Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

# Комп’ютерна арифметика

# Розрахункова робота по курсу «КЛ-2»

Виконав: студент групи ІО-82

Давиденко Кирило Юрійович

Керівник: Жабін В.І.

2009 р.

**Обґрунтування варіанту**

Номер залікової книжки: 821010 = 100000000100102

X2 = –10010001,1000110

Y2 = +10100,0011000101

**Виконання роботи**

***Завдання 1***

В прямому коді:

**X2:**

Зн.Р P=+810 Зн.М M= -,100100011000110

### Y2:

Зн.Р P=+510 Зн.М M= +,101000011000101

***Завдання 2***

**1. Операція множення чисел**.

Z=YX, де Y-множене, X-множник.

**1.1 Перший спосіб множення.**

**1.1.1 Теоретичне обгрунтування.**

Числа множаться у прямих кодах, знакові та основні розряди обробляються окремо. Для визначення знака добутку здійснюють підсумування по модулю 2 цифр, що розміщуються в знакових розрядах співмножників.

Множення мантис першим способом здійснюється з молодших розрядів множника, сума часткових добутків зсувається вправо, а множене залишається нерухомим. Тоді добуток двох чисел представляється у вигляді:

що рівносильно

.

Отже, сума часткових добутків в *i*-му циклі, де , зводиться до обчислення виразу:

.

**1.1.2 Операційна схема.**

***RG2(0)***

***RG2***

***RG1***

***n-1***

***0***

***n-1***

***2***

***n+1***

***n***

***n***

***0***

***0***

***SM***

***n+1***

***n+1***

***0***

***RG3***

***n***

***CT***

***0***

***s***

***CT=0***

*Рисунок 1.1- Операційна схема.*

**1.1.3 Змістовний мікроалгоритм.**

Початок

Кінець

RG1:=0

RG2:=X

RG3:=Y

CT:=15

RG1:=RG1+RG3

RG1:=0.r(RG1)

RG2:=RG1[0].r(RG2)

CT:=CT-1

RG2[0]

CT=0

1

1

*Рисунок 1.2- Змістовний мікроалгоритм.*

**1.1.4 Таблиця станів регістрів.**

*Таблиця 1.1-Таблиця станів регістрів.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № ц. | RG1 | RG2 | RG3 | СТ |
| П.С. | 0000000000000000 | 10010001100011**0** | 0101000011000101 | 1111 |
| 1🡪 | 0000000000000000 | 01001000110001**1** |  | 1110 |
| 2  🡪 | +0101000011000101  0101000011000101  0010100001100010 | 10100100011000**1** |  | 1101 |
| 3  🡪 | +0101000011000101  0111100100100111  0011110010010011 | 11010010001100**0** |  | 1100 |
| 4🡪 | 0001111001001001 | 11101001000110**0** |  | 1011 |
| 5🡪 | 0000111100100100 | 11110100100011**0** |  | 1010 |
| 6🡪 | 0000011110010010 | 01111010010001**1** |  | 1001 |
| 7  🡪 | +0101000011000101  0101100001010111  0010110000101011 | 10111101001000**1** |  | 1000 |
| 8  🡪 | +0101000011000101  0111110011110000  0011111001111000 | 01011110100100**0** |  | 0111 |
| 9🡪 | 0001111100111100 | 00101111010010**0** |  | 0110 |
| 10🡪 | 0000111110011110 | 00010111101001**0** |  | 0101 |
| 11🡪 | 0000011111001111 | 00001011110100**1** |  | 0100 |
| 12  🡪 | +0101000011000101  0101100010010100  0010110001001010 | 00000101111010**0** |  | 0011 |
| 13🡪 | 0001011000100101 | 00000010111101**0** |  | 0010 |
| 14🡪 | 0000101100010010 | 10000001011110**1** |  | 0001 |
| 15  🡪 | +0101000011000101  0101101111010111  **0,010110111101011** | **110000001011110** |  | 0000 |

**1.1.5 Функціональна схема з відображенням управляючих сигналів.**

***x1***

***DR2***

***R***

***W1***

***ShR1***

***DR=”0”***

***RG2***

***0***

***n-1***

***RG1***

***0***

***n***

***W2***

***ShR2***

***n***

***n+1***

***x2***

***X***

***n***

***n***

***0***

***0***

***SM***

***n+1***

***n+1***

***CT***

***s***

***WCT***

***dec***

***W3***

***RG3***

***0***

***n***

***n***

*Рисунок 1.3- Функціональна схема.*

***Y***

**1.1.6 Закодований мікроалгоритм.**

*Таблиця 1.2-Таблиця кодування операцій і логічних умов.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблиця кодування мікрооперацій | |  | Таблиця кодування логічних умов | |
| МО | УС |  | ЛУ | Позначення |
| G1:=0  RG2:=X  RG3:=Y  CT:=15  RG1:=RG1+RG3  RG1:=0.r(RG1)  RG2:=RG1[0].r(RG2)  CT:=CT-1 | R  W2  W3  WCT  W1  ShR1  ShR2  dec |  | RG2[0]  CT=0 | X1  X2 |
|
|
| Початок |

Z1

Z2

R, W2, W3, WCT

1

X1

Z3

W1

Z4

X2

ShR1,ShR2,dec

1

Кінець

Z5

*Рисунок 1.4-Закодований мікроалгоритм.*

**1.1.7 Граф управляючого автомата Мура з кодами вершин.**

001

000

Q3Q2Q1

-

Z1

-

Z2

R, W2, W3, WCT

110

X2

Z5

**-**

X1

Z4

ShR1, ShR2,dec

-

Z3

W1

010

011

*Рисунок 1.5-Граф автомата Мура*

**1.1.8 Обробка порядків і нормалізація**

.

