

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Локтионова Оксана Геннадьевна
Должность: проректор по учебной работе
Дата подписания: 12.02.2021 16:02:46
Уникальный программный ключ:
0b817ca911e6668abb13a5d426d39e5f1c11eabbf73e943df4a4851fda56d089

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное
учреждения высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра информационной безопасности

УТВЕРЖДАЮ
Проректор по учебной работе
О.Г. Локтионова
« 12 » 2018 г.



**Моделирование работы операционного автомата,
выполняющего операции деления в прямом коде**
Методические рекомендации для лабораторной работы
для студентов укрупненной группы специальностей и
направлений подготовки 10.00.00 «Информационная безопасность»

Курск 2018

УДК 004

Составитель: С. С. Шевелёв

Рецензент

Кандидат технических наук, доцент кафедры «Информационная безопасность» А.Л. Марухленко

Моделирование работы операционного автомата, выполняющего операции деления в прямом коде [Текст] : методические рекомендации для лабораторной работы №3 по дисциплине «Организация ЭВМ и вычислительных систем»/ Юго-Зап. гос. ун-т; сост.: С. С. Шевелёв. – Курск, 2018. – 24 с.: блок-схемы алгоритма. 5, табл. 1. – Библиогр.: с. 24.

Содержат сведения по вопросам работы операционного автомата, выполняющего операции деления в прямом коде. Указывается порядок выполнения практических и самостоятельных работ, правила оформления отчета.

Методические рекомендации соответствуют требованиям программы, утвержденной учебно-методическим объединением по специальности.

Предназначены для студентов укрупненной группы специальностей и направлений подготовки 10.00.00 «Информационная безопасность».

Текст печатается в авторской редакции

Подписано в печать 1.02.19. Формат 60x84 1/16.

Усл.печ. л. 1,39. Уч.-изд. л. 1,26. Тираж 100 экз. Заказ. Бесплатно. 23P

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

Лабораторная работа №3

Моделирование работы операционного автомата, выполняющего операции деления в прямом коде.

Цель работы: изучить структуру операционного автомата, на основе данной структуры создать модель операционного автомата, выполняющего операции деления чисел с восстановлением остатков, без восстановления остатков, машинное деление.

Задача: По представленным блок-схемам алгоритмов написать программу, протестировать на языке высокого уровня.

Теоретическая часть

Деление чисел с восстановлением остатков

Операция деления в ЭВМ выполняется в прямом коде. Это значительно упрощает комбинационные схемы вычислительных устройств. При делении правильных дробей возможно переполнение разрядной сетки в случае, когда делимое больше делителя. Признаком переполнения является появление единицы в знаковом разряде частного. Это ведет к искажению результата. Во избежание такой ситуации необходимо соблюдать условие: $|A| < |B|$, где A — делимое; B — делитель. Частное определяется путем деления модулей исходных чисел.

Знак частного при делении в прямом коде определяется как сумма по модулю 2 знаковых цифр делимого и делителя и присваивается частному в конце операции деления.

Пусть требуется разделить A на B с точностью до i -го разряда.

Тогда

$$Y_i = \frac{A}{B} = 0, y_1 y_2 y_3 \dots y_{i-1} y_i = y_1 \cdot 2^{-1} + y_2 \cdot 2^{-2} + \dots + y_{i-1} \cdot 2^{-(i-1)} + y_i \cdot 2^{-i}. \quad (1)$$

При любом значении i должно выполняться неравенство

$$0 \leq A - BY_i < B \cdot 2^{-i} \quad (2)$$

т. е. остаток от деления $(A - BY_i)$ должен быть меньше делителя умноженного на 2^{-i} .

Преобразуем (2) к виду

$$0 \leq (A - BY_i) \cdot 2^i < B. \quad (3)$$

Обозначив $(A - BY_i) \cdot 2^i = C_i$, получим

$$0 \leq C_i < B. \quad (4)$$

Цифры частного определяются последовательно, начиная со старшего разряда.

В результате выполнения i циклов получены старшие i разрядов частного, т. е. приближенное значение частного Y_i . В следующем $(i+1)$ -м цикле будет получено значение $(i+1)$ -го разряда частного. Исходными данными для этого цикла являются Y_i и C_i .

Цифра y_{i+1} может иметь одно из двух значений: 1 или 0. Если $y_{i+1}=0$, то

$$Y_{i+1} = Y_i + y_{i+1} \cdot 2^{-(i+1)} = Y_i \quad (5)$$

$$C_{i+1} = (A - BY_{i+1}) \cdot 2^{i+1} = 2C_i \quad (6)$$

т. е. в частном записывается 0 при условии

$$0 \leq C_{i+1} = 2C_i < B \quad (7)$$

Если $y_{i+1} = 1$, то

$$Y_{i+1} = Y_i + 2^{-(i+1)}, \quad (8)$$

$$C_{i+1} = (A - B \cdot Y_{i+1}) \cdot 2^{i+1} = (A - BY_i - B \cdot 2^{-(i+1)}) \cdot 2^{i+1} = 2C_i - B, \quad (9)$$

или т. е. цифра частного равна 1, если выполняется условие

$$0 \leq C_{i+1} = 2C_i - B < B, \quad (10) \text{ и } (11)$$

$$B \leq 2C_i < 2B.$$

После завершения n -го цикла мы получим n -значное частное Y_n , вычисленное с недостатком $C_n = (A - BY_n) 2^n$, который равен остатку от деления A на B , сдвинутому влево на n разрядов.

Правило деления с восстановлением остатков формулируется следующим образом. Делитель вычитается из делимого и определяется знак нулевого (по порядку) остатка. Если остаток положительный, т. е. $|A| > |B|$, то в псевдознаковом разряде частного проставляется 1, при появлении которой формируется признак переполнения разрядной сетки и операция прекращается. Если остаток отрицательный, то в псевдознаковом разряде частного записывается 0, а затем производится восстановление делимого путем добавления к остатку делителя. Далее выполняется сдвиг восстановленного делимого на один разряд влево и повторное вычитание делителя. Знак получаемого таким образом остатка определяет первую значащую

цифру частного: если остаток положителен, то в первом разряде частного записывается 1, если отрицательный, то записывается 0. Далее, если остаток положителен, то он сдвигается влево на 1 разряд и из него вычитается делитель для определения следующей цифры частного. Если остаток отрицателен, то к нему прибавляется делитель для восстановления предыдущего остатка, затем восстановленный остаток сдвигается на 1 разряд влево и от него вычитается делитель для определения следующей цифры частного и т. д. до получения необходимого количества цифр частного с учетом одного разряда для округления, т. е. до обеспечения требуемой точности деления.

Таким образом, цифры частного получаются как инверсное значение знаковых разрядов текущего остатка, которые принимают значение 00 или 11. Однако при сдвиге остатка влево в знаковых разрядах может возникнуть сочетание 01. В некоторых случаях для того чтобы цифры частного формировались как прямое значение знакового разряда текущего остатка, деление выполняют с инверсными знаками. При этом делимое передается в сумматор не прямым, а инверсным кодом, а на нулевом шаге выполняется операция «+В», вместо операции «—В».

