МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

**Лабораторная работа №3**

по дисциплине: Информатика

тема: «Арифметические операции над числами в двоичной системе счисления

(умножение и деление)»

Выполнил: ст. группы ПВ-223

Пахомов Владислав Андреевич

Проверили: ст. пр.

Бондаренко Т. В.

Белгород 2022 г.

**Цель работы:** изучить правила выполнения арифметических операций умножение и деление над числами в двоичной системе счисления.

Вариант № 10 ПВ-223

**Задания к работе:**

1. Выполнить перевод целых чисел E, F, X и вещественных чисел G, H, I и J в двоичную систему счисления «вручную». Действия по переводу расписать в отчете полностью.

2. Выполнить операцию умножения над числами F и X в двоичной системе счисления (F \* X). Результат перевести в десятичную и шестнадцатеричную систему счисления. Выполнить умножение чисел F и X в десятичной системе и сравнить с полученным результатом.

3. Выполнить операцию деления с остатком над числами E и F в двоичной системе счисления (E / F). Представить неполное частное в прямом и обратном коде, и в десятичной системе счисления. Представить остаток от деления в прямом и дополнительном коде, и перевести в десятичную систему счисления. Выполнить деление чисел E и F в десятичной системе счисления и сравнить с полученным результатом.

4. Выполнить над двоичными числами G и H операцию умножения (G \* H). Результат представить в нормализованном виде и в десятичной системе счисления. Выполнить умножение чисел G и H в десятичной системе и сравнить с полученным результатом.

5. Выполнить над двоичными числами I и J операцию деления (I / J) с точностью 0,0001 (до 4 знаков после запятой). Результат представить в нормализованном виде и в десятичной системе счисления. Выполнить деление чисел I и J в десятичной системе и сравнить с полученным результатом. Замечание. Действия над числами в двоичной системе счисления выполнять «в столбик» подробно и с указанием единиц переноса. Умножения и деление для вещественных двоичных чисел можно выполнять в нормализованном виде.

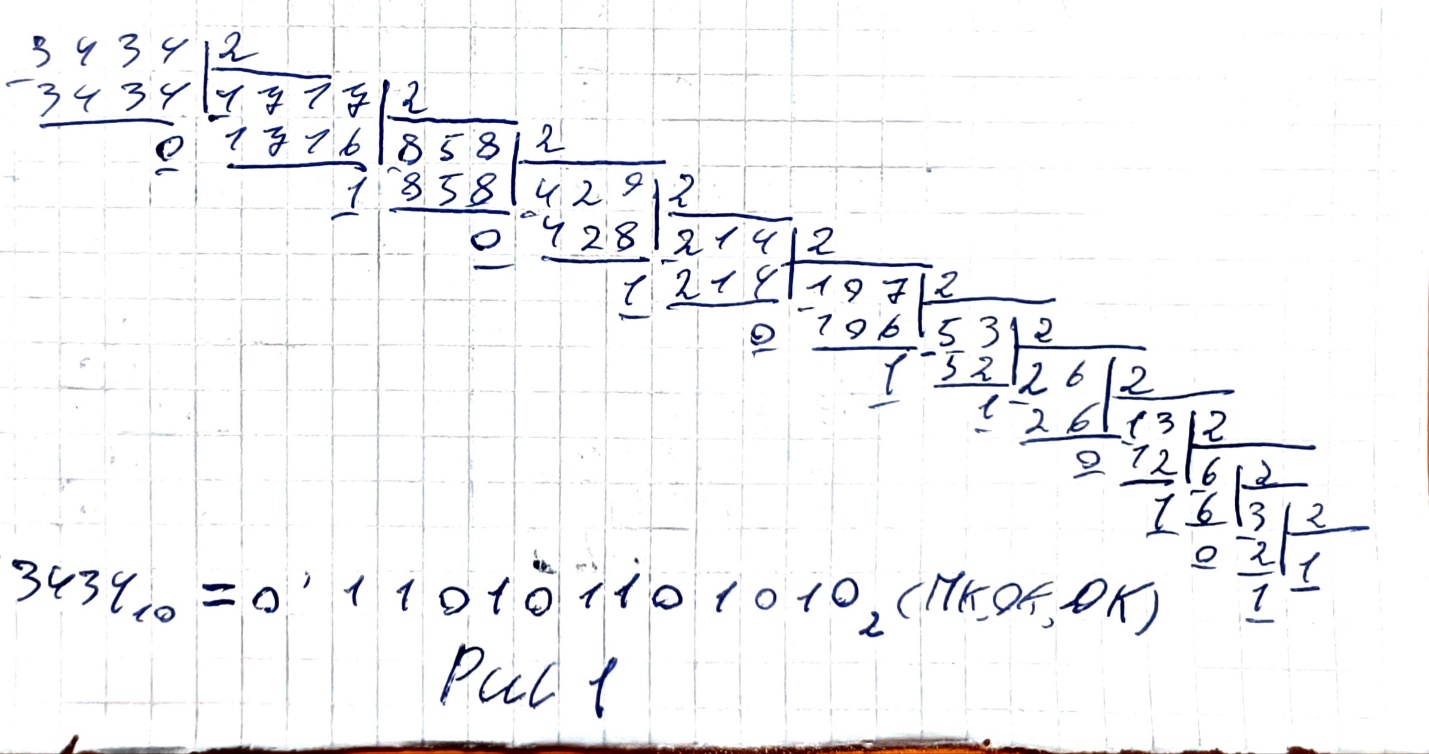
6. Разработать программу, моделирующую выполнение основных арифметических операций: умножение и деление над числами, представленными в двоичной системе счисления.

