u>0.00001110000000</u> 0.00001010100000	Сложение Сдвиги
1.1111110 <u>0</u>	0.00011100000000	0.00001010100000	Сдвиги
1.1111111 <u>0</u>	0.00111000000000	0.00001010100000 <u>1.11001000000000</u> 1.11010010100000	Вычитание Сдвиги
1.11111111	0.01110000000000	1.11010010100000	Результат

$(C * D)_{\text{ДК}} = 1.11010010100000$ ($M = 2^{14}$)

Проверка: $C * D = -101101100000 = -2912_{10}$

Рисунок 11 – Умножение в ДК с автокоррекцией

Ине. №	Взам. инв.	Подп. и дата	Ине. №	Подп. и дата

3.5 Умножение чисел в прямом коде с ускорением второго порядка

Алгоритм умножения чисел с ФЗ в ПК с ускорением второго порядка

В данном методе ускорения работают с четверичными цифрами. Разряды двоичного числа группируются по два и сдвиги множителя (а также множимого или суммы частичных произведений) выполняются сразу на два двоичных разряда.

Количество разрядов двоичной сетки выбирается кратным двум. Такой подход сокращает количество шагов умножения вдвое.

На i -м шаге умножения при анализе пары двоичных разрядов (a_{2i+1} , a_{2i}) множителя A должны выполняться следующие действия:

1) набор «00» не требует сложения множимого с предыдущей суммой частичных произведений;

2) набор «01» требует сложения множимого с предыдущей суммой частичных произведений;

3) набор «10» требует сложения удвоенного множимого из предыдущей суммы частичных произведений (в процессе умножения легко получить удвоенное множимое «на лету», с помощью сдвига);

4) наборы «11» требует сложения утроенного множимого (M) из предыдущей суммы частичных произведений.

a_{2i+1}	a_{2i}	Действия над СЧП
0	0	+0, нет действий
0	1	+M, прибавить множимое
1	0	+2M, прибавить множимое, сдвинутое на один разряд влево
1	1	+3M, прибавить утроенное множимое

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата						
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066					Лист
										25

При этом могут использоваться все четыре способа умножения с некоторыми особенностями:

– в I и II способе, при получении набора «11» работает правило: «На текущем шаге умножения вместо сложения с утроенным множимым можно выполнить вычитание множимого (-М) и учесть единицу переноса в старшую на следующем шаге»;

– в III и IV способе, при получении набора «11» работает правило: «На текущем шаге умножения вместо сложения с удвоенным множимым следует выполнить вычитание множимого (-2М)».

– Для способов, в которых анализ идет со старших разрядов (III и IV) нужно рассматривать не только 2 текущих разряда, но и перенос, для того, чтобы предсказать последующие действия с СЧП, которые соответствуют таблице:

a_{i+1}	a_i	a_{i-1}	Действия над СЧП
0	0	0	+0
0	1	0	+М
1	0	0	+2М
1	1	0	-М
0	0	1	+М
0	1	1	+2М
1	0	1	-М
1	1	1	+0

3.5.1 В данном задании число С – положительно, число D – отрицательно. Необходимо представить числа в форме с ФЗ в ПК и перемножить их четвертым способом с ускорением второго порядка, проверить результат операции. Данное действие представлено на рисунке 12.

Ине. №	Подп. и дата	Ине. №	Взам. инв.	Ине. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066
					Лист 26

Масштаб: $M = 2^6$

$C_{ПК} = 0.110100$ $D_{ПК} = 1.111000$

Перемножение модулей – 4 способ с ускорением второго порядка:

← Множитель	Множимое →	Сумматор	Пояснения
0 110100	0.011100000000	$\begin{array}{r} 0.000000000000 \\ 0.111000000000 \\ \hline 0.111000000000 \end{array}$	+2D Сдвиги
1 010000	0.000111000000	$\begin{array}{r} 0.111000000000 \\ 0.111001000000 \\ \hline 0.110001000000 \end{array}$	-D Сдвиги
1 000000	0.000000111000	$\begin{array}{r} 0.110001000000 \\ 0.111100100000 \\ \hline 0.101101100000 \end{array}$	-2D Сдвиги
0 000000	0.000000011100	0.101101100000	Результат

$(C * D)_{ПК} = 1.101101100000$ ($M = 2^{12}$)

Проверка: $C * D = -101101100000 = -2912_{10}$

Рисунок 12 – Умножение в ПК с ускорением второго порядка четвертым способом

3.5.2 В данном задании число C – отрицательно, число D – положительно. Необходимо представить числа в форме с ФЗ в ПК и перемножить их третьим способом с ускорением второго порядка, проверить результат операции. Данное действие представлено на рисунке 13.