Деление чисел без восстановления остатков

Деление чисел А и В без восстановления остатков отличается тем, что при получении отрицательного остатка необходимо алгебраически прибавить его с положительном делителем. Рассмотрим случай, когда $C_i < 0$.

Если не восстанавливать остаток, а сразу сдвинуть отрицательный остаток - C_i на один разряд влево, то получим

$$C'_{i+1} = 2C_i = 2(2C_{i-1} - |B|) = 4C_{i-1} - 2|B|. \quad (12)$$

Результат в данном случае отличается от действительного на величину $+|B|$. Поэтому в качестве второго шага необходимо произвести коррекцию результата на эту величину;

$$C'_{i+1} = 4C_{i-1} - 2|B| + |B| = 4C_{i-1} - |B|. \quad (13)$$

В результате получаем требуемую величину последующего остатка C_{i+1} за 2 шага.

Таким образом, чтобы определить следующую цифру частного, необходимо сдвинуть текущий остаток влево на один разряд, а затем

алгебраически прибавить к нему модуль делителя, которому приписывается знак, противоположный знаку текущего остатка. Знак полученного таким образом следующего остатка и определяет следующую цифру частного: если остаток положительный, то в частном записывается 1, если отрицательный—записывается 0. Операция сдвигов и алгебраических сложений повторяется до тех пор, пока в частном не получится требуемое количество цифр.

Деление чисел во всех ЭВМ производится по способу без восстановления остатков – это значительно повышает производительность вычислительных устройств:

1. упрощает схему управления процессом деления,
2. увеличивает быстродействие ЭВМ, так как длительность операции деления без восстановления остатков равна минимальной длительности операции деления с восстановлением остатков.

При выполнении операции деления результат получится одинаковым, если сдвигать остатки от деления влево либо делитель вправо. Следовательно, возможны две схемы выполнения деления:

1. деление без восстановления остатков со сдвигом делителя вправо;
2. деление без восстановления остатков со сдвигом влево.

Для реализации первого варианта необходимы $2n$ -разрядный регистр делителя со сдвигом вправо, $2n$ -разрядный сумматор, $(n+1)$ -разрядный регистр частного со сдвигом влево.

Для реализации второго варианта необходимы: n -разрядный регистр делителя; $(n+1)$ -разрядный регистр частного со сдвигом влево; n - или $(n + 1)$ -разрядный сумматор со сдвигом влево и схема управления. Сумматор производит сдвиг текущего остатка влево и алгебраическое сложение его с делителем.

Деление чисел в машинах с плавающей запятой

Если числа A и B заданы в нормальной форме, то их частное будет равно

$$Y = a : b = (a : b) \cdot 2^{m_1 - m_2}, \quad (14)$$

где a и b — мантиссы, а m_1 и m_2 — порядки соответственно чисел A и B . Отсюда следует, что операция деления в машинах с плавающей запятой выполняется в пять этапов.

1. Определение знака частного путем сложения по модулю 2 знаковых цифр мантисс операндов.
2. Деление модулей мантисс операндов по правилам деления чисел с фиксированной запятой.
3. Определение порядка частного путем вычитания порядка делителя из порядка делимого.
4. Нормализация результата и его округление.
5. Присвоение знака мантиссе результата. Два первых этапа полностью совпадают с этапами деления чисел с фиксированной запятой. Третий этап представляет собой обычное сложение в инверсных кодах.

При делении нормализованных чисел денормализация результата возможна только влево и только на один разряд. Это обусловлено тем, что мантисса любого нормализованного числа лежит в пределах

$$2^{-1} \leq |a| < 1 - 2^{-n}. \quad (15)$$

Тогда наименьшая и наибольшая возможные величины мантиссы частного равны соответственно

$$|y_{\min}| = \frac{2^{-1}}{1 - 2^{-n}} > 2^{-1}; \quad |y_{\max}| = \frac{1 - 2^{-n}}{2^{-1}} = 2 - 2^{-(n-1)} < 2, \quad (16)$$

т. е. мантисса частного лежит в пределах

$$2^{-1} < |y| < 2 \quad (17)$$

Поэтому на четвертом этапе может возникнуть необходимость нормализации мантиссы частного путем ее сдвига вправо на один разряд и увеличения порядка частного на единицу. Если же перед делением сдвинуть делимое на один разряд вправо, то на 4-м этапе может потребоваться нормализация результатов влево на 1 разряд.

При выполнении третьего этапа порядок частного может оказаться больше допустимого, т. е. произойдет переполнение разрядной сетки, которое расценивается как аварийная ситуация. Может быть также получен порядок частного, который меньше допустимого. Если на втором этапе вычислено частное $|y| > 1$, то на четвертом этапе при сдвиге частного вправо его порядок должен быть увеличен на единицу. При этом, во-первых, возникло ли переполнение разрядной сетки порядка или нет, во-вторых, будет ли порядок частного меньше допустимого значения.

Структурная схема операционного устройства, выполняющего операцию деления чисел в дополнительном и обратном коде.