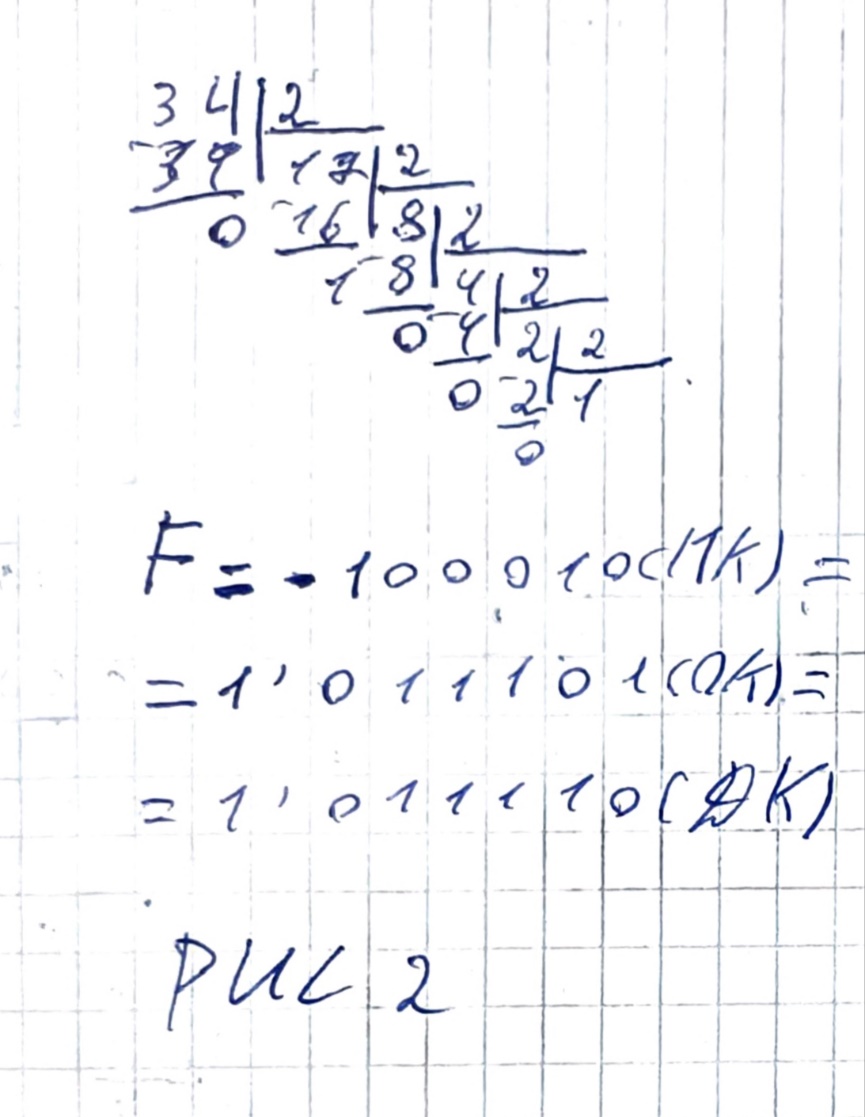




**Задание 1 (E = 3434, F = -34, G = 54,115, H = -33,25, I = -5353,25, J = 322,5,**

**X = 24)**

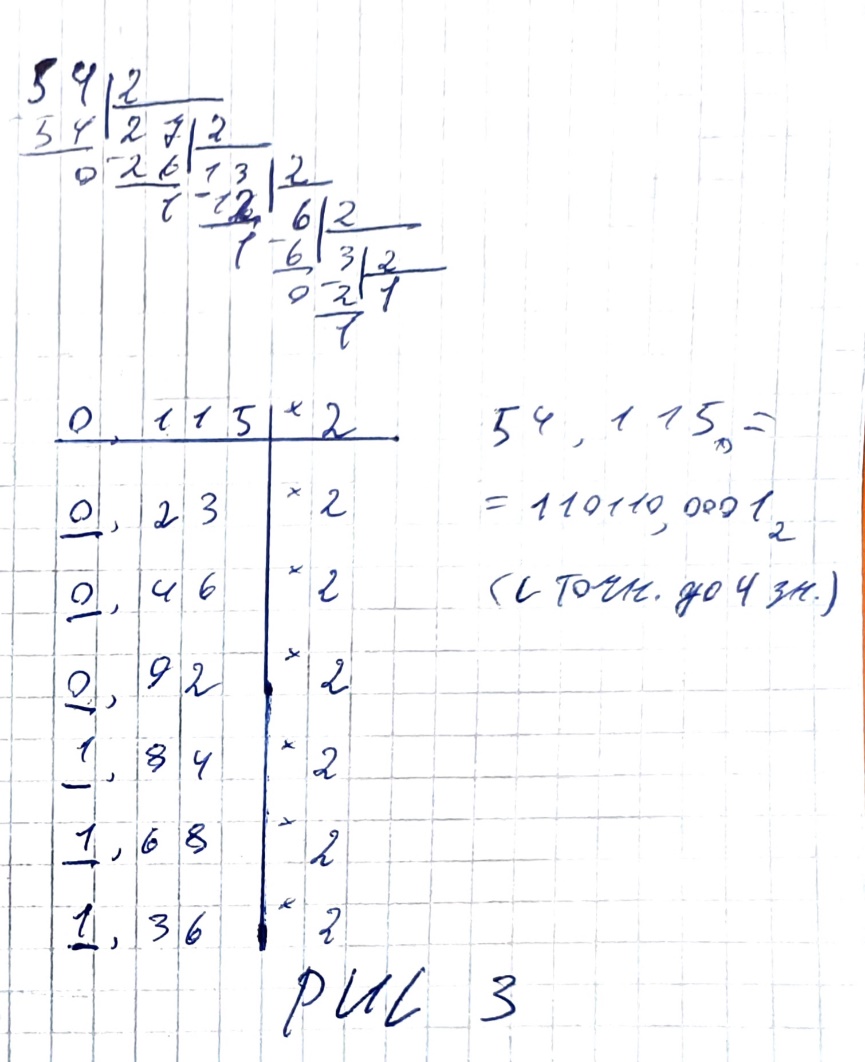
E = 343410 = 0’1101 0110 10102 (ПК, ОК, ДК)