Инв. №	Подп. и дата	Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата	
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066		Лист
							27

Масштаб: $M = 2^6$

$C_{ПК} = 1.110100$ $D_{ПК} = 0.111000$

Перемножение модулей – 3 способ с ускорением второго порядка:

← Множитель	← Сумматор	Пояснения
00 110100	0.000000000000 0.000000111000 0.000000111000	+D Сдвиги
11 010000	0.000011100000 0.111111001000 0.000010101000	-D Сдвиги
01 000000	0.001010100000 0.000000111000 0.001011011000	+D Сдвиги
00 000000	0.101101100000	Результат

$(C * D)_{ПК} = 1.101101100000$ ($M = 2^{12}$)

Проверка: $C * D = -101101100000 = -2912_{10}$

Рисунок 13 – Умножение в ПК с ускорением второго порядка третьим способом

3.6 Умножение чисел в форме с ПЗ

Алгоритм умножения двоичных чисел в форме с ПЗ:

Если числа представлены в виде формулы (7), то их произведение определяется следующим образом:

$$C = A \times B = \pm m_A * m_B * 2^{\pm(P_A+P_B)} = \pm m_C * 2^{\pm P_C}, \quad (8)$$

где m_C – мантисса числа C ,

P_C – порядок числа C ,

т.е. мантисса произведения m_C равна произведению мантисс сомножителей, а порядок P_C – сумме порядков сомножителей.

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата						
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066					Лист
										28

Это позволяет сформулировать алгоритм умножения чисел в форме ПЗ.

1) Определить знак произведения путем сложения по модулю два знаковых разрядов сомножителей.

2) Перемножить модули мантисс сомножителей по правилам умножения дробных чисел с ФЗ.

3) Определить порядок произведения алгебраическим сложением порядков сомножителей с использованием модифицированного дополнительного или обратного кодов.

4) Нормализовать мантиссу результата и выполнить округление, если это необходимо.

Так как мантиссы исходных сомножителей нормализованы, то денормализация мантиссы произведения возможна только на один разряд.

При умножении чисел с ПЗ возможно возникновении ПРС при сложении порядков, поэтому необходимо предусматривать выявление признаков ПРС в устройствах умножения чисел с ПЗ.

В данном задании оба числа С и D – положительные. Необходимо представить их в форме с плавающей запятой в разрядной сетке условной машины (с порядками). При умножении мантисс использовать второй способ умножения. Выполнив последовательно все пункты алгоритма умножения чисел в форме с ПЗ, изобразить произведение в разрядной сетке выбранной условной машины и проверить результат. Данное действие представлено на рисунке 14.

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата						
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066					Лист
										29

4 Деление различными способами и алгоритмами

4.1 Краткие сведения о способах деления и оценке погрешности

Процесс деления состоит из последовательности операций вычитания и сдвигов, при этом операция вычитания заменяется операцией сложения остатка с делителем, представленным в обратном или дополнительном кодах.

При делении чисел в прямом коде знак частного определяется сложением по модулю два знаковых разрядов делимого и делителя, и далее в процессе деления участвуют модули операндов.

Так как операция деления обратная умножению и начинается всегда со старших разрядов, то существуют два способа деления – обращенный третий и четвертый способы умножения. Причем нередко для реализации умножения и деления целесообразно использовать одно и то же оборудование: регистр множимого как регистр делителя, регистр множителя – как регистр частного, а регистр частных сумм – как регистр делимого, в который затем заносят остатки от деления.