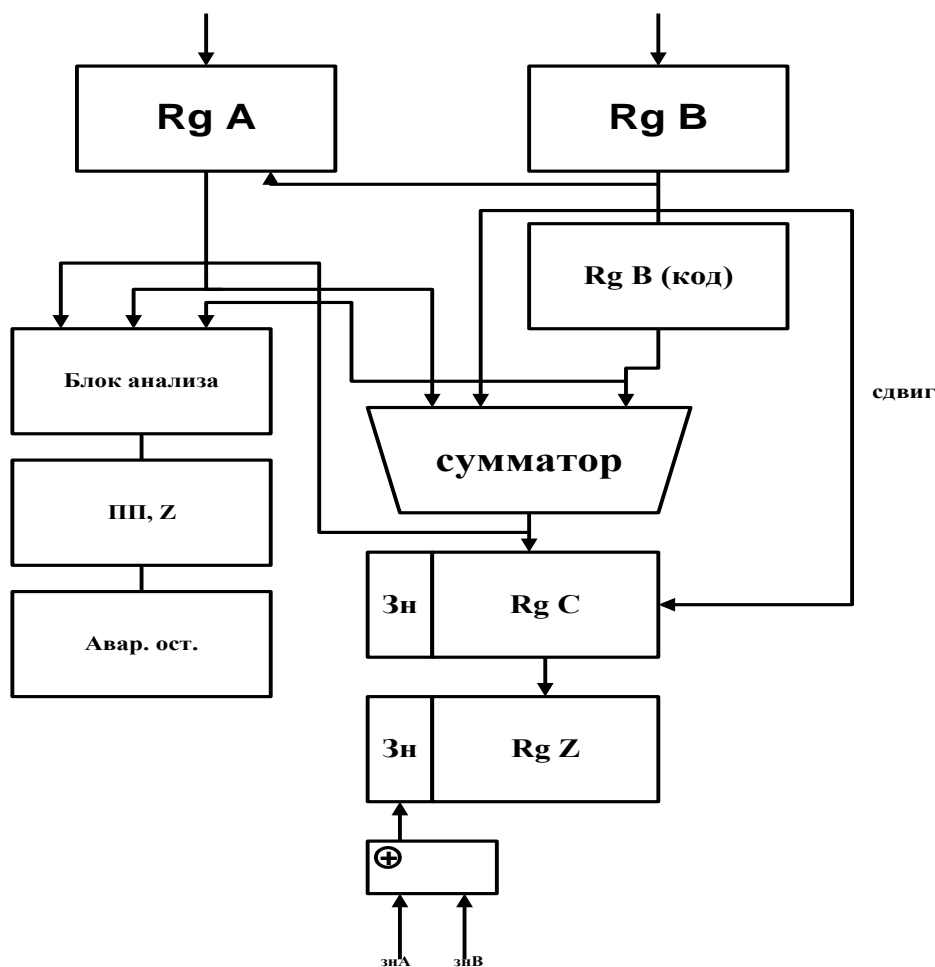


Рис. 1

На Рис.1 представлена структурная схема операционного автомата, выполняющего операцию деления чисел в прямых кодах, где представлено: $Z_p A$ –знаковый разряд числа A, $Rg A_{пр}$ – регистр для хранения модуля числа A в прямом коде, $Rg A_k$ – регистр для хранения числа A в коде, $Z_p B$ – знаковый разряд числа B, $Rg B_{пр}$ – регистр для хранения модуля числа B в прямом коде, $Rg B_k$ – регистр для хранения числа B в коде, Σ - сумматор, K_{op} - сумматор корректор, $Z_p Z$ –знаковый разряд числа Z, $Rg Z_k$ – регистр для хранения результата Z в коде, $Rg Z_{пр}$ – регистр для хранения результата Z в прямом коде. Знаковый разряд частного $Z_n Z$

определяется на сумматоре по модулю 2. Входными величинами, которого являются знаковые разряды - $Z_n A$ и $Z_n B$. Если операнды имеют одинаковые знаки, то знак частного положительный, если разные, то знак отрицательный.

Блок-схема алгоритма деления без восстановления остатков

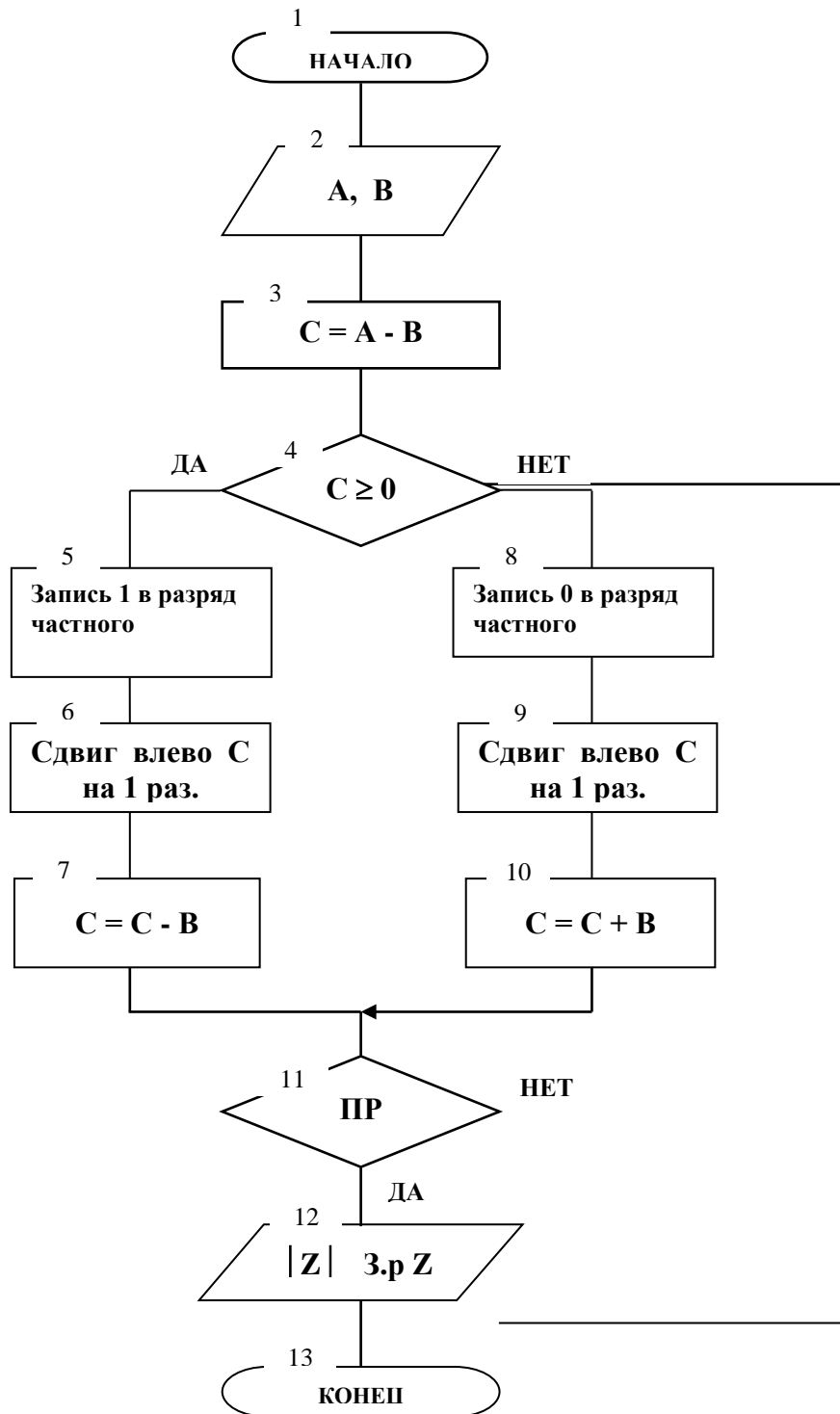


Рис. 2

Работа алгоритма деления чисел без восстановления остатков в прямом коде.

Блок 1 алгоритма является начальным. В блоке 2 алгоритма осуществляется ввод чисел A и B в десятичной системе счисления, где число A это - делимое, а число B это - делитель. В этом же блоке происходит перевод чисел A и B в двоичную систему счисления. Число B (делитель) будет представлено положительном в прямом коде, так и отрицательном в дополнительном коде. Число A будет положительным в прямом коде.