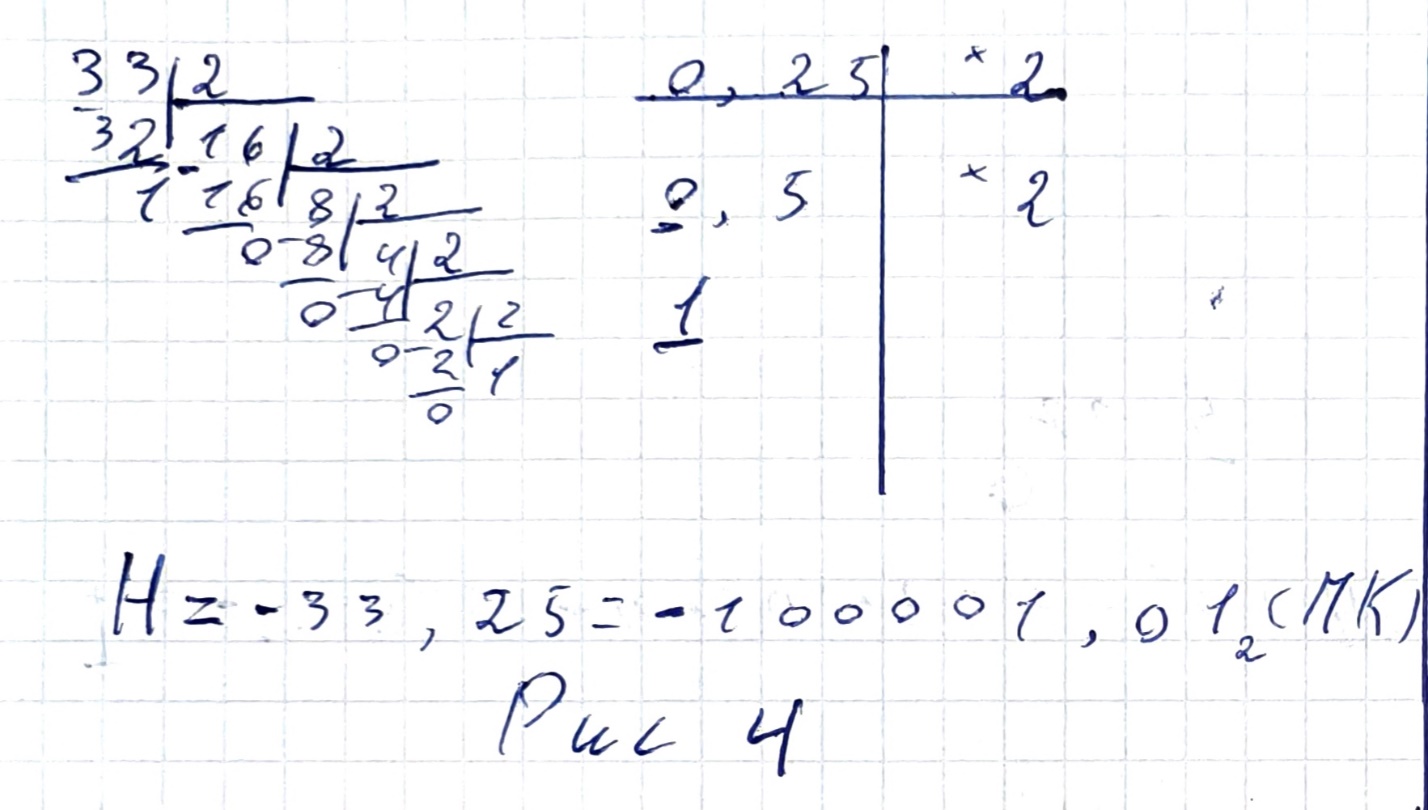
F = -3410 = -1000102 (ПК) =

= 1’10111012 (ОК) =

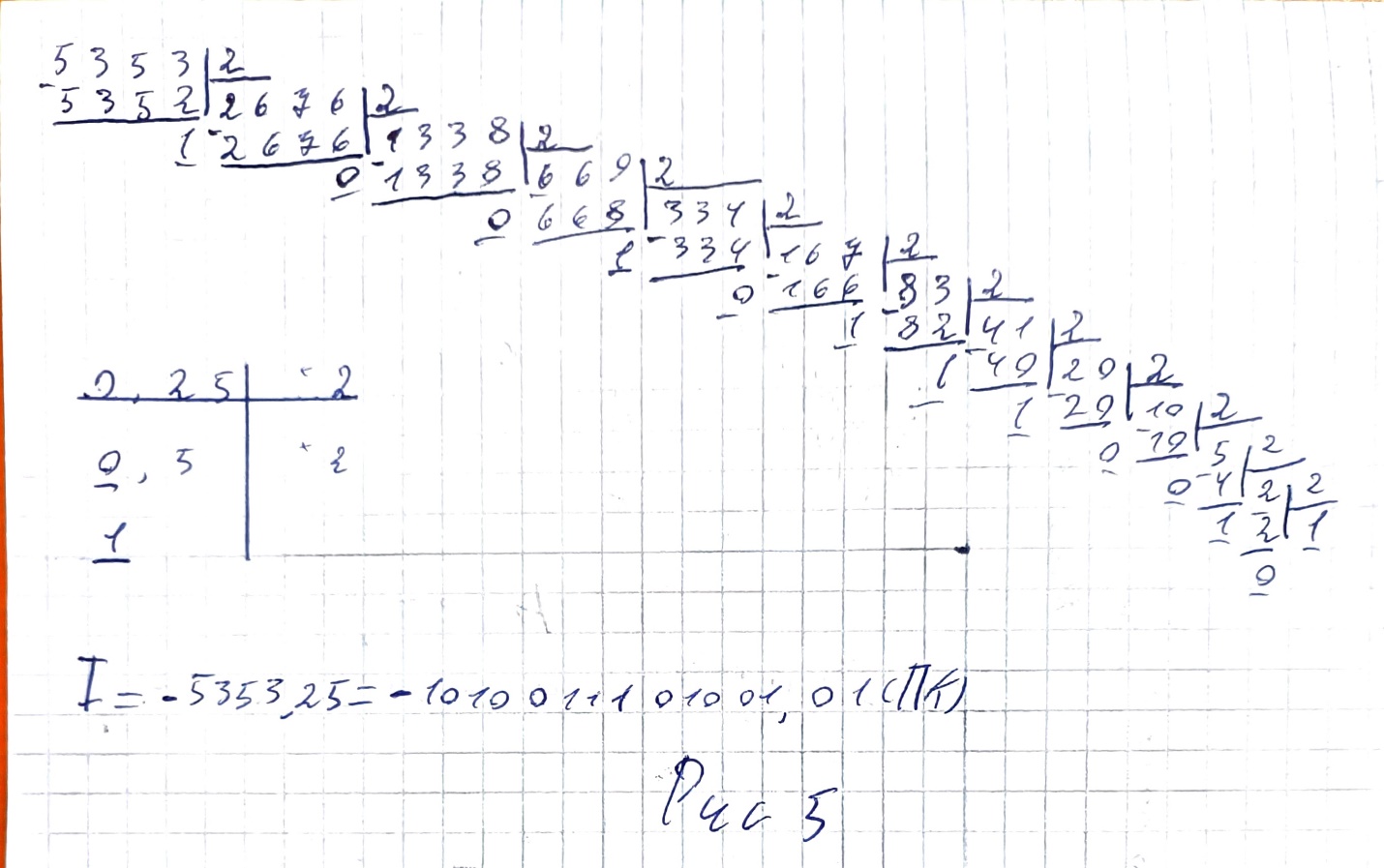