Приведенные выше два способа деления можно выполнять, используя два алгоритма:

- с восстановлением остатков
- без восстановления остатков

4.2 Деление с восстановлением остатка

Алгоритм деления с восстановлением остатков:

- 1) Определить знак частного сложением по модулю 2 знаковых разрядов делимого и делителя. Далее использовать модули операндов.
- 2) Вычесть из делимого делитель, путем сложения в ОК или ДК.

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				Лист
				31

- 3) Проанализировать знак остатка после первого вычитания:
 - если остаток положительный, то произошло ПРС! Операцию следует прекратить для смены масштабов операндов;
 - если остаток отрицательный, то в частное заносится 0 (в последующем, он будет в знаковом разряде), восстановить остаток, прибавив к нему делитель. В цикле формирования цифр частного вычитать из остатка делитель, прибавляя его в ОК или ДК.
- 4) Проанализировать знак полученного остатка:
 - если >0 , то в частное заносится 1
 - если <0 , то в частное заносится 0
- 5) Восстановить отрицательный остаток, сложив его с делителем.
- 6) Выполнить сдвиги.
- 7) Сформировать $(n+1)$ -ый остаток для округления частного.
- 8) Выполнить округление результата и присвоить частному знак из п.1.

Необходимо разделить исходные операнды первым способом с восстановлением остатков, при вычитании использовать ДК, оценить погрешность результата. Число С следует взять положительным, а число D – отрицательным, причем С – делимое. Данные действия показаны на рисунке 15.

Ине. №	Подп. и дата	Взам. инв.	Ине. №	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066	Лист
						32

Масштаб: $M = 2^7$

$C = 52_{10} = 110100_2$; $D = -56_{10} = -111000_2$

$C_{ПК} = 0.0110100$ $D_{ПК} = 1.0111000$

Знак произведения: $0 \oplus 1 = 1$;

Деление модулей - 1 способ (с восстановлением остатка с использованием дополнительного кода при вычитании):

Частное	Делимое (остатки)	Пояснения
0.0000000	0.0110100 <u>1.1000111</u>	Вычитание
0.0000000	1.1111011 (+1) <u>0.0111000</u>	1 остаток Восстановление
	0.0110100 (+1) <u>0.1101000</u>	Сдвиги
	<u>1.1000111</u>	Вычитание
0.0000001	0.0110000 (+1) <u>0.1100000</u>	2 остаток, сдвиги
	<u>1.1000111</u>	Вычитание
0.0000011	0.0101000 (+1) <u>0.1010000</u>	3 остаток, сдвиги
	<u>1.1000111</u>	Вычитание
0.0000111	0.0011000 (+1) <u>0.0110000</u>	4 остаток, сдвиги
	<u>1.1000111</u>	Вычитание
0.0001110	1.1110111 <u>0.0111000</u>	5 остаток
	0.0110000 (+1) <u>0.1100000</u>	Восстановление
	<u>1.1000111</u>	Сдвиги
0.0011101	0.0101000 (+1) <u>0.1010000</u>	Вычитание
	<u>1.1000111</u>	6 остаток, сдвиги
0.0111011	0.0011000 (+1) <u>0.0110000</u>	Вычитание
	<u>1.1000111</u>	7 остаток, сдвиги
0.1110110	1.1110111 <u>0.0111000</u>	8 остаток
1.1110110	0.0110000 (+1)	Восстановление Результат

Делитель: 1.0111000

$(C/D)_{ПК} = 1.1110110 = -0.921875$

Проверка: $52 / (-56) = -0.928571$

Абсолютная погрешность:

$\Delta = |-0.921875 - (-0.928571)| = 0.006696$

Относительная погрешность:

$\delta \approx (|0.006696| / |-0.921875|) * 100\% = 7.2\%$

Рисунок 15 – Деление с ФЗ с восстановлением остатков

Инв. №	Взаим. инв.	Подп. и дата	Инв. №	Подп. и дата					
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066				
					Лист				
					33				

4.3 Деление без восстановления остатка

Алгоритм деления без восстановления остатков:

1) Определить знак частного путем сложения по модулю два знаковых разрядов делителя и делимого. Далее использовать модули операндов.

2) Вычесть из делимого делитель путем сложения в ДК или ОК.

3) Проанализировать знак остатка после первого вычитания:

– если положителен, то произошло ПРС, операцию следует прекратить для смены масштаба операндов

– если остаток отрицателен, то в частное занести 0 и продолжить операцию деления

4) Выполнить сдвиги.

