КУРСОВЫЕ РАБОТЫ (ПРОЕКТЫ)

Щенях Д. С.
(Фамилия, И.О. студента)

№ п/п	Наименование дисциплин (модуля)	Тема курсовой работы (проекта)	Семестр	Оценка	Дата сдачи	Подпись преподавателя	Фамилия преподавателя
1	Информатика	Арифметические основы ЭВМ	II	отлично	05.06.15		Шихов
2	Комп. граф.	Разработка компьютерной игры „Побег из лабиринта“	3	отлично	17.12.15		[illegible]
3.	Теория автоматов	Синтез микропрограммного управляющего автомата	4	хорошо	30.06.16		Мельцов
4.	РПС	Разработка приложения „Мобильный ВятГУ“	5	хорошо	20.01.17		Чистяков
5	Разработка модулей СПО	Разработка программы логического вывода методом деления дизъюнктов	6	отлично	14.06.17		Шихов

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (ВятГУ)
Факультет автоматики и вычислительной техники

ПРОВЕРЕНО

Декан ______________________ (подпись)

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Вятский государственный университет»
(«ВятГУ»)
Факультет автоматики и вычислительной техники
Кафедра электронных вычислительных машин

Допущено к защите
Руководитель проекта
________/Шихов М.М./
(подпись) (Ф.И.О)
«__»__________2015г.

АРИФМЕТИЧЕСКИЕ ОСНОВЫ ЭВМ

Пояснительная записка
Курсовая работа по дисциплине
«Информатика»
ТПЖА 09.03.01.24 ПЗ

Разработал студент группы ИВТ-11 ________________/Щесняк Д.С./

Руководитель ассистент кафедры ЭВМ ____________/Шихов М. М./

Проект защищен с оценкой «_____________» _________________
(оценка) (дата)

Члены комиссии ____________________/ __________________/
(подпись) (Ф.И.О)

___________________ /___________________/

___________________ /___________________/

Киров 2015

Оглавление

1. Перевод чисел из одной позиционной системы в другую

Необходимо выполнить перевод чисел A и B из одной позиционной системы в другую с использованием промежуточных систем счисления и изобразить их в форматах современных ЭВМ.

1.1 Теоретическая часть

Любое смешанное число A в позиционной системе счисления (СС) с основанием q можно записать по формуле

$$A_q = a_n q^n + a_{n-1} q^{n-1} + \ldots + a_1 q^1 + a_0 q^0 + a_{-1} q^{-1} + \ldots + a_{-m} q^{-m} \quad , \qquad (1)$$

где a_n– цифра числа в данной СС;
q^n – разрядный вес цифры ak;
n+1 – количество разрядов в целой части числа;
m- количество разрядов в дробной части числа.

1.1.1 Для перевода целого числа A из СС с основанием p в СС сосновании q нужно воспользоваться формулой

$$A_p = a_n q^n + a_{n-1} q^{n-1} + \ldots\ldots + a_1 q^1 + a_0 . \qquad (2)$$

Необходимо разделить обе части формулы (2) на новое основание q

$$\frac{A_p}{q} = a_n q^{n-1} + a_{n-1} q^{n-2} + \ldots\ldots + a_1 + \frac{a_0}{q} \qquad (3)$$

.

В правой части формулы (3) сформировалась целая часть первого частного и первый остаток от деления a_0 - младшая цифра целого числа в новой СС. Далее целую часть первого частного следует разделить на основание новой СС q , и новый остаток даст вторую искомую цифру a_1 и т.д. Это позволяет сформулировать правило.

Чтобы перевести целое число в новую СС, его надо последовательно делить на основание новой СС до тех пор , пока не получится частное, у которого целая часть равна «0». Число в новой СС записывают из остатков от последовательного деления, причем последний остаток будет старшей цифрой целого числа в новой СС.

1.1.2 Для перевода дробного числа A из СС с основанием p в СС с основанием q нужно воспользоваться формулой

$$A_p = a_1 q^{-1} + a_2 q^{-2} + ... + a_m q^{-m} \quad (4)$$

Необходимо обе части формулы (4) умножить на новое основание q

$$A_p \times q = a_1 + a_2 q^{-1} + ... + a_m q^{-m+1} \quad (5)$$

В правой части равенства (5) a_1 - целая часть первого произведения, являющаяся старшей цифрой дроби в новой СС. Далее, умножением на новое основание q дробной части первого произведения, определяется вторая цифра дроби - a_2 ,как целая часть второго произведения и т.д. Отсюда следует правило.

Чтобы перевести правильную дробь из одной позиционной СС в другую, её надо последовательно умножать на основание новой СС до тех пор, пока в новой дроби не будет получено требуемого количества цифр, определяемого заданной точностью. Правильная дробь в новой СС записывается из целых частей произведений, и целая часть первого произведения будет старшей цифрой новой дроби.

Перевод дробей - бесконечный процесс и может быть выполнен только приближенно. Для сохранения точности исходной дроби определяется количество цифр в изображении дроби по новому основанию q по формуле

$$m_2 = \frac{m_1}{\log_p q} \cong \left] \frac{m_1}{\log_p q} \right[+ 1 \quad (6)$$

,

где m1 – количество цифр в исходной дроби с основанием p;

m2 – количество цифр в дроби с новым основанием q.

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

1.1.3 Использование вспомогательных систем счисления позволяет ускорить процесс перевода чисел. Чаще всего используют восьмеричную (8СС) и шестнадцатеричную (16СС) системы счисления.

Правила перевода чисел из 10СС в 2СС и обратно с использованием в качестве вспомогательных восьмеричную и шестнадцатеричную системы счисления формулируются следующим образом.

Чтобы перевести число из 10СС в 2СС с использованием 8СС или 16СС, надо перевести десятичное число в 8СС или 16СС указанными выше способами, а затем представить цифры восьмеричного (шестнадцатеричного) числа триадами (тетрадами).

Обратный перевод чисел из 2СС в 10СС с использованием вспомогательных СС выполняется по следующему правилу.

Вправо и влево от запятой двоичное число разбивается на триады (тетрады), которые заменяются соответствующими восьмеричными (шестнадцатеричными) цифрами. Далее по основной формуле переходят к 10СС. Причем, если в крайних триадах (тетрадах) недостаточно разрядов, то они дополняются «0»: старшие разряды – слева, младшие – справа.

1.1.4 Для двоичных чисел с ФЗ используют три формата фиксированной длины: полуслово – короткий с ФЗ(16 разрядов); слово – длинный с ФЗ(32 разряда); двойное слово – для промежуточных действий, чтобы обеспечить высокую точность вычислений(64 разряда). Двоичные операнды имеют вид целых чисел в дополнительном коде, у которых крайний левый разряд – знаковый. Это правило справедливо как для ЕС ЭВМ так и для ПЭВМ.

1.1.5 Двоичные числа с ПЗ изображаются по-разному в ЕС и ПЭВМ. В ПЭВМ смещенный порядок занимает восемь разрядов (смещение равно 127), крайний левый разряд сетки занимает знак числа, остальные разряды отводятся под мантиссу, изображаемую в 2СС (23 разряда в коротком формате). Смещенный порядок содержит информацию о положении запятой в двоичной мантиссе числа.