= 1’10111102 (ДК)



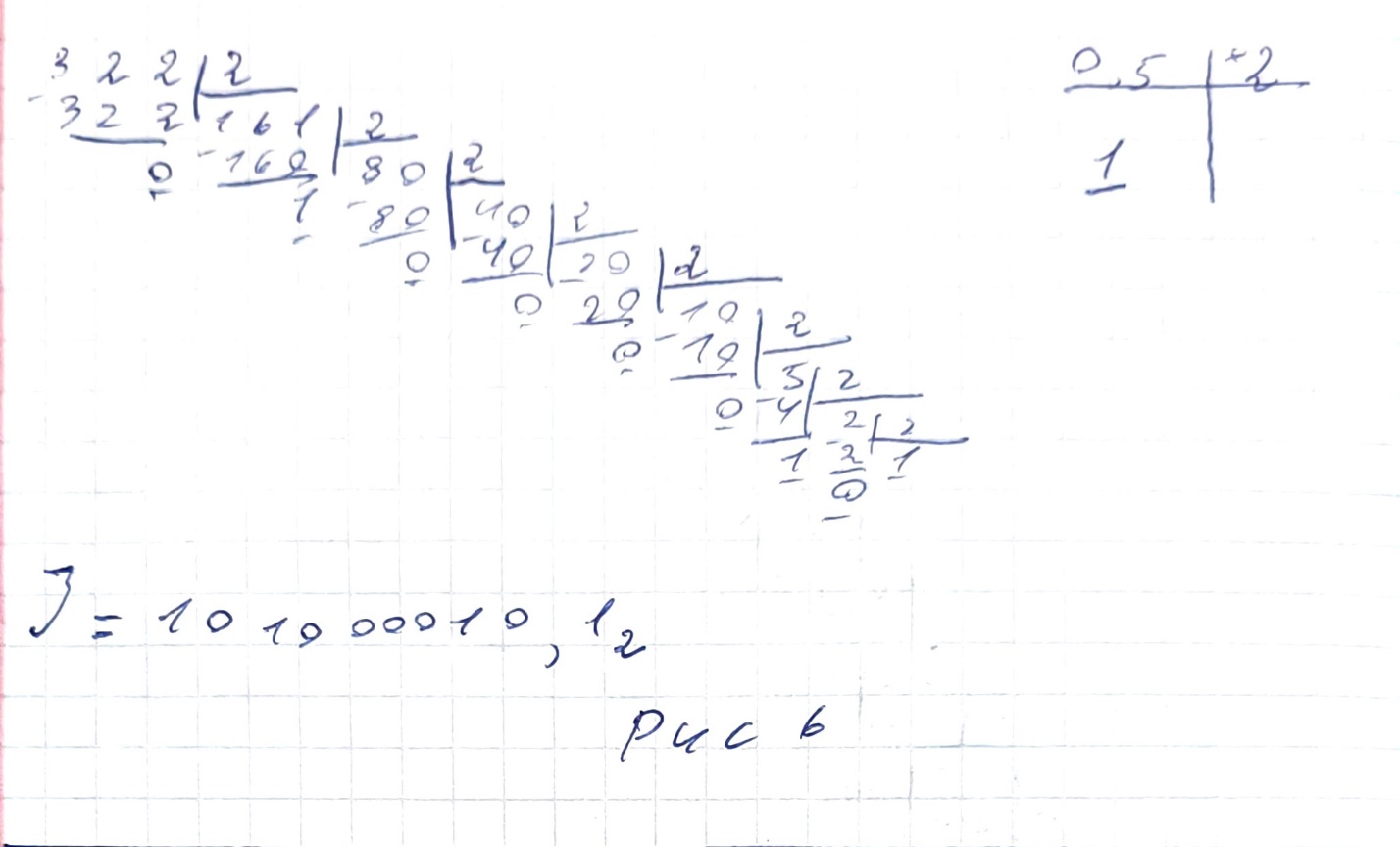
G = 54,11510 = 110110,00012 (с точностью 4 знака после запятой)