5) Если до сдвига остаток был положителен, то вычесть из остатка делитель, если был отрицателен – прибавить к остатку делитель.

6) Если вновь полученный остаток положителен, то в очередной разряд частного занести 1, в противном случае 0.

7) Выполнить пп. 4-6 алгоритма (n+1) раз, причем последний сдвиг частного не выполнять, т.к. (n+1)-ый разряд формируется для округления.

8) Выполнить округление результата и присвоить частному знак из п.1 алгоритма.

Необходимо разделить исходные операнды вторым способом без восстановления остатков, применяя ДК при вычитании, оценить погрешность результата. Операнды следует взять отрицательными. Данные действия показаны на рисунке 16.

Ине. №	Подп. и дата	Взам. инв.	Ине. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				Лист
				34

Масштаб: $M = 2^7$

$C = -52_{10} = 110100_2$; $D = -56_{10} = -111000_2$

$C_{ПК} = 1.0110100$ $D_{ПК} = 1.0111000$

Знак произведения: $1 \oplus 1 = 0$;

Деление модулей - 2 способ (без восстановления остатков с использованием дополнительного кода при вычитании):

← Частное	Делитель →	Делимое (остатки)	Пояснения
0.0000000	0.01110000000000	0.01101000000000 <u>1.10010000000000</u>	Вычитание
0.0000000		1.11111000000000	Сдвиги
	0.00111000000000	1.11111000000000 <u>0.00111000000000</u>	Сложение
0.0000001		0.00110000000000	Сдвиги
	0.00011100000000	0.00110000000000 <u>1.11100100000000</u>	Вычитание
0.0000011		0.00010100000000	Сдвиги
	0.00001110000000	0.00010100000000 <u>1.11110010000000</u>	Вычитание
0.0000111		0.00000110000000	Сдвиги
	0.00000111000000	0.00000110000000 <u>1.11111001000000</u>	Вычитание
0.0001110		1.11111111000000	Сдвиги
	0.00000011100000	1.11111111000000 <u>0.00000011100000</u>	Сложение
0.0011101		0.00000010100000	Сдвиги
	0.00000001110000	0.00000010100000 <u>1.11111110010000</u>	Вычитание
0.0111011		0.00000000110000	Сдвиги
	0.00000000111000	0.00000000110000 <u>1.11111111001000</u>	Вычитание
0.1110110		1.11111111110000	Результат

$(C/D)_{ПК} = 0.1110110 = 0.921875$

Проверка: $52 / (-56) = 0.928571$

Абсолютная погрешность:

$\Delta = |0.921875 - 0.928571| = 0.006696$

Относительная погрешность:

$\delta \approx (|0.006696| / |0.921875|) * 100\% = 7.2\%$

Рисунок 16 – Деление с ФЗ без восстановления остатка

Инв. №	Подп. и дата	Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата	Инв. №	Подп. и дата	Лист 35
	Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066			

4.4 Деление в ДК

Алгоритм деления:

- 1) Если знаки делимого и делителя совпадают, то в частное заносится 0, в противоположном случае – 1. Этот разряд знаковый.
- 2) Если знаки операндов совпадают, то делитель вычитается из делимого, в противном случае делитель прибавляется к делимому.
- 3) Если знак первого остатка совпадает со знаком делимого, то произошло ПРС! Операцию деления прекратить.
- 4) Выполнить сдвиги.
- 5) Все последующие остатки формируются по правилу: если знаки делителя и остатка до сдвига совпадают, то делитель вычесть из остатка, в противном случае – делитель прибавить к остатку.
- 6) Если знаки нового остатка и делителя совпадают, то в очередной разряд частного занести 1, в противном случае – 0.
- 7) Выполнить пп. 4-6 (n+1) раз, причем последний сдвиг частного не выполнять. Выполнить округление результата.

Необходимо разделить исходные операнды вторым способом в ДК, оценить погрешность результата. Число С взять отрицательным, а число D - положительным, причем D – делимое. Данные действия показаны на рисунке 17.