В блоке 3 алгоритма вычисляется разность чисел A и B . Переменной C присваивается результат разности по команде: $C = A - B$. В блоке 4 алгоритма анализируется разность чисел A и B – число C . Если число C больше нуля ($C \geq 0$), то в блоке 5 алгоритма формируется разряд частного. Записывается единица в очередной разряд частного. В блоке 6 алгоритма происходит сдвиг разрядов числа C на один разряд влево, при этом справа к остатку C приписывается очередной разряд числа A . Крайний левый разряд числа C при этом теряется. В блоке 7 алгоритма вычисляется разность между числами C и B . Переменной C присваивается результат разности по команде $C = C - B$. Если число C меньше нуля ($C \leq 0$), то в блоке 8 алгоритма формируется разряд частного. Записывается значение нуля в очередной разряд частного. В блоке 9 алгоритма происходит сдвиг разрядов числа C на один разряд влево. Очередной разряд числа A сносится к числу C и приписывается в крайний правый разряд. Крайний левый разряд числа C при этом теряется. В блоке 10 алгоритма вычисляется сумма между числами C и B . Переменной C присваивается результат суммы по команде

$$C = C + B.$$

В блоке 11 алгоритма анализируется признак получения всех разрядов частного - ПР. Если не все разряды частного получены - выход НЕТ, то осуществляется переход на блок 4 алгоритма для формирования очередного разряда частного. Если все разряды частного получены – вход ДА, при этом все разряды делимого учтены, то происходит выдача результата Z в виде модуля числа $|Z|$ и знакового разряда (знака частного) - З.р Z в блоке 12 алгоритма. Блок 13 алгоритма является конечным.

Чтобы определить следующую цифру частного необходимо сдвинуть текущий остаток влево на один разряд, а затем алгебраически прибавить к нему модуль делителя, которому приписывается знак, противоположный по знаку текущего остатка. Знак полученного таким образом следующего остатка и определяет следующую цифру частного, если остаток положительный, то в частном записывается 1, если остаток отрицательный, то в частном записывается 0. Операция сдвигов и алгебраических сложений повторяется до тех пор, пока в частном не получится требуемое количество цифр.

Если при делении чисел все разряды частного получены, но при этом все разряды делимого учтены, а полученный остаток равен отрицательному числу, то в этом случае к отрицательному остатку необходимо прибавить прямой код делителя. Остаток всегда должен быть нулю.

Блок-схема алгоритма с восстановлением остатков:

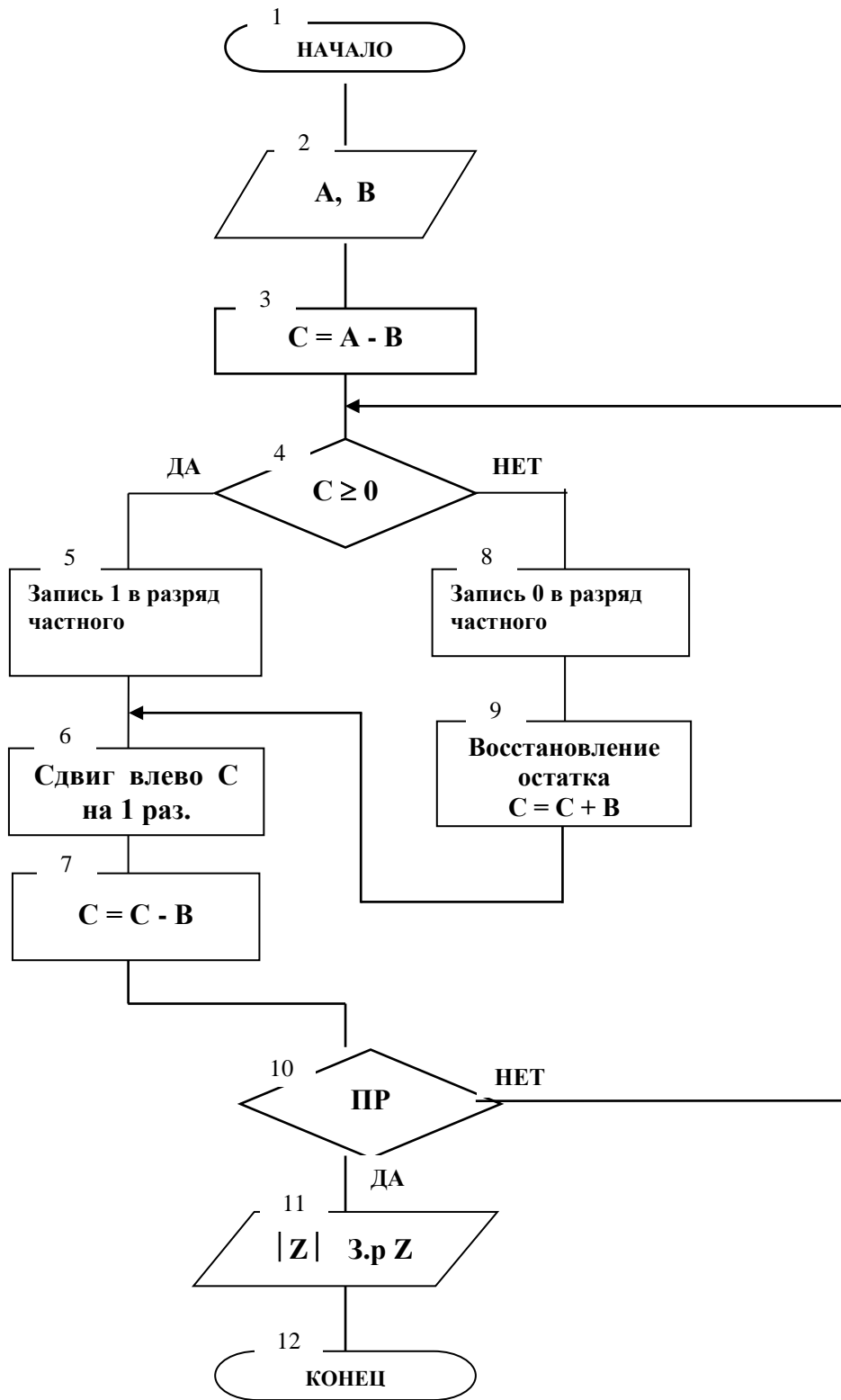


Рис. 3

Работа алгоритма деления чисел с восстановления остатков в прямом коде.

Блок 1 алгоритма является начальным. В блоке 2 алгоритма осуществляется ввод чисел A и B в десятичной системе счисления. В этом же блоке происходит перевод чисел A и B из десятичной в двоичную систему счисления.

В блоке 3 алгоритма вычисляется разность между числами A и B . Переменной C присваивается результат разности по команде: $C = A - B$. В блоке 4 алгоритма анализируется результат разности чисел A и B – число C .

Если число C больше нуля ($C \geq 0$), то в блоке 5 алгоритма формируется разряд частного. Записывается единица в очередной разряд частного. В блоке 6 алгоритма происходит сдвиг разрядов числа C на один разряд влево. Крайний левый разряд числа C при этом теряется, но справа приписывается очередной разряд числа A . В блоке 7 алгоритма вычисляется разность между числами C и B . Переменной C присваивается результат разности по команде $C = C - B$.

Если число C меньше нуля ($C \leq 0$), то в блоке 8 алгоритма формируется разряд частного в связи с этим записывается значение нуля в очередной разряд частного. В блоке 9 алгоритма восстанавливается остаток $-C$. Вычисляется сумма между числами B и C . Переменной C присваивается результат суммы по команде: $C = C + B$. С выхода блока 9 алгоритма осуществляется переход на блок 6 алгоритма.