В ЕС ЭВМ смещенный порядок занимает семь разрядов (смещение равно 64) и размещается в старшем байте вместе со знаковым разрядом числа. Остальные разряды (24 для короткого формата) занимает мантисса числа, изображаемая в 16СС. Каждые 4 бита воспринимаются машиной как одна 16-ричная цифра, а в смещенном порядке содержится информация о положении запятой между 16-ричными, а не двоичными цифрами. Мантисса изображается в ПК и должна быть нормализована.

1.2 Перевод чисел из десятичной системы счисления в двоичную

Необходимо осуществить перевод чисел А = 333,81 и В = 803,35 из 10СС в 2СС через промежуточные системы счисления. Число А нужно перевести с использованием восьмеричной системы счисления, а число В – шестнадцатеричной, выполнить проверку. Данные действия показаны на рисунках 1 – 2.

A: 10СС → 8СС → 2СС → 16СС → 10СС;

A: 10СС →8СС:

$333{,}81_{10} \to 515{,}636_8$

$m2= \left]\frac{2}{\log_{10} 8}\right[+1=3$

$$\begin{array}{r|l} 333 & 8 \\ \hline 328 & 41 \;|\; 8 \\ \mathbf{5} & 40 \;|\; \mathbf{5} \\ & \mathbf{1} \end{array}$$

0,81
8
6,48
8
3,84
8
6,72

8СС → 2СС: $515{,}636_8 \to 101001101{,}110011110_2$

2СС →16СС: $101001101{,}110011110_2 = 14D{,}CF5_{16}$

16СС →10СС: $14D{,}CF5_{16} = 1*16^2 + 4*16^1 + 13*16^0 +12*16^{-1} +15*16^{-2}+5*16^{-3}=$
$256 + 64 + 13 + 0.75 + 0{,}0586 + 0{,}0012 = 333.8098_{10}$

Ответ: $333{,}81_{10} \to 515{,}636_8 \to 101001101{,}110011110_2 \to 14D{,}CF5_{16} \to 333.8098_{10}$

Рисунок 1 – Перевод числа А через восьмеричную систему счисления

B: 10СС → 16СС → 2СС → 8СС → 10СС.

m2=2

B: 10СС →16СС: $803{,}35_{10} \to 323{,}5(9)_{16}$

$$\begin{array}{r|l} 803 & 16 \\ \hline 800 & 50 \;|\; 16 \\ \mathbf{3} & 48 \;|\; \mathbf{3} \\ & \mathbf{2} \end{array}$$

0,35 0,60
16 16
5,60 **9,60**
16
9,60

16СС → 2СС: $323{,}5(9)_{16} \to 1100100011.010110011001_2$

2СС →8СС: $1100100011.010110011001_2 = 1443.2631_8$

8СС →10СС: $1443.2631_8 = 1*8^3 + 4*8^2 + 4*8^1 +3*8^0 +2*8^{-1} +6*8^{-2}+3*8^{-3}+8^{-4} =$
$512 + 256 + 32 + 3 + 0{,}25+0{,}0937+0{,}0069+0{,}0002= 803.3498_{10}$

Ответ :$803{,}35_{10} \to 323{,}5(9)_{16} \to 1100100011.010110011001_2 \to 1443.2631_8 \to 803.3498_{10}$

Рисунок 2 – Перевод числа B через шестнадцатеричную систему счисления

1.3 Изображение чисел в форме с фиксированной запятой

Необходимо изобразить числа А и В в форме с фиксированной запятой (ФЗ) в 32-разрядной сетке ЭВМ, указав масштаб операндов. При чем число А - положительное, число В - отрицательное. Представление операндов показано на рисунке 3.

$A = 333{,}81_{(10)} = 101001101.11001111_2$

$B = -803{,}35_{(10)} = -1100100011.01011001_2$

Представим число А в прямом коде:

$A_{ПК} = 0.1001110101010010_2$

Так как А– положительное, то его дополнительный код совпадает с прямым кодом.

Представим число А в длинном формате: $M = 2^{-8}$

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

Представим число В в прямом коде:

$B_{ПК} = 1.110010001101011001_2$

Запишем число В в дополнительном коде:

$B_{ДК} = 1.\ 001101110010100111_2$

Представим число В в длинном формате: $M = 2^{-8}$

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

Рисунок 3 - Изображение чисел в форме с фиксированной запятой

1.4 Изображение чисел в форме с плавающей запятой

Необходимо изобразить числа А и В в форме с плавающей запятой (ПЗ) в 32-разрядной сетке ЭВМ, представив мантиссу в 2СС (ПЭВМ) и 16СС (ЕС ЭВМ) и отведя соответственно под смещенные порядки (характеристики) восемь разрядов (ПЭВМ) и семь (ЕС ЭВМ). Причем, А – отрицательное, В - положительное. Представление операндов изображено на рисунке 4.

$A = -333{,}81_{(10)} = -101001101.11001111_2$

$B = 803{,}35_{(10)} = 1100100011.01011001_2$

ЕСЭВМ:

Для А (-101001101.11001111_2)

Зн	Характеристика							Мантисса																							
1	1	0	0	0	0	1	1	0	0	0	1	0	1	0	0	1	1	0	1	1	1	0	0	1	1	1	1	0	0	0	0

Для В (1100100011.01011001_2)

Зн	Характеристика							Мантисса																							
0	1	0	0	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	0	1	0	1	1	0	0	1	0	0	0	0

ПЭВМ:

Для А (-101001101.11001111_2)

Зн	Характеристика								Мантисса																						
1	1	0	0	0	1	0	0	1	1	0	1	0	0	1	1	0	1	1	1	0	0	1	1	1	1	0	0	0	0	0	0

Для В (1100100011.01011001_2)

Зн	Характеристика								Мантисса																						
0	1	0	0	0	1	0	1	0	1	1	0	0	1	0	0	0	1	1	0	1	0	1	1	0	0	1	0	0	0	0	0

Рисунок 4 - Изображение чисел в форме с плавающей запятой

2. Сложение в различных кодах и формах представления

2.1 Теоретические сведения о сложении двоичных чисел

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.

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

Сложение чисел в форме с ФЗ в ДК: при алгебраическом сложении чисел в ДК результат получают также в ДК, а при возникновении единицы переноса из знакового разряда ее отбрасывают.

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

$$A = \pm m_a \cdot 2^{\pm p_a} \;\; B = \pm m_b \cdot 2^{\pm p_b}, \qquad (3)$$

где A,B – исходные операнды;
m – мантисса числа;
p – порядок числа.

Чтобы сложить их, надо выполнить различные действия над мантиссами и порядками:

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.
2) Сложение мантисс операндов по правилам сложения чисел с ФЗ.
3) При необходимости, нормализация результата. При этом денормалиэация вправо, т.е. ситуация, когда в старшем разряде двоичной мантиссы «О», 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 возникновении истинного ПРС числа с ПЗ, хотя вероятность этого невелика.

Смещенные порядкииспользуются 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», «-» -«О». В результате порядки чисел увеличиваются (в ЕСЭВМ на 2^6=64, в СМЭВМ 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 В первом пункте необходимо сложить исходные операнды с ФЗ в ОК. Знаки операндов: A>0, B<0. Действие изображено на рисунке 5

2.2.2 Во втором пункте необходимо сложить исходные операнды с ФЗ в ДК. A<0,B>0. Действие изображено на рисунке 6.