Име. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата	ТПЖА 09.03.01.066					Лист
										36
					Изм.	Лист	№ докум.	Подп.	Дата	

Масштаб: $M = 2^7$

$C = -52_{10} = 110100_2$; $D = 56_{10} = 111000_2$

$C_{ПК} = 1.0110100$ $D_{ПК} = 0.0111000$

Знак произведения: $1 \oplus 0 = 1$;

Деление модулей - 2 способ (с автоматической коррекцией):

Частное	Делитель	Делимое (остатки)	Пояснения
0.0000001	1.10011000000000	0.01110000000000 <u>1.10011000000000</u>	Сложение
		0.00001000000000	ПРС
	1.10011000000000	0.00111000000000 <u>1.10011000000000</u>	Сложение
0.0000001		1.11010000000000	Сдвиги
	1.11001100000000	1.11010000000000 <u>0.00110100000000</u>	Вычитание
0.0000010		0.00000100000000	Сдвиги
	1.11100110000000	0.00000100000000 <u>1.11100110000000</u>	Сложение
0.0000101		1.11101010000000	Сдвиги
	1.11110011000000	1.11101010000000 <u>0.00001101000000</u>	Вычитание
0.0001011		1.11110111000000	Сдвиги
	1.11111001100000	1.11110111000000 <u>0.00000110100000</u>	Вычитание
0.0010111		1.11111011000000	Сдвиги
	1.11111100110000	1.11111011000000 <u>0.00000011010000</u>	Вычитание
0.0101110		0.00000000110000	Сдвиги
	1.11111110011000	0.00000000110000 <u>1.11111110011000</u>	Сложение
0.1011101		1.11111111001000	Сдвиги
	1.11111111001100	1.11111111001000 <u>0.00000000110100</u>	Вычитание
1.0111011		1.11111111111100	

$(D/C)_{ДК} = 1.0111011$

$(D/C)_{ПК} = 0.1000101$

Проверка:

$M = M_D / M_C = 2^1$

$56 / (-52) = -1.0769$

$D/C = -00.1000101 * 2 = -1.0781$

Абсолютная погрешность:

$\Delta = |-1.0781 - (-1.0769)| = 0.0012$

Относительная погрешность:

$\delta \approx (|0.0012| / |1.0781|) * 100\% = 1.1 \%$

Рисунок 17 – Деление с ФЗ в ДК

Инв. №	Подп. и дата	Инв. №	Подп. и дата	Взам. инв.	Подп. и дата	Инв. №	
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066		Лист
							37

4.5 Деление с ПЗ

Алгоритм деления:

- 1) Определить знак частного путем сложения по модулю два знаковых разрядов операндов.
- 2) Разделить модуль мантиссы делимого на модуль мантиссы делителя по правилам деления дробных чисел с ФЗ.
- 3) Определить порядок частного вычитанием порядка делителя из порядка делимого, используя ОК или ДК.
- 4) Нормализовать мантиссу результата и присвоить знак из п.1 алгоритма.

В отличие от деления чисел с ФЗ при выполнении п.2 алгоритма, получение положительного остатка при первом вычитании не означает ПРС! При обработке чисел с ПЗ такая ситуация требует денормализации мантиссы делимого, сдвигом её на 1 разряд вправо с одновременным увеличением порядка, делимого на 1.

Необходимо разделить исходные операнды первым способом с ПЗ, используя алгоритм без восстановления остатков для мантисс, оценить погрешность результата. Исходные операнды взять положительными. Представить операнды в условной разрядной сетке ЭВМ, нормализовать при необходимости мантиссу результата. Данные действия показаны на рисунке 18.

Ине. №	Подп. и дата	Взам. инв.	Ине. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				
				Лист
				38

5 Сложение двоично-десятичных чисел

5.1 Код с естественными весами (8-4-2-1)

8-4-2-1 – это простейший код, в котором каждая десятичная цифра представлена в двоичном эквиваленте. Выполнение операции сложения требует перевода из ПК в ОК или ДК. Однако простое инвертирование не дает ОК, так как исходную тетраду оно дополняет до пятнадцати, а не до девяти, как это требуется. Это означает, что в инвертированной тетраде число получается на шесть (0110) больше, чем нужно. Назовем простую инверсию псевдо-ОК. Чтобы получить ОК из псевдо-ОК, необходимо вычесть из последнего двоичную шестерку. Заменяя вычитание сложением в ДК, получим корректирующий код (КК) – 1010. Таким образом для получения ОК, необходимо инвертировать тетрады и прибавлять к ним КК. Единицы переноса на данном этапе отбрасываются. При сложении сумма в k -том разряде определяется формулой

$$\sum_k = A_k + B_k + C_k, \quad (9)$$

где A_k, B_k - слагаемые тетрады;

C_k - единица переноса.