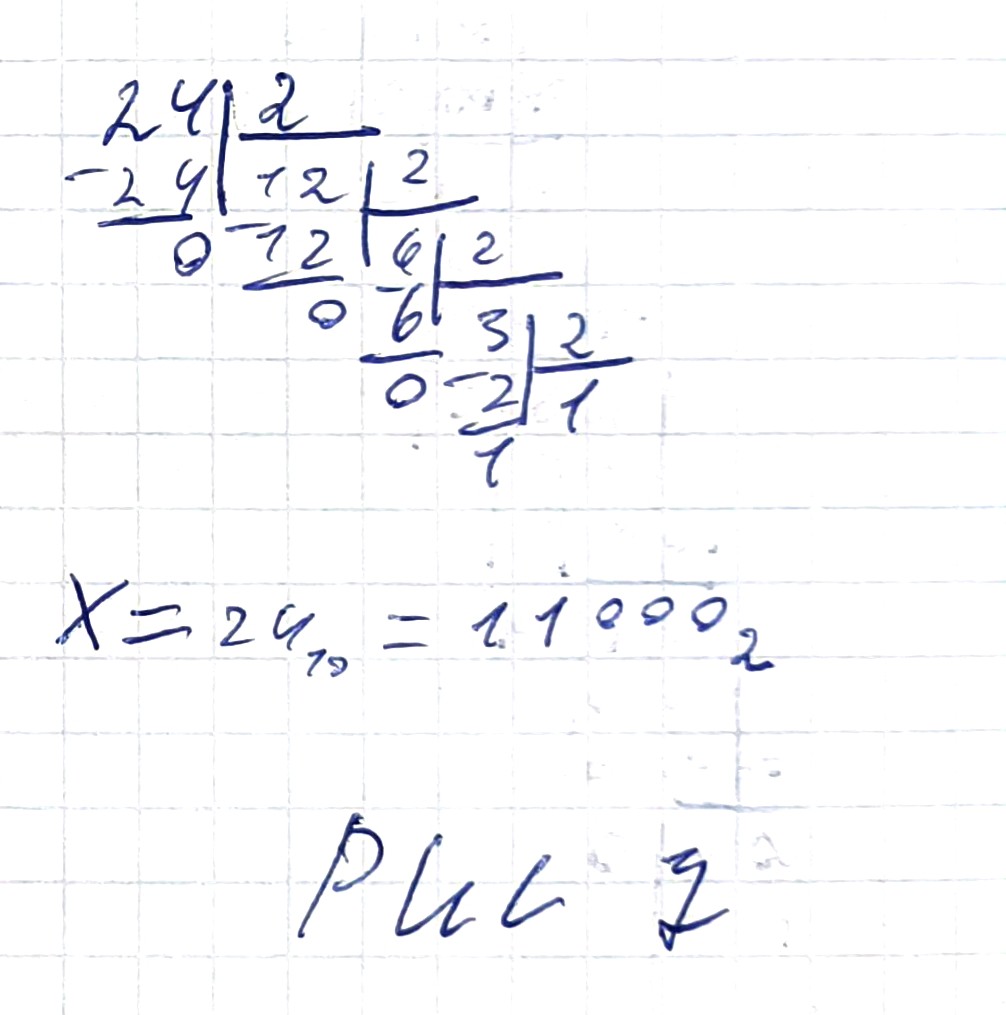


H = -33,2510 = -100001,012 (ПК)