$A = 333{,}81_{(10)} = 101001101.1100111_2$
$B = -803{,}35_{(10)} = -1100100011.010110_2$
Масштаб: $M = 2^{10}$
$A_{ПК} = 0{,}0101001101110 0111$
$B_{ПК} = 1{,}11001000110101100$

Сложение:

$A_{ОК}$=	0	,	0	1	0	1	0	0	1	1	0	1	1	1	0	0	1	1	1
$B_{ОК}$=	1	,	0	0	1	1	0	1	1	1	0	0	1	0	1	0	0	1	1
$(A+B)_{ОК}$=	1	,	1	0	0	0	1	0	1	0	1	0	0	1	1	1	0	1	0
$(A+B)_{ПК}$=	1	,	0	1	1	1	0	1	0	1	0	1	1	0	0	0	1	0	1
(A+B)=	$-0111010101{,}1000101 = (-469.5390625)_{10}$																		

Проверка: 333,81 – 803,35 = -469.54

Рисунок 5 - Сложение с ФЗ в ОК

$A = -333{,}81_{(10)} = -101001101.1100111_2$
$B = 803{,}35_{(10)} = 1100100011.0101100_2$
Масштаб: $M = 2^{10}$
$A_{ПК} = 1{,}010100110111 00111$
$B_{ПК} = 0{,}1100100011.0101100$

Сложение:

$A_{ДК}$=	1	,	1	0	1	0	1	1	0	0	1	0	0	0	1	1	0	0	1
$B_{ДК}$=	0	,	1	1	0	0	1	0	0	0	1	1	0	1	0	1	1	0	0
$(A+B)_{ДК}$=	10	,	0	1	1	1	0	1	0	1	0	1	1	0	0	0	1	0	1
$(A+B)_{ПК=}$	0	,	0	1	1	1	0	1	0	1	0	1	1	0	0	0	1	0	1
$(A+B)_{=}$	$111010101{,}1000101=(469.539)_{10}$																		

Проверка: 803,35 – 333,81 =469.539

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 кода выберем МДК. Данное действие изображено на рисунке 7.

$A = -333{,}81_{(10)} = -101001101.1100111_2$

$B = -803{,}35_{(10)} = -1100100011.0101100_2$

Масштаб: $M = 2^{10}$

$A_{ПК} = 1{,}01010011011100111$

$B_{ПК} = 1{,}1100100011.0101100$

$A_{МДК}$=	11	.	1	0	1	0	1	1	0	0	1	0	0	0	1	1	0	0	1
$B_{МДК}$=	11	.	0	0	1	1	0	1	1	1	0	0	1	0	1	0	1	0	0
$(A+B)_{МДК}$=	**10**	.	1	1	1	0	0	0	1	1	1	0	1	1	0	1	1	0	1

10 - запрещенная комбинация, выбираем другой масштаб слагаемых:

$M = 2^{11}$

$A_{МДК}$=	11	.	1	1	0	1	0	1	1	0	0	1	0	0	0	1	1	0	0	1
$B_{МДК}$=	11	.	1	0	0	1	1	0	1	1	1	0	0	1	0	1	0	1	0	0
$(A+B)_{МДК}$=	**11**	.	0	1	1	1	0	0	0	1	1	1	0	1	1	0	1	1	0	1
$(A+B)_{ПК=}$	1	.	1	0	0	0	1	1	1	0	0	0	1	0	0	1	0	0	1	1
$(A+B)_{=}$	1,10001110001001001 1																			

$(A+B) = -10001110001{,}0010011_2 = 1137{,}15$

Проверка: -333,81-803,35=1137,15

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.

Под мантиссы со знаком отведем 20 разрядов, под порядки со знаком - 5 разрядов.

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

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

P_A= 1001 $P_{A\ ПК}$=0,1001 $P_{A\ ДК}$=0,1001
P_B= 1010 $P_{B\ ПК}$=1,1010 $P_{B\ ДК}$=1,0110

0,1001
<u>1,0110</u> $(P_A\text{-}P_B)_{ПК}=(1{,}0001)_2=-1_{10}$
1,1111

Мантиссу числа A необходимо сдвинуть на 1 разряд вправо(денормализовать) и увеличить порядок на 1:

A	**0**	0	1	0	1	0	0	1	1	0	1	1	1	0	0	1	1	1	1	0	**0**	1	0	1	0

Складываем мантиссы в ДК:

m_A 0,01010011011100111110
m_B <u>0,11001000110101100 11</u>
m_A+m_B 1,00011100010010100 01

Временное ПРС суммы мантисс. Необходимо мантиссу сдвинуть на один разряд вправо, с одновременным увеличением порядка на 1.

	Мантисса																				Порядок				
A+B	**0**	1	0	0	0	1	1	1	0	0	0	1	0	0	1	0	1	0	0	0	**0**	1	0	1	1

Проверка: 10001110001,001001 = $1137{,}16_{10}$
333,81+803,35 = 1137,16

Рисунок 8 - Сложение в формате с плавающей запятой

3. Умножение двоичных чисел

3.1 Теоретическая часть

В цикле умножения двоичных чисел три элементарных операции:

1) Анализ цифры очередного разряда множителя;
2) Суммирование множимого с накопленной суммой частичных произведений, если цифра множителя «1»;
3) 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 4 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.

3.1.1 I способ – умножение с младших разрядов множителя со сдвигом суммы частичных произведений вправо. Устройства, которые хранят операнды, регистры, имеют следующую разрядность:

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

Суммирование множимого следует выполнять в старших n разрядах 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 (ПРС) в регистре суммы частичных произведений, которое ликвидируется при очередном сдвиге вправо.

3.1.2 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.

3.1.3 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. 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.

3.1.4 IV способ – умножение со старших разрядов множителя со сдвигом множимого вправо. Этот способ требует одного n-разрядного регистра множителя и двух 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 (ПК), так и в алгоритмах умножения в дополнительном коде(ДК).

3.1.5 Умножение двоичных чисел в дополнительном коде.

Алгоритм умножения двоичных чисел 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 коррекцию по следующим правилам:
   a) если один сомножитель отрицателен, к псевдопроизведению прибавляется дополнительный код от модуля положительного сомножителя;
   b) 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.

3.1.6 Умножение двоичных чисел в ДК с автоматической коррекцией.

Этот алгоритм разработан Бутом и является универсальным для умножения чисел в ДК. Сомножители участвуют в операции со знаковыми разрядами, которые рассматриваются как цифровые разряды числа. Результат получается сразу в дополнительном коде со знаком.

В процессе умножения анализируются две смежные цифры множителя: та, на которую выполняется умножение в данном такте – m1, и соседняя младшая цифра - m2. В двоичном множителе этой паре соответствуют четыре возможных набора - «00», «01», «10», «11», каждый из которых требует выполнения следующих действий:

1) набор «01» требует сложения множимого с предыдущей суммой частичных произведений;
2) набор «10» требует вычитания множимого из предыдущей суммы частичных произведений;
3) наборы «00» и «11» не требуют ни сложения, ни вычитания, так как частичное произведение равно нулю.

В цикле умножения в каждом такте выполняются соответствующие сдвиги на один разряд. При этом могут использоваться все четыре способа умножения с некоторыми особенностями:

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

3.2 Умножение чисел в форме с фиксированной запятой I способом

Необходимо выполнить умножение чисел C=88 (множимое) и D=-41 (множитель). Данные действия показаны на рисунке 9.