Возможны три случая при сложении тетрад операндов. Первый – сумма меньше десяти, единицы переноса не возникает. Второй – сумма меньше пятнадцати, но больше десяти. Естественно единица переноса в этом случае не возникнет – необходимо сформировать её. Необходим КК – 0110. Признак введения коррекции – неправильные тетрады, у которых единица есть как в старшем разряде, так и во втором или третьем. Третий случай – если сумма больше шестнадцати. Возникает единица переноса, КК – 0110. Из

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата	том разряде определяется формулой
$\sum_K = A_k + B_k + C_k ,$					(9)
где A_k, B_k - слагаемые тетрады;					
C_k - единица переноса.					
Возможны три случая при сложении тетрад операндов. Первый –					
сумма меньше десяти, единицы переноса не возникает. Второй – сумма					
меньше пятнадцати, но больше десяти. Естественно единица переноса в этом					
случаи не возникнет – необходимо сформировать её. Необходим КК – 0110.					
Признак введения коррекции – неправильные тетрады, у которых единица					
есть как в старшем разряде, так и во втором или третьем. Третий случай –					
если сумма больше шестнадцати. Возникает единица переноса, КК – 0110. Из					
					</

приведенного выше вытекает алгоритм сложения в 8-4-2-1:

1) Проверить знаки слагаемых, отрицательные перевести в ОК, инвертированием и прибавлением КК – 1010. Единицы переноса отбрасывать.

2) Сложить двоично-десятичные цифры по правилам сложения в 2СС.

3) Выполнить коррекцию результата, кодом 0110, прибавляя его к «неправильным» тетрадам и тем тетрадам, в которых возникла единица переноса. Единицы переноса при коррекции учитывать.

4) Проверить знак результата, если число отрицательное – перевести в ПК инвертированием тетрад и прибавлением КК- 1010. Единицы переноса отбрасывать.

Код 8-4-2-1 представлен в таблице 1.

Таблица 1 – Код 8-4-2-1

10 сс	2сс- ПК	2сс- ОК
0	0000	1001
1	0001	1000
2	0010	0111
3	0011	0110
4	0100	0101
5	0101	0100
6	0110	0011
7	0111	0010
8	1000	0001
9	1001	0000

Знаки операндов: $A < 0$, $B > 0$. Сложить числа в коде с естественными весами 8-4-2-1 и проверить результат.

Сложение в коде 8-4-2-1 представлено на рисунке 19.

Име. №	Подп. и дата	Име. №	Взам. инв.	Подп. и дата	Име. №	Лист 41
	Изм.	Лист	№ докум.	Подп.		

ТПЖА 09.03.01.066

Проверка: $A + B = 601,35_{10} - 482,73_{10} = 118,62_{10}$

инвертированием тетрад.

Код представлен в таблице 2.

Таблица 2 – Код 8-4-2-1+3

10 сс	2сс- ПК	2сс- ОК
0	0011	1100
1	0100	1011
2	0101	1010
3	0110	1001
4	0111	1000
5	1000	0111
6	1001	0110
7	1010	0101
8	1011	0100
9	1100	0011

Знаки операндов: $A < 0, B < 0$. Сложить числа в коде с избытком три 8-4-2-1+3 и проверить результат.

Сложение представлено на рисунке 20.