В блоке 11 алгоритма анализируется признак получения всех разрядов частного ПР. Если не все разряды частного получены - выход НЕТ, то осуществляется переход на блок 4 алгоритма для формирования очередного разряда частного. Если все разряды частного получены – вход ДА, с учетом всех разрядов числа A , то происходит выдача результата Z в виде модуля числа $|Z|$ и знакового разряда (знака частного) - З.р Z в блоке 12 алгоритма. Блок 13 алгоритма является конечным.

Для определения следующей цифры частного необходимо выполнить ряд операций:

1. Если остаток положителен $C \geq 0$, то записывается 1 в разряд частного.

2. Если остаток отрицателен $C < 0$, то в разряд частного записывается 0 и при этом происходит восстановление делимого путем добавления к отрицательному остатку положительного делителя для восстановления предыдущего остатка, затем восстановленный остаток сдвигается на 1 разряд влево и от него вычитается делитель для определения следующей цифры частного и т.д.

Таблица 1

Знак остатка	Знак делителя	Операция	Значение очередной цифры частного
+	+	вычитание	1
+	-	сложение	0
-	+	сложение	0
-	-	вычитание	1

Пример 1. Выполнить операцию деления чисел А и В без восстановления остатков.

$A = 30; \quad B = 3.$

$V_{ПР} = 0.11$

$V_{ДП} = 1.01$

$A_{ПР} = 0.11110$

$A_{ПР}$	+	0.11110	
$V_{ДП}$		1.01	
$C > 0$		0.00	←
Сдвиг	+	0.01	
$\oplus Z_p B$		1.01	
$= 0$		1.10	
$C < 0$		1.10	←
Сдвиг	+	1.01	
$V_{ПР}$		0.11	
$C > 0$		0.00	
Сдвиг	+	0.00	

$Z = 0.1010_2 = 10_{10}$

$Z_p Z = Z_p A$

$Z_p Z = 0 \oplus 0$

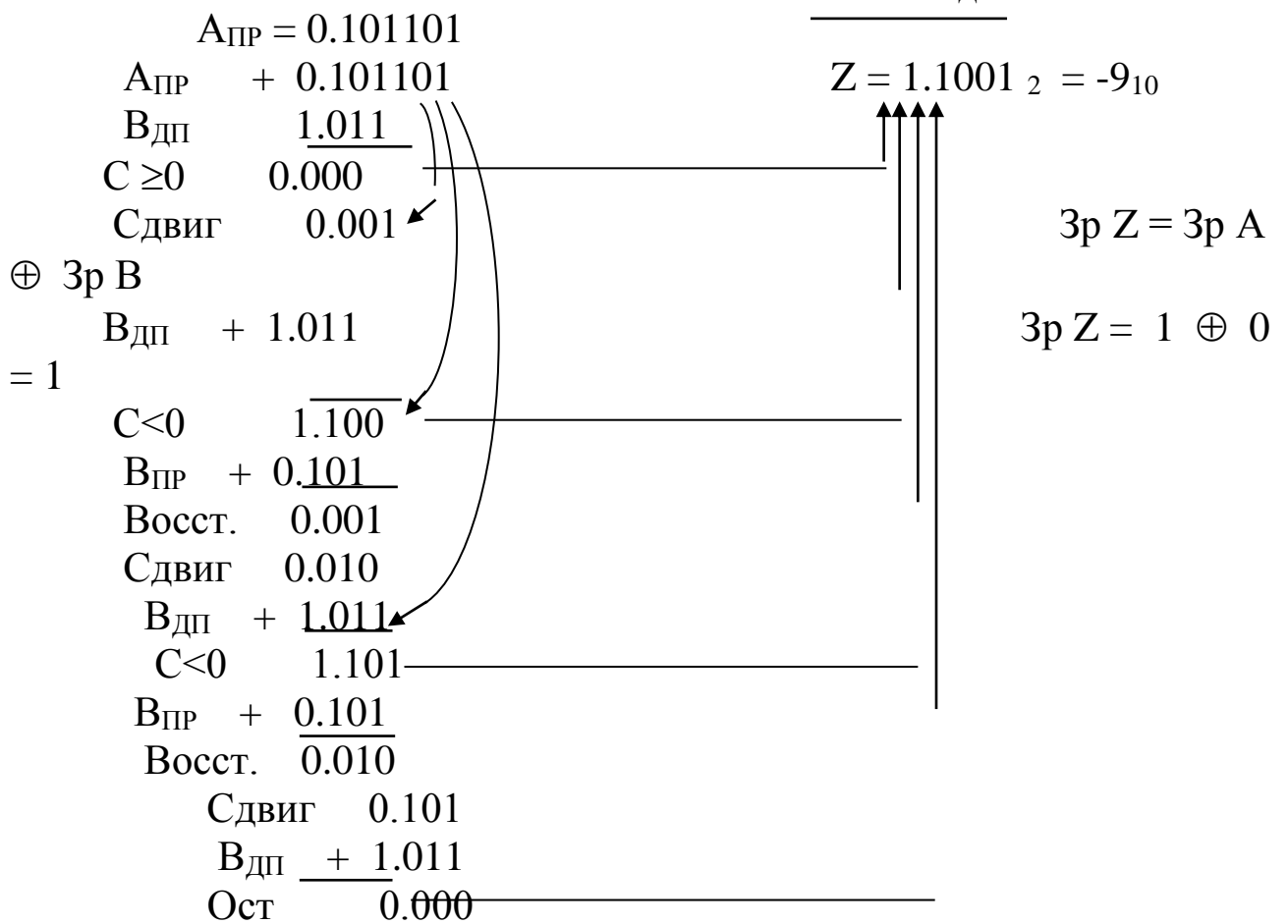
$$\begin{array}{r}
 \text{В}_{\text{ДП}} \quad \underline{1.01} \\
 \text{С} < 0 \quad + \quad \underline{1.01} \quad \text{-----} \\
 \text{В}_{\text{ПР}} \quad \underline{0.11} \\
 \text{Ост} \quad \underline{0.00}
 \end{array}$$

Пример 2. Выполнить операцию деления чисел А и В с восстановлением остатков.