$C = 88_{10} = 1011000_2 D = -41_{10} = -101001_2 (M = 2^7)$

$C_{ПК} = 0.1011000$ $D_{ПК} = 1.0101001$

1) Знак произведения: $0 \oplus 1 = 1$;

2) Перемножение модулей

Множитель→	СЧП →	Примечание
*,0101001	0,0000000 0000000 0,1011000 0000000 **0,1011000 0000000**	+ →
*,*010100	**0,0101100 0000000**	→
*,**01010	**0,0010110 0000000**	→
*,***0101	0,0001011 0000000 0,1011000 0000000 **0,1100011 0000000**	+ →
*,****010	**0,0110001 1000000**	→
*,*****01	0,0011000 1100000 0,1011000 0000000 **0,1110000 1100000**	+ →
*,******0	**0,0111000 0110000**	→
*,*******	**0,0011100 0011000**	→

$C*D_{ПК} = 1,00111000011000$; $M_{(C*D)} = 2^7*2^7 = 2^{14}$

Проверка: $C*D = -0,00111000011000_2$;$= -3608_{10}$

Рисунок 9 - Умножение чисел в форме с фиксированной запятой I способом

3.3 Умножение чисел в форме с фиксированной запятой II способом с автоматической коррекцией.

Необходимо выполнить умножение чисел C=-88 (множитель) и D=41 (множимое)с ФЗ в дополнительном коде II способом, используя алгоритм с автоматической коррекцией. Проверить результат операции. Данные действия показаны на рисунке 10.

$C = -88_{10} = -1011000_2$ $C_{ПК}=1,1011000$ $C_{ДК}=1,0101000$
$D=41_{10}=101001_2$ $D_{ПК}=0,1010010$ $D_{ДК}=0,1010010$

Множитель	Множимое	Сумма ЧП	Комментарий
1,0101000 0	0,00000001010010	0,0000000 0000000	Сдвиг
1,1010100 0	0,00000010100100	0,0000000 0000000	Сдвиг
1,1101010 0	0,00000101001000	0,0000000 0000000	Сдвиг
1,1110101 0	0,00001010010000	0,0000000 0000000 1,1111010 1110000 1,1111010 1110000	Сдвиг Вычитание
1,1111010 1	0,00010100100000	1,1111010 1110000 0,0001010 0100000 0,0000101 0010000	Сдвиг Сложение
1,1111101 0	0,00101001000000	0,0000101 0010000 1,1101011 1000000 1,1110000 1010000	Сдвиг Вычитание
1,111110 1	0,01010010000000	1,1110000 1010000 0,0101001 0000000 0,0011001 1010000	Сдвиг Сложение
1,11111 0	0,10100100000000	0,0011001 1010000 1,0101110 0000000 1,1000111 1010000	Сдвиг Вычитание

$(A*B)_{ДК}=1,10001111010000$
$(A*B)_{ПК}= 1,01110000110000$; $M_{(C*D)} = 2^7 * 2^7 = 2^{14}$
Проверка: A*B= - 111000011000= - 3608

Рисунок 10 - Умножение чисел в форме с фиксированной запятой II способом с автоматической коррекцией

3.4 Умножение чисел в форме с фиксированной запятой III способом с простой коррекцией

Необходимо выполнить умножение чисел C=-88 (множитель) и D=-41 (множимое)с ФЗ в дополнительном коде III способом, используя алгоритм с простой коррекцией. Проверить результат операции. Данные действия показаны на рисунке 11.

$C = -88_{10} = -1011000_2$ $C_{ПК}=1{,}1011000$ $C_{ДК}=1{,}0101000$

$D=-41_{10}=101001_2$ $D_{ПК}=1{,}101001$ $D_{ДК}=1{,}1010111$

1) Знак произведения: $1 \oplus 1 = 0$;

2) Перемножение модулей

Множитель	Множимое	Сумма ЧП	Комментарий
0101000	0,1010111	0,0000000 0000000	Сдвиг
101000*		0,0000000 0000000 0,0000000 1010111 0,0000000 1010111	Сдвиг Сложение
01000**		0,0000001 0101110	Сдвиг
1000***		0,0000010 1011100 0,0000000 1010111 0,0000011 0110011	Сдвиг Сложение
000****		0,0000110 1100110	Сдвиг
00*****		0,00011011001100	Сдвиг
0******		0,00110110011000	Сдвиг

Получено псевдопроизведение: 0,0011011 0011000

Коррекция (оба сомножителя отрицательны)

0,0011011 0011000
0,1011000 0000000
0,1110011 0011000
0,0101001 0000000
0,0011100 0011000

$(C*D)_{ДК} = (C*D)_{ПК} = 0{,}0011100\ 0011000_2$ $M_{(C*D)} = 2^7 * 2^7 = 2^{14}$

Проверка: $C*D = 11100\ 0011000_2 = 3608_{10}$

Рисунок 11 - Умножение чисел в форме с фиксированной запятой III способом с простой коррекцией

3.5 Умножение чисел в форме с плавающей запятой IV способом

Необходимо выполнить умножение чисел C = 88 и D = 41 с плавающей запятой. При умножении мантисс использовать IV способ. Проверить результат операции и изобразить его в разрядной сетке условной машины. Данные действия показаны на рисунке 12.

$C = 88_{10} = 1011000_2$
$D = 41_{10} = 101001_2$

Разрядная сетка:

	Мантисса		Порядок	
D	0	1011000	0	0110
C	0	1010010	0	0111

1. Определение знака произведения: $0 \oplus 0 = 0$
2. Определение порядка произведения:
   $p_d =$ 0110
   $p_c =$ 0111
   $p_{(c*d)} =$ 1101
3. Перемножение модулей мантисс 4 способом:

Множитель	Множимое	Сумма ЧП	Комментарий
0,1011000	0,01010010000000	0,0000000 0000000 0,0101001 0000000 0,0101001 0000000	Сдвиг Сложение
0,011000*	0,00101001000000	0,0101001 0000000	Сдвиг
0,11000**	0,00010100100000	0,0101001 0000000 0,0001010 0100000 0,0110011 0100000	Сдвиг Сложение
0,1000***	0,00001010010000	0,0110011 0100000 0,0000101 0010000 0,0111000 0110000	Сдвиг Сложение
0,000****	0,00000101001000	0,0111000 0110000	Сдвиг
0,00*****	0,00000010100100	0,0111000 0110000	Сдвиг
0,0******	0,00000001010010	0,0111000 0110000	Сдвиг

4. Нормализация мантиссы:

Мантисса денормализована, сдвигаетсяна1 разряд влево, значит необходимо уменьшить порядок на 1.

Результат:

C*D	0	1110000	0	1100

Проверка: $C*D = 111000000000_2 = 3584_{10}$
Погрешность округления = 24

Рисунок 12 - Умножение чисел в форме с плавающей запятой IV способом

4. Деление двоичных чисел

4.1 Теоретическая часть

4.1.1 Алгоритм деления с восстановлением остатка с использованием обратного кода при вычитании:

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.

3) Проанализировать знак остатка после первого вычитания:

а). Если остаток положительный, то произошло ПРС! Операцию следует прекратить для смены масштабов операндов;

б). Если остаток отрицательный, то в частное заносится 0 (после он будет в знаковом разряде), восстановить остаток, прибавив к нему делитель.

4) Выполнить сдвиги.

5) В цикле формирования цифр частного вычитать из остатка делитель, прибавляя его в ОК или ДК.

6) Проанализировать знак полученного остатка:

а). Если >0, то в частное заносится 1;

б). Если <0, то в частное заносится 0.