Ине. №	Подп. и дата	Взам. инв.	Ине. №	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата

ТПЖА 09.03.01.066

$A_{10} = -0,048273$ $B_{10} = -0,060135$ $M = 10^4$
 ПК: $A_{8-4-2-1+3} = 1,0011.0111.1011.0101.1010.0110$
 $B_{8-4-2-1+3} = 1,0011.1001.0011.0100.0110.1000$
 ОК: $A_{OK} = 1,1100.1000.0100.1010.0101.1001$
 $B_{OK} = 1,1100.0110.1100.1011.1001.0111$

Сложение:

$1,1100.1000.0100.1010.0101.1001$
 $1,1100.0110.1100.1011.1001.0111$
 $1,1000.1111.0001.0101.1111.0000$

Коррекция:

$1,1000.1111.0001.0101.1111.0000$
 $0,0011.1101.0011.0011.1101.0011$
 $1,1011.1100.0100.1000.1100.0011$
 $\underline{\hspace{1.5cm}1\hspace{0.5cm}}$
 $1,1011.1100.0100.1000.1100.0100$

$(A + B)_{OK} = 1,1011.1100.0100.1000.1100.0100$

$(A + B)_{ПК} = 1,0100.0011.1011.0111.0011.1011$

С учетом масштаба, $(A + B)_{8-4-2-1+3} = 0100.0011.1011.0111,0011.1011 = -1084,08$

Проверка: $A + B = -601.35_{10} - 482.73_{10} = -1084,08_{10}$

Рисунок 20 – Сложение в коде 8-4-2-1+3

5.3 Код Айкена (2-4-2-1)

Для данного кода введены специальные правила введения коррекции при алгебраическом сложении:

1) Если каждая из исходных тетрад < 5 , то:

- если суммарная тетрада < 5 , то коррекция не нужна
- если суммарная тетрада ≥ 5 , то коррекция кодом 0110

2) Если одна из исходных тетрад < 5 , а вторая ≥ 5 , то коррекции нет.

3) Если каждая тетрада ≥ 5 , т.е. при сложении возникает единица переноса, то

- если тетрада суммы ≥ 5 , то коррекция не нужна
- если тетрада суммы < 5 , то коррекция кодом 1010

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата	Име. №	Лист	
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066		44

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

- | Инв. № | Подп. и дата | Взам. инв. | Инв. № | Подп. и дата |
|--------|--------------|------------|--------|--------------|
| | | | | |

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата

$A_{10} = 0,48273 \quad B_{10} = -0,60135 \quad M = 10^3$
 ПК: $A_{2-4-2-1} = 0,0100.1110.0010.1101.0011$
 $B_{2-4-2-1} = 1,1100.0000.0001.0011.1011$
 ОК: $A_{OK} = 0,0100.1110.0010.1101.0011$
 $B_{OK} = 1,0011.1111.1110.1100.0100$

Сложение:

$0,0100.1110.0010.1101.0011$
 $1,0011.1111.1110.1100.0100$
 $1,1000.1110.0001.1001.0111$

Коррекция:

$1,1000.1110.0001.1001.0111$
 $0,0110.0000.0000.1010.0110$
 $1,1110.1110.0001.0011.1101$

$(A + B)_{OK} = 1,1110.1110.0001.0011.1101$

$(A + B)_{ПК} = 0,0001.0001.1110.1100.0010$

С учетом масштаба, $(A + B)_{2-4-2-1} = 0001.0001.1110.1100.0010 = -118,62$

Проверка: $A + B = -601.35_{10} + 482.73_{10} = -118,62$

Рисунок 21 – Сложение в коде Айкена 2-4-2-1

5.4 Пентадный код (3a+2)

Алгоритм сложения:

1) Проверить знаки слагаемых, отрицательные преобразовать в ОК, инвертируя пентады.

2) Сложить 2-10 числа по правилам двоичной арифметики, фиксируя единицы переноса между пентадами.

3) Выполнить коррекцию, прибавив код 11110 к пентадам, в которые и из которых не формировались единицы переноса, и, прибавив код 00010 к пентадам, в которые и из которых формировались единицы переноса. Единицы переноса между пентадами при коррекции отбрасывать.

4) Проверить знак результата, отрицательный преобразовать в ПК, инвертируя пентады.

Код представлен в таблице 4.