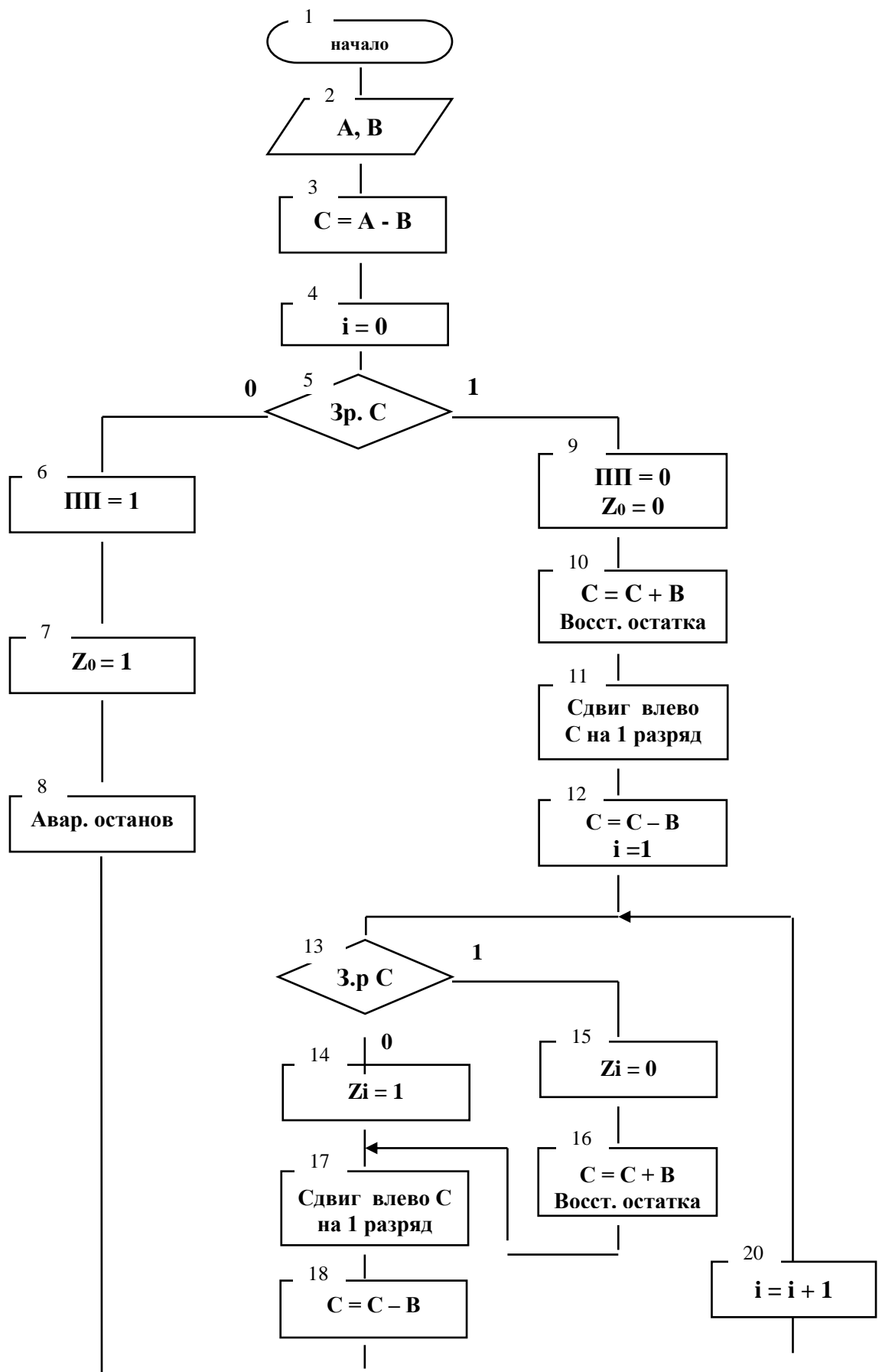
$$A = -45; \quad B = 5.$$

$$\text{В}_{\text{ПР}} = 0.101$$

$$\text{В}_{\text{ДП}} = 1.011$$



Блок-схема алгоритма машинного деления чисел с фиксированной запятой



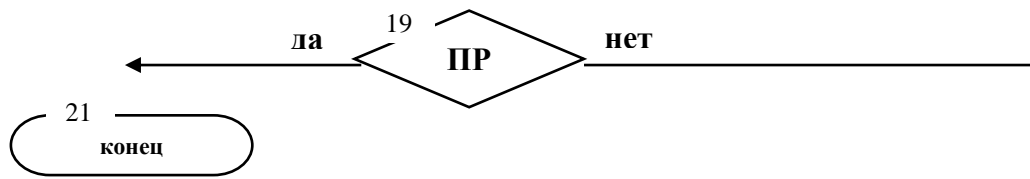


Рис. 4

Работа алгоритма машинного деления чисел с фиксированной запятой.

Блок 1 алгоритма является начальным. В блоке 2 алгоритма происходит ввод чисел A и B в десятичной системе счисления. В этом блоке осуществляется перевод из десятичной системы счисления в двоичную с учетом знакового разряда. В блоке 3 алгоритма вычисляется остаток – C . Переменной C присваивается результата разности между числами A и B по команде: $C = A - B$. В блоке 4 алгоритма переменной I – счетчика количества разрядов частного, присваивается значение нуля $I = 0$. В блоке 5 алгоритма анализируется знаковый разряд остатка – C (разности между числами A и B). Если знаковый разряд Z_r C равен единице т.е. отрицательный - выход **1**, то осуществляется переход на блок 9 алгоритма. Если знаковый разряд Z_r C равен нулю т.е. положительный - выход **0**, то происходит переход на блок 6 алгоритма. В блоке 6 алгоритма признаку переполнения $ПП$ присваивается значение единицы по команде: $ПП = 1$. В блоке 7 алгоритма псевдоразряду Z_0 присваивается значение единице по команде: $Z_0 = 1$. В блоке 8 алгоритма формируется сигнал Аварийного останова. Эта ситуация возникает тогда, когда делимое больше делителя $A > B$, в этом случае операция деления прекращается и вырабатывается Авар. останов.

В блоке 9 алгоритма признаку переполнения $ПП$ и псевдоразряду Z_0 присваивается значение нуля по командам: $ПП = 0, Z_0 = 0$. В блоке 10 алгоритма переменной C присваивается сумма чисел C и B тем самым восстанавливается значение остатка C . В блоке 11 алгоритма формируется сдвиг влево на один разряд остатка C . В блоке 12 алгоритма вычисляется разность между остатком C и делителем B , результат присваивается переменной C : по команде $C = C - B$.

В блоках 13 – 19 алгоритма формируется цикл, в результате которого поразрядно вычисляются значащие разряды частного. В результате выполнения команд в цикле формируется один разряд

частного. Количество разрядов частного (K – коэффициент степени точности) задается пользователем. Если скажем, степень точности K равно восьми, то процесс вычисления разрядов частного будет выполняться до девятого разряда. Последний разряд, в нашем случае девятый, анализируется. Если он равен нулю, то выполняется операция усечения результата. В этом случае отбрасываются все разряды частного, начиная с девятого. Если девятый разряд равен единице, то вначале выполняется операция округления результата. К восьмому разряду прибавляется единица, а затем выполняется операция усечения результата.

В блоке 13 алгоритма анализируется знаковый разряд остатка $C - Zp$. C . Если он равен нулю – выход **0**, то осуществляется переход на блок 15 алгоритма. Если знаковый разряд равен единице – выход **1**, то очередному разряду частного Zi присваивается значение единицы: по команде $Zi = 1$ - блок 14 алгоритма. В блоке 15 алгоритма разряду частного Zi присваивается значение нуля по команде: $Zi = 0$. В блоке 16 алгоритма переменной C присваивается значение суммы остатка C и делителя B по команде: $C = C + B$. В блоке 17 алгоритма выполняется операция сдвига влево остатка C на один разряд. В блоке 18 алгоритма переменной C присваивается разность чисел C и B по команде: $C = C - B$. В блоке 19 алгоритма анализируется признак получения всех разрядов частного с учетом степени точности – K . Если не все разряды частного получены – выход **НЕТ**, то осуществляется переход на блок 20 алгоритма, в котором к текущему значению счетчика разрядов i прибавляется единица по команде: $i = i + 1$. Если результата получен с указанной точностью – выход **ДА**, то происходит переход на блок 21 алгоритма, который является конечным.