7) Восстановить отрицательный остаток, сложив его с делителем.

8) Выполнить сдвиги.

9) Сформировать (n+1)-ый остаток для округления частного.

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

4.1.2 Алгоритм деления без восстановления остатков с использованием дополнительного кода при вычитании:

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:

а) если положителен, то произошло ПРС, операцию следует прекратить для смены масштаба операндов;

б) 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

4.1.3Алгоритм деления чисел в дополнительном коде(с автоматической коррекцией):

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.

4.1.4 Алгоритм деления числе в форме с плавающей запятой:

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.

4.2 Деление чисел в форме с фиксированной запятой I способом с восстановлением остатков.

Необходимо выполнить деление чисел C = 88 и D = -41с ФЗ I способом (с восстановлением остатка с использованием обратного кода при вычитании). Проверить результат операции, оценить погрешность округления. Данные действия показаны на рисунке 13.

$C = 88_{10} = 1011000_2$; $C_{ПК} = 0.1011000$ – делимое

$D = -41_{10} = -101001_2$; $D_{ПК} = 1.\ 0101001$ – делитель

Масштаб: $M = 2^7$

1) Знак частного: $0 \oplus 1 = 1$;
2) Деление модулей – 1 способ:

Частное	Делимое	Комментарий
0,0000000	0,1011000 1,1010110 0,0101110	ПРС!
0,0000000	0,0101100 1,1010110 0,0000010	ПРС!
0,000000**0**	0,0010110 1,1010110 1,1101100 0,0101001 0,0010110	Вычитание Первый остаток Восстановление
0,00000**01**	0,0101100 1,1010110 0,0000010	Сдвиги Вычитание Второй остаток
0,0000**010**	0,0000100 1,1010110 1,1011010 0,0101001 0,0000100	Сдвиги Вычитание Третий остаток Восстановление
0,000**0100**	0,0001000 1,1010110 1,1011110 0,0101001 0,0001000	Сдвиги Вычитание Четвертый остаток Восстановление
0,00**01000**	0,0010000 1,1010110 1,1100110 0,0101001 0,0010000	Сдвиги Вычитание Пятый остаток Восстановление
0,0**010000**	0,0100000 1,1010110 1,1110110 0,0101001 0,0100000	Сдвиги Вычитание Шестой остаток Восстановление
0,**0100001**	0,1000000	Сдвиги

	1,1010110 0,0010111	Вычитание Седьмой остаток
0,1000011	0,0101110 1,1010110 0,0000100	Сдвиги Вычитание Восьмой остаток
0,1000011 **(0)**	0,0001000 1,1010110 1,1011110	Сдвиги Вычитание Девятый остаток (для округления)

$(C/D)_{ПК}$ = -10,00011= $-2,09375_{10}$

Проверка: 88/(-41) ≈ -2,1463

Абсолютная погрешность:

Δ = |-2,1463-(-2,09375)| =0,05255

Относительная погрешность:

δ≈(|-2,1463-(-2,09375)| /|-2,1463|)* 100% = 2,45%

Рисунок 13 - Деление чисел в форме с фиксированной запятой I способом с восстановлением остатков

4.3 Деление чисел в форме с фиксированной запятой II способом без восстановления остатков.

Необходимо выполнить деление чисел C = -88 и D = -41 с ФЗ II способом. Проверить результат операции, оценить погрешность округления. Данные действия показаны на рисунке 14.

$C = -88_{10} = -1011000_2$; $C_{ПК}$ =1,1011000 – делимое

$D = -41_{10} = -101001_2$;$D_{ПК}$ =1,0101001 – делитель

Масштаб: $M=2^7$

1) Знак произведения: $1 \oplus 1 = 0$;
2) Деление модулей – 2 способ:

Частное	Делитель	Делимое	Комментарий
0,0000000	0,0101001 0000000	0,1011000 0000000 1,1010111 0000000 0,0101111 0000000	ПРС! $M = 2^8$
0,0000000	0,0101001 0000000	0,0101100 0000000 1,1010111 0000000 0,0000011 0000000	ПРС! $M = 2^9$
0,000000**0**	0,0101001 0000000	0,0010110 0000000 1,1010111 0000000 1,1101101 0000000	Вычитание Первый остаток
0,00000**01**	0,0010100 1000000	1,1101101 0000000 0,0010100 1000000 0,0000001 1000000	Сдвиги Сложение Второй остаток
0,0000**010**	0,0001010 0100000	0,0000001 1000000 1,11101011100000	Сдвиги Вычитание

		1,11101110100000	Третий остаток
0,0000**0100**	0,0000101 0010000	1,1110111 0100000 0,0000101 0010000 1,1111100 0110000	Сдвиги Сложение Третий остаток
0,00**01000**	0,0000010 1001000	1,1111100 0110000 0,0000010 1001000 1,1111110 1111000	Сдвиги Сложение Четвертый остаток
0,0**010001**	0,0000001 0100100	1,1111110 1111000 0,0000001 0100100 0,0000000 0011100	Сдвиги Сложение Пятый остаток
0,**0100010**	0,0000000 1010010	0,0000000 0011100 1,1111111 0101110 1,1111111 1001010	Сдвиги Сложение Шестой остаток
0,1000100	0,0000000 0101001	1,1111111 1001010 0,0000000 0101001 1,1111111 1110011	Сдвиги Сложение Седьмой остаток
0,1000100(1)	0,0000000 0010100	1,1111111 1110011 0,0000000 0010100 0,0000000 0000111	Сдвиги Сложение Восьмой остаток (для округления)

$(C/D)_{ПК} = 10,00101_2 = 2.15625_{10}$

Проверка: 88/(41) ≈2,1463

Абсолютная погрешность:

$\Delta = |2,1463-2.15625| = 0,00995$

Относительная погрешность:

$\delta \approx (|2,1463-2.15625| / |2.15625|) * 100\% = 0.46\%$

Рисунок 14 - Деление чисел в форме с фиксированной запятой II способом без восстановления остатков

4.4 Деление чисел в форме с фиксированной запятой в ДК II способом с автоматической коррекцией.

Необходимо выполнить деление чисел D=-41 и C=88с ФЗ в ДК II способом. D – делимое. Проверить результат операции, оценить погрешность округления. Данные действия показаны на рисунке 15.

$C = -88_{10} = -1011000_2$;$C_{ПК} = 1,1011000$ $C_{ДК} = 1,0101000$

$D = 41_{10} = 101001_2$;$D_{ПК} = 0,0101001$

Масштаб: $M=2^7$

1) Знак частного: $1 \oplus 0 = 1$;
2) Деление модулей – 2 способ:

Частное	Делитель	Делимое	Комментарий
0,0000000	0,01010010000000	1,01010000000000	Сравнение знаков