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				Лист
				46

Таблица 4 – Пентадный код 3а+2

10 сс	2сс- ПК	2сс- ОК
0	00010	11101
1	00101	11010
2	01000	10111
3	01011	10100
4	01110	10001
5	10001	01110
6	10100	01011
7	10111	01000
8	11010	00101
9	11101	00010

Знаки операндов: $A > 0$, $B > 0$. Сложить числа в пентадном коде 3а+2 и проверить результат. Сложение в пентадном коде 3а+2 представлено на рисунке 20.

$$A_{10} = 0,048273 \quad B_{10} = 0,060135 \quad M = 10^4$$

$$\text{ПК: } A_{3a+2} = 0,00010.01110.11010.01000.10111.01011$$

$$B_{3a+2} = 0,00010.10100.00010.00101.01011.10001$$

$$\text{ОК: } A_{\text{ОК}} = 0,00010.01110.11010.01000.10111.01011$$

$$B_{\text{ОК}} = 0,00010.10100.00010.00101.01011.10001$$

Сложение:

$$0,00010.01110.11010.01000.10111.01011$$

$$\underline{0,00010.10100.00010.00101.01011.10001}$$

$$0,00101.00010.11100.01110.00010.11100$$

Коррекция:

$$0,00101.00010.11100.01110.00010.11100$$

$$\underline{0,00000.00000.11110.00000.00000.11110}$$

$$0,00101.00010.11010.01110.00010.11010$$

$$(A + B)_{\text{ОК}} = 0,00101.00010.11010.01110.00010.11010$$

$$(A + B)_{\text{ПК}} = 0,00101.00010.11010.01110.00010.11010$$

$$\text{С учетом масштаба, } (A + B)_{2-4-2-1} = 00101.00010.11010.01110.00010.11010 = 1084,08$$

$$\text{Проверка: } A + B = 601.35_{10} + 482.73_{10} = 1084,08$$

Рисунок 22 – Сложение в пентадном коде

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата	ТПЖА 09.03.01.066					Лист
										47
					Изм.	Лист	№ докум.	Подп.	Дата	

Заключение

В ходе проектирования изучены базовые элементы вычислительных устройств. Из этих элементов построены устройства, выполняющие различные алгоритмы сложения, умножения и деления. Показана работа данных алгоритмов на примере двух чисел. Проведена проверка результата вычисления и оценка абсолютной и относительной погрешности вычисления.

В процессе выполнения работы были освоены способы сложения, умножения, деления чисел в форматах с плавающей запятой и фиксированной запятой. Это позволяет ознакомиться с используемыми алгоритмами вычислений в современных ЭВМ.

Также данная работа позволяет оценить эффективность и удобность того или иного алгоритма вычисления.

Например, для заданных чисел наиболее быстрый способ деления – деление в ПК вторым способом без восстановления остатка.

Также были изучены алгоритмы сложения в двоично-десятичных кодах, которые обычно используются в вычислительной технике, если объем вычислений невелик и проводить преобразования из двоично-десятичного кода в двоичный код перед расчетами, а затем обратное преобразование — не целесообразно.

Име. №	Подп. и дата	Взам. инв.	Име. №	Подп. и дата	ТПЖА 09.03.01.066					Лист
										48
Изм.	Лист	№ докум.	Подп.	Дата						

Приложение А
(справочное)
Библиографический список

1 Фадеева Т.Р. Арифметические основы ЭВМ [Текст] / Фадеева Т.Р., Матвеева Л.И., Долженкова М.Л. – Киров: Изд-во ВятГУ, 2007. – 44 с.

2 Фадеева Т.Р. Сложение и умножение чисел в двоичной системе счисления [Текст] / Фадеева Т.Р., Матвеева Л.И., Шихов М.М. – Киров: Изд-во ВятГУ, 2015. – 53 с.

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата
Изм.	Лист	№ докум.	Подп.	Дата
ТПЖА 09.03.01.066				
				Лист 49

Приложение Б

(справочное)

Список сокращений

ЦЭВМ – цифровая электронно-вычислительная машина

ЕСЭВМ – единая система электронно-вычислительных машин

ПЭВМ – персональная электронно-вычислительная машина

2СС – двоичная система счисления

10СС – десятичная система счисления

Инв. №	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата						
Изм.	Лист	№ докум.	Подп.	Дата	ТПЖА 09.03.01.066					Лист
										50