Пример

$A=3$; $B=5$ степень точности - $K = 5$ деление
необходимо производить до 6 разряда

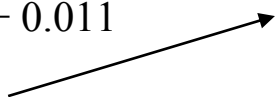
$$A_{\text{ПР}} = 0.011$$

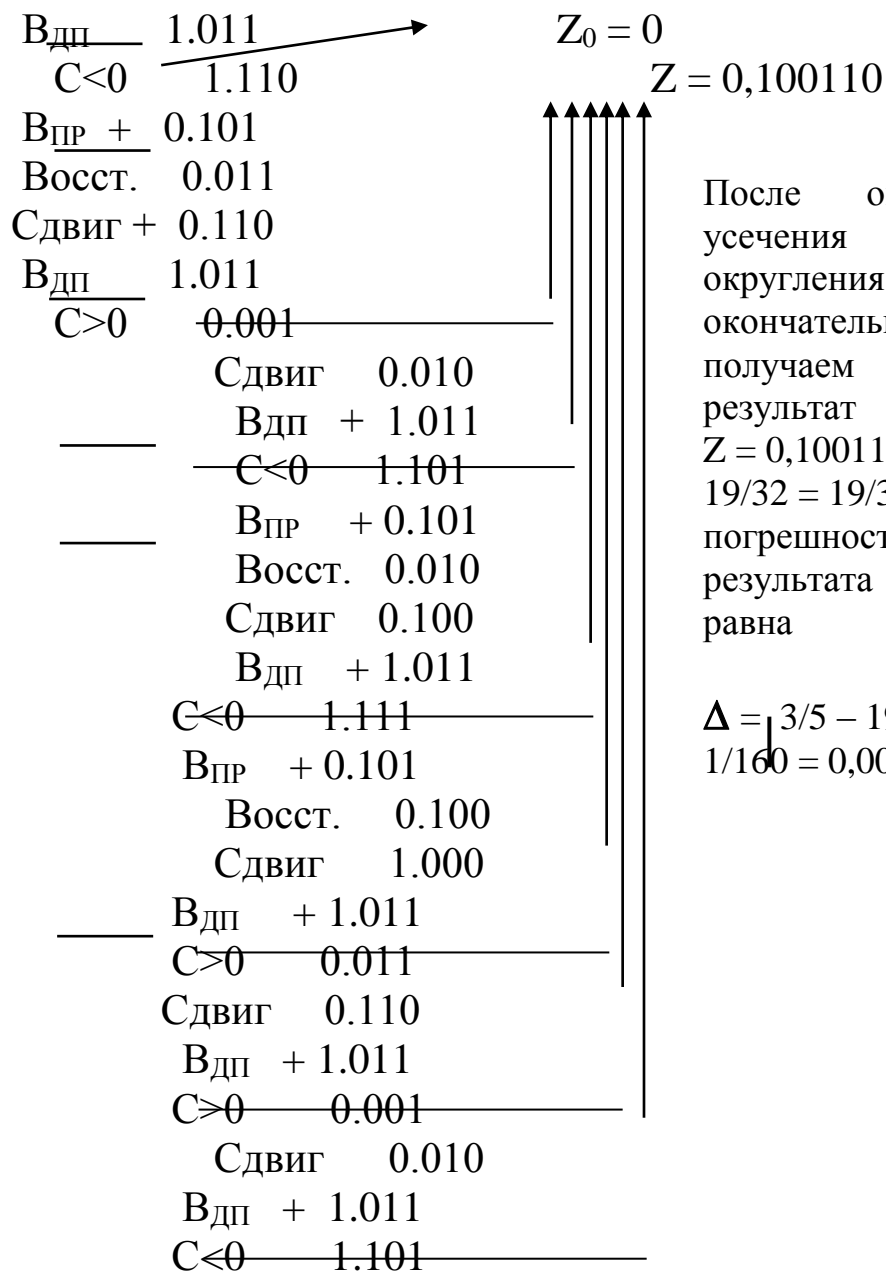
$$B_{\text{ПР}} = 0.101$$

$$B_{\text{ДП}} = 1.011$$

$$A_{\text{ПР}} + 0.011$$

$$ПП = 0$$





После операции усечения и округления окончательно получаем результат $Z = 0,10011 = 19/32 = 19/32$ погрешность результата равна

$$\Delta = \left| \frac{3}{5} - \frac{19}{32} \right| = \frac{1}{160} = 0,00625$$

Навыки в обращении с двоичными числами.

											0	1	2
К					6	2	4	28	56	12	024	048	096

$$1. \underbrace{100000\dots}_K 0 = 2^K \qquad 10000 = 2^4 = 16$$

К нулей

$$2. \underbrace{111111\dots}_K 1 = 2^K - 1 \qquad 11111 = 2^5 - 1 = 31$$

К единиц

$$3. \underbrace{a_5 a_4 a_3 a_2 a_1}_{\text{малое число}} \underbrace{000000\dots}_K 0 = a * 2^K$$

малое число К нулей

$$11010000 = 1101 * 2^4 = 13 * 16 = 208$$

$$4. \underbrace{a_5 a_4 a_3 a_2 a_1}_{\text{малое число a}} \underbrace{000000\dots}_b 0 \underbrace{b_5 b_4 b_3 b_2 b_1}_{\text{малое число b}} = a * 2^K + b$$

К разрядов

$$1010000110 = 101 * 2^7 + 110 = 5 * 128 + 6 = 646$$

чтение двоичных дробей

$$5. A = 0, \underbrace{0000\dots}_n 001 = 2^{-n} \qquad \underbrace{0,00001}_{4 \text{ нуля}} = 2^{-5} = 1/32$$

n - 1 нулей

4 нуля

$$6. A = 0, \underbrace{1001\dots}_n 1 = A_{10} / 2^n \qquad A = 0,101101 = 45/2^6 = 45/64$$

n - разрядов

Совмещенный алгоритм деления чисел в прямых кодах.