		0,01010010000000 1,1010001 0000000	Сложение ПРС! M = 2^8
0,0000000	0,01010010000000	1,10101000000000 0,01010010000000 1,1111101 0000000	Сдвиги Сложение ПРС! M = 2^8
0,000000**1**	0,01010010000000	1,11010100000000 0,01010010000000 0,0010011 0000000	Сдвиги Сложение
0,00000**10**	0,00101001000000	0,0010011 0000000 1,11010111000000 1,1111110 1000000	Сдвиги Вычитание
0,0000**101**	0,00010100100000	1,1111110 1000000 0,00010100100000 0,0001000 1100000	Сдвиги Сложение
0,000**1011**	0,00001010010000	0,0001000 1100000 1,1111010 1110000 0,0000011 1010000	Сдвиги Вычитание
0,00**10111**	0,00000101001000	0,0000011 1010000 1,1111101 0111000 0,0000001 0001000	Сдвиги Вычитание
0,0**101110**	0,00000010100100	0,0000001 0001000 1,11111101011100 1,11111111100100	Сдвиги Вычитание
0,**1011101**	0,00000001010010	1,1111111 1100100 0,0000000 1010010 0,00000000110110	Сдвиги Сложение
1,0111011	0,0000000 0101001	0,0000000 0110110 1,1111111 1010111 0,00000000001101	Сдвиги Вычитание
1,0111011(0)	0,0000000 0010100	0,0000000 0001101 1,1111111 1101100 1,1111111 1111001	Сдвиги Вычитание (Для округления)

$(D/C)_{ДК}$ =1,0111011(0)

$(D/C)_{ПК}$ =1,1000101(0)

$M=2^9/2^7=2^2$

(D/C)= -10,00101= -2.15625_{10}

Проверка: -88/(41)≈-2,1463

Абсолютная погрешность:

Δ = |-2.15625-(-2,1463)| = 0,00995

Относительная погрешность:

δ≈ (|-2.15625-(-2,1463)| /|-2.15625|)* 100% = 0.46%

Рисунок 15 - Деление чисел в форме с фиксированной запятой в ДК II способом с автоматической коррекцией

4.5 Деление чисел в форме с плавающей запятой I способом

Необходимо выполнить деление чисел C = 88 и D = 41 в форме с ПЗ. При делении мантисс использовать I способ. D – делимое. Изобразить частное в разрядной сетке условной машины, проверить результат операции, оценить погрешность округления. Данные действия показаны на рисунке 16,17.

$C = 88_{10} = 1011000_2$; $C_{ПК}$ =0.1011000 –делитель
$D = 41_{10} = 101001_2$;$D_{ПК}$ =0.0101001 – делимое

	Знак числа	Мантисса	Знак порядка	Порядок
C	0	1011000	0	0111
D	0	1010010	0	0110

Порядок частного: 0,0110
1,1001
1,1111

Рисунок 16 – Представление чисел в разрядной сетке

1) Знак частного: 0 ⊕ 0 = 0;
2) Деление 1 способом с восстановлением остатков с использованием дополнительного кода при вычитании:

Частное	Делимое	Комментарий
0,000000**0**	0,1010010 1.0101000 1,1111010 0.1011000 0,1010010	Вычитание Первый остаток Восстановление
0,00000**01**	1,0100100 1,0101000 0,1001100	Сдвиги Вычитание Второй остаток
0,0000**011**	1,0011000 1,0101000 0,1000000	Сдвиги Вычитание Третий остаток
0,000**0111**	1,0000000 1,0101000 0,0101000	Сдвиги Вычитание Четвертый остаток
0,00**01110**	0,1010000 1,0101000 1,1111000 0.1011000 0,1010000	Сдвиги Вычитание Пятый остаток Восстановление
0,0**011101**	1,0100000 1,0101000	Сдвиги Вычитание

	0,1001000	Шестой остаток
0,**<u>0111011</u>**	1,0010000 <u>1,0101000</u> 0,0111000	Сдвиги Вычитание Седьмой остаток
<u>0,1110111</u>	0,1110000 <u>1,0101000</u> 0,0011000 <u>0,1011000</u> 0,1001000	Сдвиги Вычитание Восьмой остаток
<u>0,1110111</u>(1)	1,0010000 <u>1,0101000</u> 0,0101000	Сдвиги Вычитание Девятый остаток (для округления)

	Знак числа	Мантисса	Знак порядка	Порядок
C/D	0	1111000	1	0001

$(C/D)_{ПК}$=0,0111100=0.46875 Абсолютная погрешность:

Проверка: 41/88≈0,4659 Δ = |0,46875-0.4659| = 0,00285

Относительная погрешность:

δ≈(|0,46875-0.4659|/|0.46875|)*100% = 0,6%

Рисунок 17 - Деление чисел в форме с плавающей запятой I способом

5. Умножение двоичных чисел ускоренным методом второго порядка

5.1 Теоретическая часть

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

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

В процесс умножения легко получить удвоенное множимое при помощи сдвига. Утроенное множимое требует предварительных вычислений, но этого можно избежать – 3=4-1, что в двоичном представлении

$$(11)_2=(100)_2-1.$$

Правило 1.1. На текущем шаге умножения вместо сложения с утроенным множимым можно выполнить вычитание множимого и учесть единицу переноса p_{i+1} в старшую пару на следующем шаге.

Для способов умножения, в которых выполняется анализ младших разрядов множителя составим таблицу:

a_{2i+1}	a_{2i}	p_i	Сумма ЧП	p_{i+1}
0	0	0	+0	0
0	1	0	+M	0
1	0	0	+2M	0
1	1	0	-M	1
0	0	1	+M	0
0	1	1	+2M	0
1	0	1	-M	1
1	1	1	+0	1

Для способов, в которых выполняется анализ старших разрядов множителя, правило 1.1 требует поправок в силу того, что распространять перенос некуда. Но можно обойтись без распространения переноса, анализируя разряды младшей пары, находящейся справа от текущей. При этом существует неопределенность возникновения переноса из младшей пары (a_{2i-1}, a_{2i-2}):

a_{2i-2}	a_{2i-1}	Перенос
0	0	0
0	1	0
1	0	Возможно
1	1	1

Действительно, если младшая пара 10 и в нее не будет переноса, то из нее также не будет переноса, но если перенос будет, то уходя от получившейся комбинации 11, из нее будет сгенерирован перенос. Тогда уйдем от комбинации 10:

$$(10)_2=(100)_2-(10)_2$$

В таком случае перенос будет всегда.

Правило 1.2. На текущем шаге умножения вместо сложения с удвоенным множимым (комбинация 10), можно выполнить вычитание удвоенного множимого (-2М).

При этом, в случае, если перенос в пару 10 будет, то с учетом правила 1.2 нужно: (4-2+1) – вычитать множимое. Таким образом, с учетом правил 1.1 и 1.2 правила выполнения действий упрощаются. Важно анализировать разряды текущей пары и разряд a_{2i-1} младшей пары. Тогда при умножении со старших разрядов действия выглядят так:

a_{2i+1}	a_{2i}	a_{2i-1}	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

5.2 Умножение чисел в форме с фиксированной запятой I способом с ускорением второго порядка.

Необходимо выполнить умножение чисел C=88 (множитель) и D=41 (множимое) с ФЗ I способом и проверить результат операции. Данные действия показаны на рисунке 18.

