**ЗАВДАННЯ ДЛЯ ЗАХИСТУ АКАДЕМІЧНОЇ РІЗНИЦІ**

**З ДИСЦИПЛІНИ «КОМП’ЮТЕРНА ЛОГІКА»**

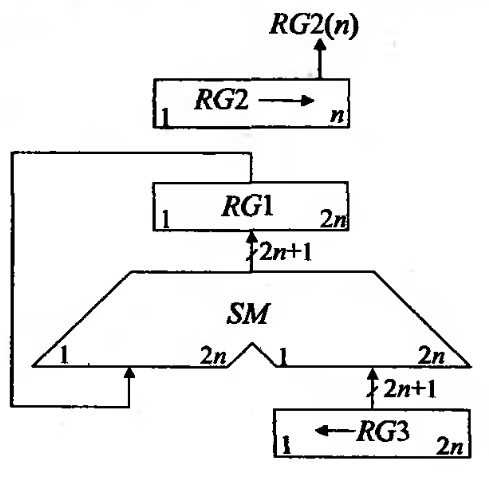
**БІЛЕТ № 18**

**Виконав:**

**Студент ННІКІТ СП-225**

**Крутофіст Олексій Іванович**

**1.** Множення чисел, поданих паралельним кодом другим способом. Принцип множення. Навести операційну схему пристрою для виконання операцій множення чисел другим способом та надати пояснення її функціонування. Розрядність операндів: знак числа – 1 розряд, число 12 - розрядів.



Операційна схема пристрою для виконання операцій множення чисел другим способом

**Пояснення функціонування операційної схеми пристрою для виконання операції множення чисел другим способом:**

Перед початком множення другим способом, множник X записують в регістр RG2,а множене Y – в молодші розряди регістру RG3 ( тобто в регістрі RG3 установлюють Y0 = Y2-n). В кожному і-му циклі множення додавання кодів RG3 і RG1 управляє цифра RG2(n), а в регістрі RG3 здійснюється зсув вліво на один розряд, в результаті чого формується величина Yi = 2Yi-1. Оскільки сума часткових добутків в процесі множення нерухома, зсув в регістрі RG3 можна сполучити в часі з підсумовуванням (як правило, tn > tз). В цьому випадку tm = ntn. Завершення операції множення визначається за нульовим вмістом регістру RG2, що також приводить до збільшення швидкодії, якщо множник ненормалізований.

**2.** Виконати операцію додавання чисел А і В у форматі з плаваючою комою згідно чотирьох етапів. Виконати дію округлення результату. Додавання виконувати у модифікованому доповнюваному коді. У процесі додавання кількість розрядів мантис чисел А і В може бути збільшено до необхідних значень, але результат додавання після округлення повинен бути в межах заданої розрядної сітки: для порядку n-розрядів, і m-розрядів для мантис чисел А і В (без урахування кількості розрядів знака). Результат множення чисел А і В надати у прямому коді. Примітка: перед виконанням операції додавання операнди А і В привести до нормалізованого вигляду.

А = 2-2 \* 1,0110111, В = 22 \* (-0,1001011), n=4, m=8.

Множення A \* B у прямому коді. Порядки додаються, а мантиси перемножуються.

0.1,0110111\*2-2

\*

1.0,1001011\*22

--------------------------------

00.0,11010110011101 \* 20

Додавання у модифікованому доповнюваному коді.

00.1,0110111 – А

+

11.1,0110101 – В

---------------------

00.0,1101100\*20010

Округлення до n=4.

00.0,1101

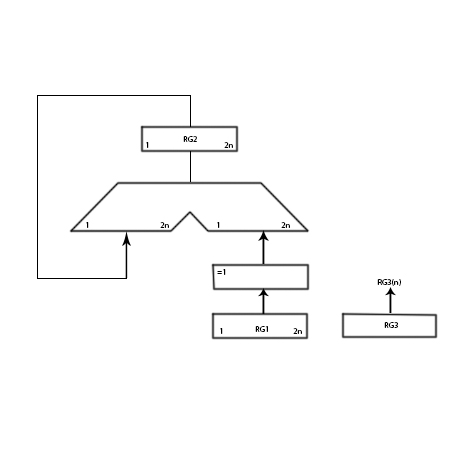
+0

-------------

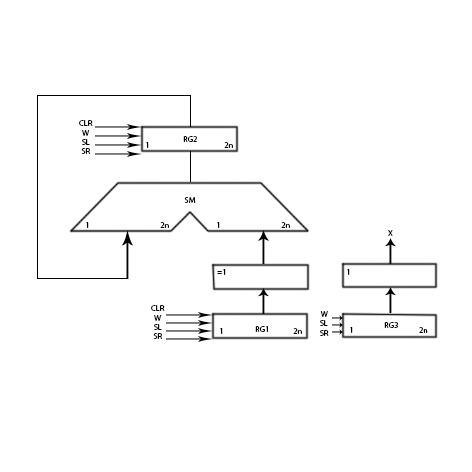
00.0,1101\*20010

**3.** Побудувати функціональну схему пристрою з розподіленою логікою для обчислення функції D. Надати пояснення та обґрунтування функціонального пристрою. Кількість розрядів для кожного з операндів А, В, С дорівнює n без урахування розрядів знака. Операцію множення виконувати другим способом.

D = -8C – 0,125A\*(B-1)



Операційна схема пристрою для обчислення функції



Функціональна схема пристрою для обчислення функції

Пояснення роботи функціонального пристрою:

Для обчислення функції D використовується другий метод множення. Другий метод множення: Множник X записують в регістр RG3,а множене Y – в молодші розряди регістру RG1. В кожному і-му циклі множення додавання кодів RG1 і RG2 управляє цифра RG3(n), а в регістрі R1 здійснюється зсув вліво на один розряд, в результаті чого формується величина Yi = 2Yi-1. Оскільки сума часткових добутків в процесі множення нерухома, зсув в регістрі RG1 можна сполучити в часі з підсумовуванням (як правило, tn > tз). В цьому випадку tm = ntn. Завершення операції множення визначається за нульовим вмістом регістру RG3, що також приводить до збільшення швидкодії, якщо множник ненормалізований. Множене A записують в молодші розряди регістру RG1, до множника B додається +1 і записується в регістр RG3. В кожному циклі множення додаванням кодів управляє цифра RG3(n). Після завершення ітераційного процесу, проводиться здвиг в право на три, таким чином завершується процес розрахунку виразу. Логічний елемент (=1), для перевірки А, якщо А буде від’ємним числом, то цей елемент переведе число в обернений код, а якщо позитивне, то ні чого не буде змінювати. На виході працює логічний елемент (Не), який інвертує значення, що приводить до готового результату.