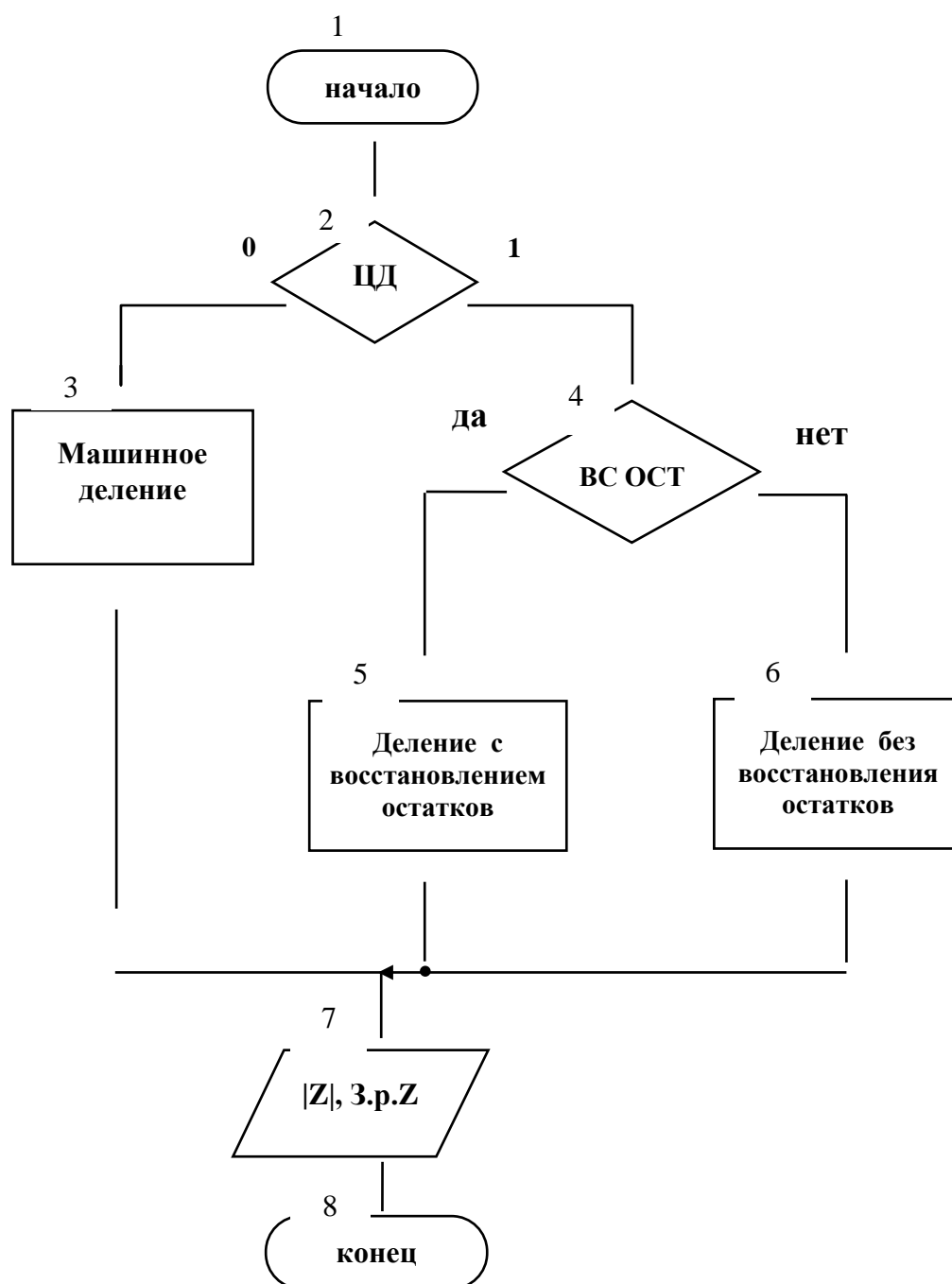


Рис.5

На Рис.5 обозначено: ЦД – признак целочисленного деления; если ЦД = 0, то происходит выполнение операции деления по алгоритму машинного деления чисел, процесс выполняется до тех пор, пока не будет получено необходимое количество разрядов частного,

если ЦД = 1, то необходимо сделать выбор алгоритмов деления: с восстановлением остатков или без восстановления остатков. ВС ОСТ - условие выбора алгоритмов деления. Если ВС ОСТ равен нулю, то операция деления осуществляется с помощью алгоритма деления с восстановлением остатков, если ВС ОСТ равен единице, то выполняется алгоритм без восстановления остатков.

2. Задание

1. Составить программу на языке высокого уровня по представленным блок-схемам алгоритмов:

- деление без восстановления остатков Рис.2
- деление с восстановлением остатков Рис.3
- машинное деление Рис.4

2. Совмещенный алгоритм деления чисел в прямых кодах (Рис.5).

3. Промоделировать (тестировать) программы на ПЭВМ.

4. Проанализировать результаты выполнения программ.

3. Содержание отчета

Отчет должен содержать:

- Титульный лист;
- Задание;
- Описание переменных, используемых в программе;
- Блок-схемы алгоритмов;
- Описание работы алгоритмов;
- Тексты программы;
- Результаты выполнения работы программ.

Контрольные вопросы

1. В каких кодах проще всего выполнять операцию деления.
2. Какие способы деления чисел с фиксированной запятой вам известны.

3. Как осуществляется работа алгоритма деления чисел с фиксированной запятой без восстановления остатков.
4. Назовите особенности работы алгоритма деления чисел с восстановлением остатков с фиксированной запятой.
5. При выполнении какой операции происходит восстановление остатков при делении чисел. Как это осуществляется в операционном автомате (рис.1) и в алгоритме.
6. Как определяется очередная цифра частного при делении чисел с восстановлением остатков.
7. Укажите главные отличия алгоритмов деления чисел с восстановлением остатков и без восстановления остатков.
8. По какому способу производится деление в современных ЭВМ. Чем это вызвано, на какие характеристики вычислительной машины это влияет и каким образом.
9. При каком условии выполняется машинное деление. Что является признаком выполнения машинного деления в ЭВМ.
10. Что такое степень точности. Как осуществляются операции округления и усечения результата.
11. Как формируются разряды частного. По какому признаку происходит выход из цикла при формировании разрядов частного.
12. Что такое переполнение разрядной сетки частного и как оно определяется.
13. Что такое псевдоразряд Z_0 и какие значения он принимает.
14. Назовите основные принципы построения операционного автомата для выполнения операции деления двоичных чисел.
15. Поясните назначение узлов операционного автомата при выполнении операции деления двоичных чисел с фиксированной запятой (рис. 1).
16. Как определяется знаковый разряд результата при делении чисел с фиксированной запятой.

Библиографический список

1. Карцев М.А. Арифметика цифровых машин. – М.: Наука. 1969. – 575 с.
2. Самофалов К.Г., Романкевич А.М., Валуйский В.Н. Прикладная теория цифровых автоматов. – Киев: Высш. шк., 1987 – 374 с. : ил.
3. Савельев А.Я. Прикладная теория цифровых автоматов. – М.: Высш. шк., 1987. – 271 с: ил.
4. Каган Б.М. Электронные вычислительные машины и системы. – М.: Энергия 1979. - 528 с.
5. Майоров С.А., Новиков Г.И. Принципы организации цифровых машин. – Л.: Машиностроение. 1974. – 431 с.
6. Поспелов Д.А. Арифметические основы вычислительных машин дискретного действия. – М.: Высш. шк. 1970. – 307 с.
7. Соловьев Г.Н. Арифметические устройства ЭВМ. – М.: Энергия. 1978. –177 с.
8. Сергеев Н.П., Вашкевич Н.П. Основы вычислительной техники: Учеб. пособие для электротех. спец. вузов. – М.: Высш. шк., 1988. – 311 с.: ил.
9. Преснухин Л.Н., Нестеров П.В. Цифровые вычислительные машины. – М.: Высш. шк. 1981, - 511 с.