$C = 88_{10} = 1011000_2$ $D = 41_{10} = 101001_2 (M = 2^8)$

$C_{ПК}$ =00,01011000$D_{ПК}$ =00,00101001

1) Знак произведения: $0 \oplus 0 = 0$;

2) Перемножение сомножителей

Множитель	Множимое	Сумма ЧП	Комментарий
$00,010110\underline{00}^{(0)}$	00,00101001	00,00000000 00000000	Сдвиг
$00,000101\underline{10}^{(0)}$		00,00000000 00000000 $\underline{00,0101001000000000}$ 00,01010010 00000000	Сдвиг +2M
$00,000001\underline{01}^{(0)}$		00,00010100 10000000 $\underline{00,0010100100000000}$ 00,0011110110000000	Сдвиг +M
$00,000000\underline{01}^{(0)}$		00,0000111101100000 $\underline{00,00101001\ 00000000}$ 00,0011100001100000	Сдвиг +M
$00,000000\underline{00}^{(0)}$		00,0000111000011000	Сдвиг

$C*D_{ПК}$ = 00,00001110 00011000; $M_{(C*D)} = 2^8*2^8 = 2^{16}$

Проверка: C*D = 0,00001110 00011000_2= 3608_{10}

Рисунок 18 - Умножение чисел в форме с фиксированной запятой I способом с ускорением второго порядка

5.3 Умножение чисел в форме с фиксированной запятой II способом с ускорением второго порядка.

Необходимо выполнить умножение чисел C = -88 (множитель) и D = 41 (множимое) с ФЗ II способом и проверить результат операции. Данные действия показаны на рисунке 19.

$C = -88_{10} = -1011000_2$ $C_{ПК} = 11{,}01011000$
$D = 41_{10} = 101001_2$ $D_{ПК} = 00{,}00101001$
1) Знак произведения: $1 \oplus 0 = 1$; $(M = 2^8)$
2) Перемножение сомножителей

Множитель	Множимое	Сумма ЧП	Комментарий
00,01011<u>00</u>$^{(0)}$	00,00000000 00101001	00,00000000 00000000	Сдвиг
00,00010<u>110</u>$^{(0)}$	00,00000000 10100100	00,00000000 00000000 <u>00,00000001 01001000</u> 00,0000000101001000	Сдвиг +2M
00,000001<u>01</u>$^{(0)}$	00,00000010 10010000	00,00000001 01001000 <u>00,00000010 10010000</u> 00,00000011 11011000	Сдвиг +M
00,000000<u>01</u>$^{(0)}$	00,00001010 01000000	00,00000011 11011000 <u>00,00001010 01000000</u> 00,00001110 00011000	Сдвиг +M
00,00000000$^{(0)}$	00,00101001 00000000	00,00001110 00011000	Сдвиг

$(A*B)_{ПК} = 0{,}00001110\ 00011000$; $M_{(C*D)} = 2^8 * 2^8 = 2^{16}$
Проверка: A*B= - 1110 00011000= -3608

Рисунок 19 - Умножение чисел в форме с фиксированной запятой II способом с ускорением второго порядка

5.4 Умножение чисел в форме с фиксированной запятой III способом с ускорением второго порядка.

Необходимо выполнить умножение чисел C=88 (множимое) и D=-41 (множитель) с ФЗ III способом и проверить результат операции. Данные действия показаны на рисунке 20.

$C = 88_{10} = 1011000_2$ $\qquad C_{ПК}=00{,}01011000$
$D = -41_{10} = -101001_2$ $\qquad D_{ПК}=11{,}00101001$

1) Знак произведения: $0 \oplus 1 = 1$;

2) Перемножение модулей

Множитель	Множимое	Сумма ЧП	Комментарий
<u>00,0</u>1011000	00,00101001	00,00000000 00000000	Сдвиг
<u>01,0</u>1100000		00,00000000 00000000 <u>00,00000000 00101001</u> 00,00000000 00101001	Сдвиг +M
<u>01,1</u>0000000		00,00000000 10100100 <u>00,00000000 01010010</u> 00,00000000 11110110	Сдвиг +2M
<u>10,0</u>0000000		00,00000011 11011000 <u>11,11111111 10101110</u> 00,00000011 10000110	Сдвиг –2M
<u>00,0</u>0000000		00,00001110 00011000	Сдвиг

$(C*D)_{ДК}=00{,}00001110\ 00011000_2 M_{(C*D)} = 2^7 * 2^7 = 2^{14}$

Проверка: $C*D = -111000011000_2 = -3608_{10}$

Рисунок 20 - Умножение чисел в форме с фиксированной запятой III способом с ускорением второго порядка

5.5 Умножение чисел в форме с фиксированной запятой IV способом с ускорением второго порядка.

Необходимо выполнить умножение чисел C = -88 (множимое) и D = -41 (множитель) с ФЗ IV способом и проверить результат операции. Данные действия показаны на рисунке 21.

$C = -88_{10} = -1011000_2$ $C_{ПК} = 11{,}01011000$

$D = -41_{10} = -101001_2$ $D_{ПК} = 11{,}00101001$

1) Знак произведения: $1 \oplus 1 = 0$;

2) Перемножение модулей

Множитель	Множимое	Сумма ЧП	Комментарий
00,01011000	00,00101001 00000000	00,00000000 00000000	Сдвиг
01,01100000	00,00001010 01000000	00,00000000 00000000 00,00001010 01000000 00,00001010 01000000	Сдвиг +М
01,10000000	00,00000010 10010000	00,00001010 01000000 00,00000101 00100000 00,00001111 01100000	Сдвиг +2М
10,00000000	00,00000000 10100100	00,00001111 01100000 11,1111111010111000 00,0000111000011000	Сдвиг -2М
00,00000000	00,00000000 00101001	00,0000111000011000	Сдвиг

$(C*D)_{ДК} = 00{,}0000111000011000_2$ $M_{(C*D)} = 2^7 * 2^7 = 2^{14}$

Проверка: $C*D = -111000011000_2 = -3608_{10}$

Рисунок 21 - Умножение чисел в форме с фиксированной запятой IV способом с ускорением второго порядка

6. Сложение двоично – десятичных чисел

6.1 Теоретическая часть

6.1.1 Код с естественными весами 8-4-2-1. Алгоритм сложения:

1) Проверить знаки слагаемых, отрицательные преобразовать в ОК путем инвертирования тетрад и прибавления к каждой кода 1010, единицы переноса между тетрадами отбрасываются.
2) Сложить двоично-десятичные числа по правилам двоичной арифметики.
3) Выполнить коррекцию результата, прибавив код 0110 к «неправильным» тетрадам, а также к тетрадам, в которых сформировались единицы переноса при выполнении сложения. Здесь единицы переноса между тетрадами учитываются.

Проверить знак результата, „ - ” преобразовать в ПК путем инвертирования тетрад и прибавления к каждой кода 1010. Единицы переноса отбрасываются.

6.1.2 Код с избытком три 8-4-2-1 + 3.Алгоритм сложения:

1) Проверить знаки слагаемых: отрицательные преобразовать в ОК путем инвертирования тетрад.
2) Сложить 2-10 числа по правилам двоичной арифметики.
3) Выполнить коррекцию результата, прибавив код 1101 к тетрадам суммы, из которых не формировались единицы переноса, и прибавив код 0011 к тетрадам суммы, из которых формировались единицы переноса. Единицы переноса между тетрадами при коррекции отбрасывать.
4) Проверить знак результата: отрицательный преобразовать в ПК инвертированием тетрад.
5) Проверить знаки слагаемых: отрицательные преобразовать в ОК путем инвертирования тетрад.
6) Сложить 2-10 числа по правилам двоичной арифметики.
7) Выполнить коррекцию результата, прибавив код 1101 к тетрадам суммы, из которых не формировались единицы переноса, и прибавив код 0011 к тетрадам суммы, из которых формировались единицы переноса. Единицы переноса между тетрадами при коррекции отбрасывать.
8) Проверить знак результата: отрицательный преобразовать в ПК инвертированием тетрад.