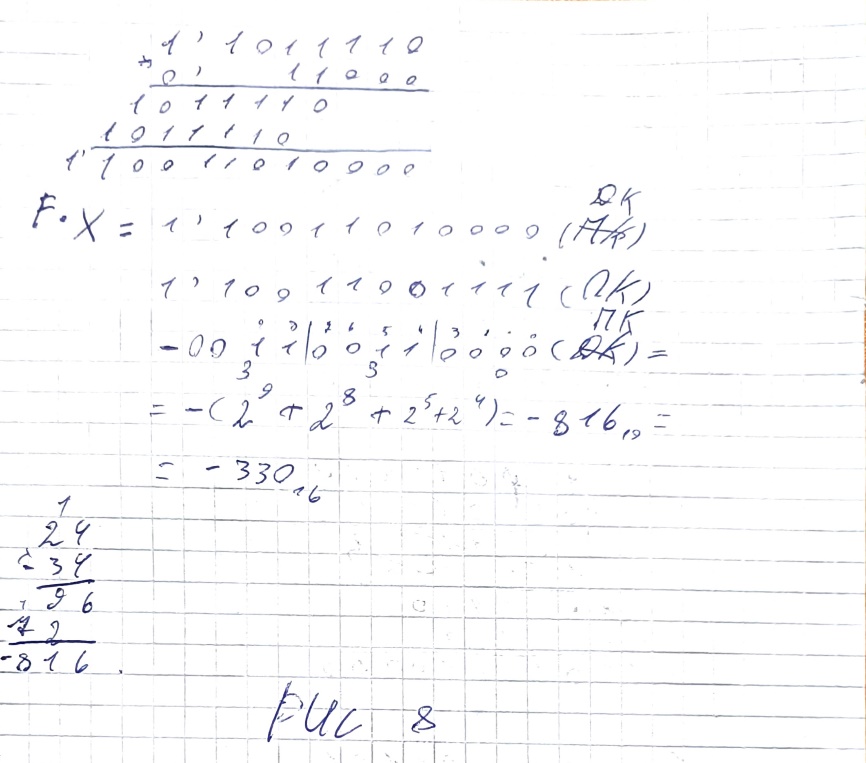
I = -5353,2510 = -1010011101001,012 (ПК)

J = 322,510 = 101000010,12



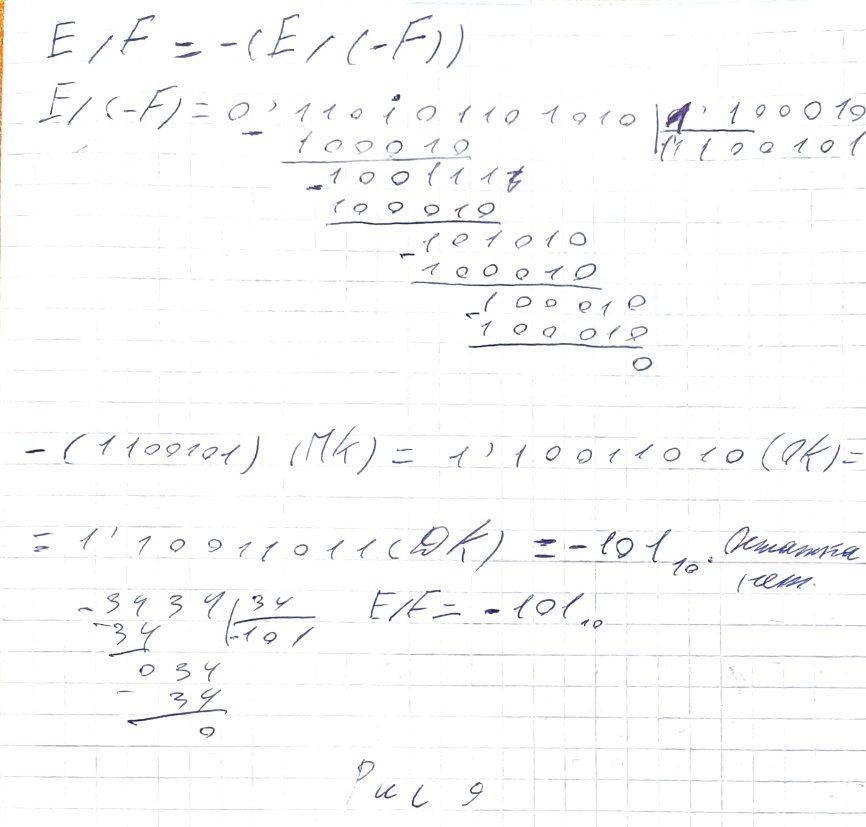
X = 2410 = 0’110002 (ПК, ОК, ДК)

**Задание 2 (F = 1’10 00102 (ПК), X = 0’1 10002)**



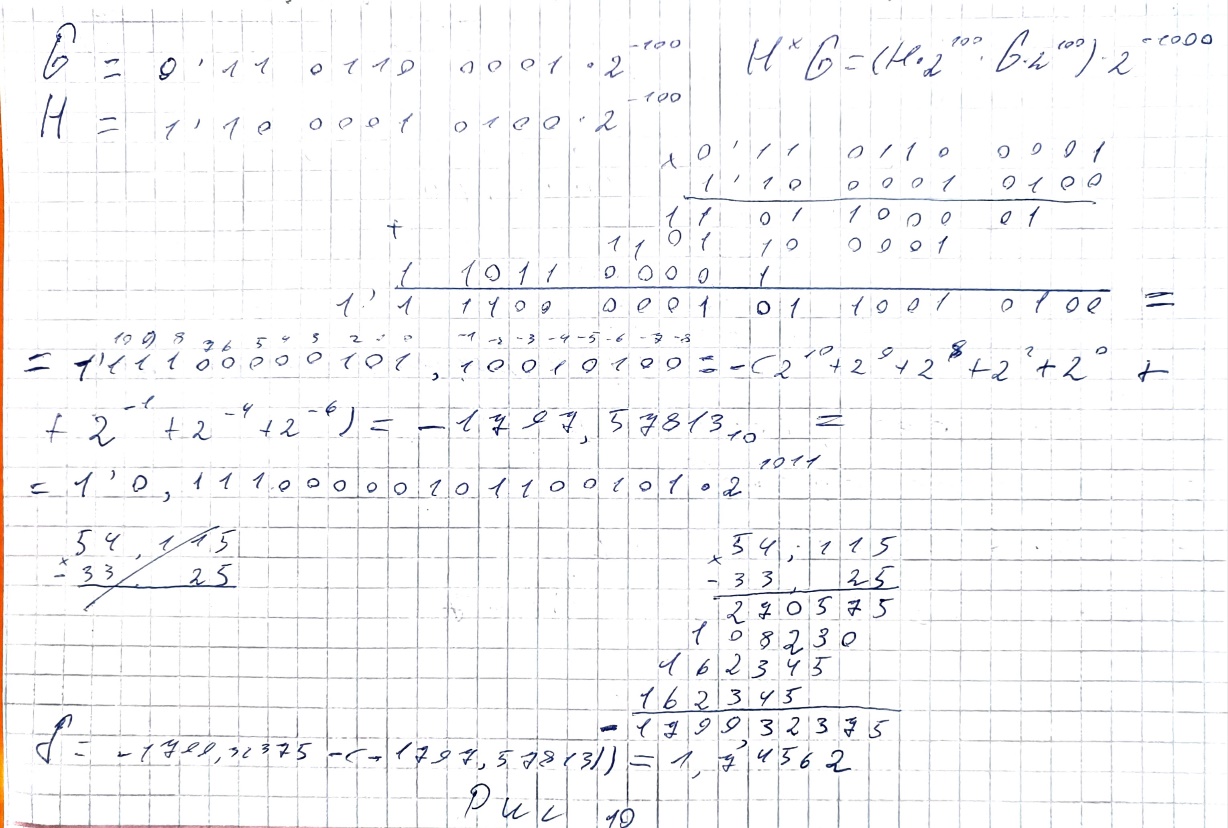
Результаты умножения в двоичной системе и в десятичной системе совпали, умножение выполнено верно.

**Задание 3 (E = 0’1101 0110 10102, F = 1’10 00102 (ПК))**



Результаты деления в двоичной системе и в десятичной системе совпали, деление выполнено верно.

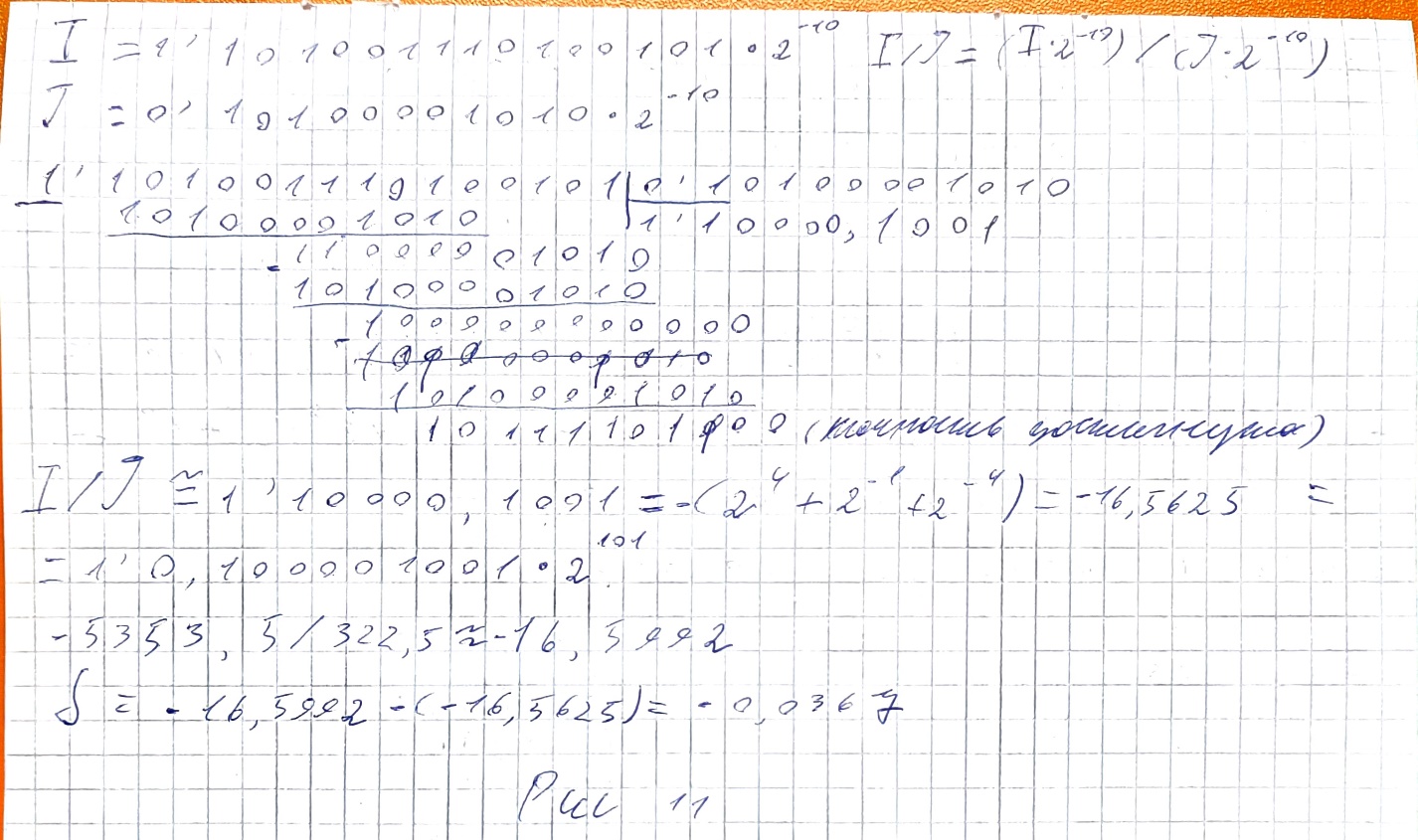
**Задание 4 (G = 0’11 0110,00012, H = 1’10 0001,012 (ПК))**



Результат двоичного умножения не равен десятичному, но близок к нему.