6.1.3 Код 2-4-2-1

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

- если каждая из исходных тетрад< 5, то:

а) если суммарная тетрада< 5, то коррекция не нужна
б) если суммарная тетрада≥ 5, то коррекция кодом 0110

- если одна из исходных тетрад< 5, а вторая ≥ 5, то коррекции нет
- если каждая тетрада≥ 5, т.е. при сложении возникает единица переноса, то

а) если тетрадасуммы≥ 5, то коррекция не нужна
б) если тетрадасуммы< 5, то коррекция кодом 1010

Алгоритм сложения:

1) Проверить знаки слагаемых, отрицательные преобразовать в ОК, инвертируя тетрады
2) Сложить 2-10 числа по правилам двоичной арифметики
3) ыполнить коррекцию суммы в соответствии с изложенными выше правилами, прибавляя коды 0110 и 1010 к соответствующим тетрадам. Единицы переноса при коррекции отбрасывать.
4) Проверить знак результата: отрицательный преобразовать в ПК инвертированием тетрад.

5.1.4 Код 3а+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.

6.2 Сложение в коде 8 – 4 – 2 – 1.

Необходимо сложить числа A = -333,81$_{10}$, B = 803,35$_{10}$ в коде с естественными весами 8-4-2-1 и проверить результат. Данные действия показаны на рисунке 22.

A_{10} = -0,33381B_{10} = 0,80335(M=10^3)
Прямой код:
$A_{8\text{-}4\text{-}2\text{-}1}$ = 1, 0011 0011 0011 1000 0001
$B_{8\text{-}4\text{-}2\text{-}1}$ = 0, 1000 0000 0011 00110101

A: 1, 1100 1100 1100 0111 1110 - инверсия
1010 1010 1010 1010 1010 - коррекция
$A_{ОК}$ = 1, 0110 0110 0110 0001 1000

$B_{ОК}$= $B_{8\text{-}4\text{-}2\text{-}1}$=0, 1000 0000 0011 0011 0101

$A_{ОК}$ = 1, 0110 0110 01100001 1000
$B_{ОК}$ = 0, 1000 0000 0011 0011 0101
1, 1110 0110 1001 01001101
0110 0110 - коррекция
$(A+B)_{ОК}$= 0, 01000110 1001 0101 0100

$(A+B)_{ПК}$= 0, 01000110 1001 0101 0100
(A+B) =01000110 1001, 0101 0100
Проверка: A+B =-333,81+803,35$_{10}$=469,54

Рисунок 22 - Сложение в коде 8 – 4 – 2 – 1

6.3 Сложение в коде 8 – 4 – 2 – 1+3

Необходимо сложить числа A = -333,81, B= -803,35 в коде с избытком 3 и проверить результат. Данные действия показаны на рисунке 23.

A_{10} = -0,033381
B_{10} = -0,080335
$M=10^4$
Прямой код:
$A_{8\text{-}4\text{-}2\text{-}1+3}$= 1, 0011 0110 0110 0110 1011 0100
$B_{8\text{-}4\text{-}2\text{-}1+3}$= 1, 0011 1011 0011 0110 0110 1000

Сложение чисел:

$A_{ОК}$=	1, $^{(1)}1100^{(0)}1001^{(1)}1001^{(1)}1001^{(0)}0100^{(1)}1011$	
$B_{ОК}$=	1, 1100 0100 1100 1001 1001 0111	
$(A+B)_{ОК}$=	1, 1000 1110 0110 0010 11100011	
	0011 1101 0011 0011 11010011	- коррекция
	1, 1011 1011 1001 0101 10110110	
$(A+B)_{ПК}$=	1, 0100 0100 0110 1010 01001001	

$(A+B)_{8\text{-}4\text{-}2\text{-}1+3}$=(-0100 0100 0110 1010,01001001)
A+B=-1137,16
Проверка: A+B =-333,81-803,35= -1137,16

Рисунок 23 - Сложение в коде 8 – 4 – 2 – 1+3

6.4 Сложение в коде Айкена

Необходимо сложить числа A = 333,81, B= -803,35 в коде Айкена и проверить результат. Данные действия показаны на рисунке 24.

A_{10}=0,33381B_{10}= -0,80335   M=10^3
Прямой код:
$A_{2-4-2-1}$= 0, 0011 0011 0011 11100001
$B_{2-4-2-1}$= 1, 1110 0000 0011 0011 1011

Сложение чисел:
$A_{ОК}$=   0, 0011 0011 001111100001
$B_{ОК}$=   1, 0001 1111 1100 1100 0100
$(A+B)_{ОК}$=   1, 0101 0011 0000 1010 0101
     0110 1010 0110   - коррекция
     1, 10110011000001001011
$(A+B)_{ПК}$=   1, 01001100111110110100

$(A+B)_{2-4-2-1}$=(-010011001111,10110100)
A+B=-469,54
Проверка: A+B =333,81-803,35= -469,54

Рисунок 24 - Сложение в коде Айкена

6.5 Сложение в пентадном коде

Необходимо сложить числа $A = 333{,}81_{10}$, $B = 803{,}35_{10}$ в пентадном коде и проверить результат. Данные действия показаны на рисунке 25.

A_{10} = 0,033381B_{10} = 0,080335M=10^4

Прямой код:

A_{3a+2}= 1, 00010 01011010110101111101000101

B_{3a+2}= 1, 00010 11010 00010 01011 0101110001

Сложение чисел:

A_{3a+2}=	0, $^{(0)}$00010$^{(1)}$01011$^{(0)}$01011$^{(0)}$01011$^{(1)}$11010$^{(0)}$00101$^{(0)}$
B_{3a+2}=	0, 00010 11010 00010 01011 01011 10001
$(A+B)_{ОК}$=	0, 00101 00101 01101 1011100101 10110
	11110 11110 - коррекция
$(A+B)_{ПК}$=	0, 00101 00101 01011 10111 00101 10100

$(A+B)_{3a+2}$=(00101 001010101110111,00101 10100)

A+B=1137,16

Проверка: A+B =333,81+803,35= 1137,16

Рисунок 25 - Сложение в пентадном коде

7. Библиографический список

1. Фадеева Т.Р., Долженкова М.Л. Организация арифметических операций над двоичными числами. – Киров: Изд-во ВятГУ, 2001. – 40 с.
2. Ростовцев, В.С. Оформление курсовых и дипломных проектов для студентов специальности 230101 [Текст] / Ростовцев В.С., Блинова С.Д. – Киров: изд-во ВятГТУ,2006. – 39 с.
3. Савельев А. Я. Прикладная теория цифровых автоматов: Учебник / Савельев, Александр Яковлевич. - М.: Высшая школа., 1987. - 272с.
4. Лысиков Б. Г. Арифметические и логические основы цифровых автоматов: Учебник / Лысиков Борис Григорьевич. – 2-е издание, переработанное и дополненное. – Минск: Высшая Школа, 1980 – 335с.
5. Дудкин В.С. и др. Машинные алгоритмы десятичной арифметики: Учебное пособие / Дудкин Виктор Степанович, Кутепова Евгения Соломоновна, Матвеев Виталий Дмитриевич; – Горький; Изд-во ГГУ, 1982 – 60 с.
6. Каган Б. М. Электронные вычислительные машины и системы: Учеб. пособие / Каган, Борис Моисеевич. - 3-е изд., переработанное и дополненное - М.: Энергоатомиздат, 1991. - 592с.