Для повышения точности необходимо продолжить уточнение числа G.

**Задание 5 (I = 1’1 0100 1110 1001,012, J = 0’1 0100 0010,12)**



Результат двоичного деления не равен десятичному, но близок к нему.

Для повышения точности необходимо продолжить деление как десятичное, так и двоичное.

**Задание 6**

|  |  |
| --- | --- |
| Входные данные | Ожидаемый результат |
| A = 1 10002,  B = -428,A\*B | 1’ 011001100002(ПК) =  1’ 100110011112(ОК) =  1’ 100110100002(ДК) =  -14608=-81610=-33016 |
| A = -343410,  B = -2216,A/B | 0’ 11001012(ПК, ОК, ДК) =  1458=10110=6516 |
| A = -5411510,  B = -CFD16,A\*B | 0’ 10101011100110001100110101112(ПК, ОК, ДК) =  12563063278=17993237510=AB98CD716 |
| A = -1504378,  B = 1100 1001 10012,A/B | 1’ 0100002(ПК) =  1’ 1011112(ОК) =  1’ 1100002(ДК) =  -208=-1610=-1016 |

В решении практической части третьей лабораторной работы используется решение второй лабораторной работы, в код добавил функции для умножения и деления, модифицировал функцию compute и вёрстку сайта.

...

function *multiply*(num1, p1, num2, p2) {  
 let sign = num1.startsWith("-") ^ num2.startsWith("-");  
 num1 = *Number*.parseInt(*toOtherP*(num1.startsWith("-") ? num1.substring(1) : num1, p1, 10));  
 num2 = *Number*.parseInt(*toOtherP*(num2.startsWith("-") ? num2.substring(1) : num2, p2, 10));  
 let pow = 0;  
 let resultSum = 0;  
 while (num2 > 0) {  
 if (num2 % 2) {  
 resultSum = *add*(*String*(resultSum), 10, *String*(num1 << pow), 10);  
 }  
  
 pow++;  
 num2 >>= 1;  
 }  
  
 return *Number*.parseInt((sign ? "-" : "") + resultSum);  
}  
  
function *division*(num1, p1, num2, p2) {  
 let sign = num1.startsWith("-") ^ num2.startsWith("-");  
 num1 = *toOtherP*(num1.startsWith("-") ? num1.substring(1) : num1, p1, 2);  
 num2 = *toOtherP*(num2.startsWith("-") ? num2.substring(1) : num2, p2, 10);  
  
 if (*Number*.parseInt(*toOtherP*(num1, 2, 10)) < num2) return *Number*.parseInt((sign ? "-" : "") + "0");  
 if (*Number*.parseInt(*toOtherP*(num1, 2, 10)) === num2) return *Number*.parseInt((sign ? "-" : "") + "1");  
  
 let result = 0;  
 let startPosition = 0;  
 let endPosition = 1;  
 let binPrefix = "";  
 while (endPosition <= num1.length) {  
 while (endPosition <= num1.length && *add*(binPrefix + num1.substring(startPosition, endPosition), 2, "-" + num2, 10) < 0){  
 endPosition++;  
 result <<= 1;  
 }  
 if (*add*(binPrefix + num1.substring(startPosition, endPosition), 2, "-" + num2, 10) >= 0) {  
 result <<= 1;  
 result += 1;  
  
 let subtractionResult = *add*(binPrefix + num1.substring(startPosition, endPosition), 2, "-" + num2, 10);  
 binPrefix = *toOtherP*(*String*(subtractionResult), 10, 2);  
 } else {  
 break;  
 }  
 startPosition = endPosition;  
 endPosition = startPosition + 1;  
 }  
  
 return *Number*.parseInt((sign ? "-" : "") + result);  
}

...

function *compute*(num1, p1, num2, p2, operation) {  
 switch (operation) {  
 case "+":  
 return *add*(num1, p1, num2, p2);  
 break;  
 case "-":  
 if (num2.startsWith("-")) return *add*(num1, p1, num2.substring(1), p2);  
 else return *add*(num1, p1, "-" + num2, p2);  
 break;  
 case "\*":  
 return *multiply*(num1, p1, num2, p2);  
 break;  
 case "/":  
 return *division*(num1, p1, num2, p2);  
 break;  
 default:  
 throw new *Error*("Unknown operation");  
 }  
}

...

<select name="operation" style="margin: 20px">  
 <option value="+">+</option>  
 <option value="-">-</option>  
 <option value="\*">\*</option>  
 <option value="/">/</option>  
</select>

...

Вывод: в ходе лабораторной работы изучили способы перевода чисел из одной позиционной системы счисления в другую; способы представления знаковых чисел в прямом, обратном и дополнительном коде; способы выполнения арифметических операций сложение и вычитание над числами в двоичной системе счисления.