Нормалізація мантиси.

MZ= ,0101101111010111 🡨 , .

,101101111010111 .

Знак мантиси: .

**1.1.9 Форма запису нормалізованого результату з плаваючою комою в пам’ять**

Зн.Р P=+1210 Зн.М M

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0. | 0 | 0 | 0 | 1 | 1 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0, | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |

**1.2 Другий спосіб множення**

**1.2.1 Теоретичне обгрунтування**

Числа множаться у прямих кодах, знакові та основні розряди обробляються окремо. Визначення знака добутку здійснюють підсумування по модулю 2 цифр, що розміщуються в знакових розрядах співмножників.

Множення мантис другим способом здійснюється з молодших розрядів, множене зсувається вліво, а сума часткових добутків залишається нерухомою.

Вираз

подамо у вигляді

.

Отже, сума часткових добутків в i-му циклі, де , зводиться до обчислення виразу:

.

з початковими умовами *Z0=0, Y0=Y2-n, i=1.*

**1.2.2 Операційна схема**

***RG2(0)***

***n-1***

***0***

***RG2***

***2n***

***RG1***

***0***

***2n-1***

***2n-1***

***2n-1***

***0***

***0***

***SM***

***2n***

***RG3***

***0***

***2n-1***

*Рисунок 1.6-Операційна схема*

**1.2.3 Змістовний мікроалгоритм**

Початок

RG1:=0

RG2:=X

RG3:=Y

RG2[0]

1

RG1:=RG1+RG3

RG2:=0.r(RG2)

RG3:=l(RG3).0

RG2=0

1

Кінець

*Рисунок 1.7-Змістовний мікроалгоритм*

**1.2.4 Таблиця станів регістрів**

*Таблиця 1.3- Таблиця станів регістрів*

|  |  |  |  |
| --- | --- | --- | --- |
| № ц. | RG1 | RG3 🡨 | RG2 🡪 |
| П.С. | 000000000000000000000000000000 | 000000000000000101000011000101 | 10010001100011**0** |
| 1 | 000000000000000000000000000000 | 000000000000001010000110001010 | 01001000110001**1** |
| 2 | +000000000000001010000110001010  000000000000001010000110001010 | 000000000000010100001100010100 | 00100100011000**1** |
| 3 | +000000000000010100001100010100  000000000000011110010010011110 | 000000000000101000011000101000 | 00010010001100**0** |
| 4 | 000000000000011110010010011110 | 000000000001010000110001010000 | 00001001000110**0** |
| 5 | 000000000000011110010010011110 | 000000000010100001100010100000 | 00000100100011**0** |
| 6 | 000000000000011110010010011110 | 000000000101000011000101000000 | 00000010010001**1** |
| 7 | +000000000101000011000101000000  000000000101100001010111011110 | 000000001010000110001010000000 | 00000001001000**1** |
| 8 | +000000001010000110001010000000  000000001111100111100001011110 | 000000010100001100010100000000 | 00000000100100**0** |
| 9 | 000000001111100111100001011110 | 000000101000011000101000000000 | 00000000010010**0** |
| 10 | 000000001111100111100001011110 | 000001010000110001010000000000 | 00000000001001**0** |
| 11 | 000000001111100111100001011110 | 000010100001100010100000000000 | 00000000000100**1** |
| 12 | +000010100001100010100000000000  000010110001001010000001011110 | 000101000011000101000000000000 | 00000000000010**0** |
| 13 | 000010110001001010000001011110 | 001010000110001010000000000000 | 00000000000001**0** |
| 14 | 000010110001001010000001011110 | 010100001100010100000000000000 | 00000000000000**1** |
| 15 | +010100001100010100000000000000  **010110111101011110000001011110** | 101000011000101000000000000000 | **000000000000000** |

**1.2.5 Функціональна схема з відображенням управляючих сигналів**

***x2***

***x1***

***W2***

***ShR***

***DR=’0’***

***2n-1***

***0***

RG1

RG2

***0***

***n-1***

***R***

***W1***

***n***

***2n***

***X***

***2n-1***

***2n-1***

***0***

***0***

SM

***2n***

***2n***

***W3***

***ShL***

***DL=’0’***

RG3

***0***

***2n-1***

***n***

***n***

***0***

***Y***

*Рисунок 1.8- Функціональна схема*

**1.2.6 Закодований мікроалгоритм**

*Таблиця 1.4 – Таблиця кодування мікрооперацій.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблиця кодування мікрооперацій | |  | Таблиця кодування логічних умов | |
| МО | УС |  | ЛУ | Позначення |
| RG1:=0  RG2:=X  RG3:=Y  RG1:=RG1+RG3  RG2:=0.r(PG2)  RG3:=l(RG3).0 | R  W2  W3  W1  ShR  ShL |  | RG2[0]  RG2=0 | X1  X2 |
|  |

Початок

Кінець

R,W2,W3

W1

ShR, ShL

X1

X2

1

1

Z1

Z2

Z3

Z4

Z5

*Рисунок 1.9- Закодований мікроалгоритм.*

**1.2.7 Граф управляючого автомата Мура з кодами вершин**

Q3Q2Q1

Z1

-

01

00

-

X1

-

Z1

-

Z3

Y4

Z4

Y5,Y6

Z2

Y1,Y2,Y3

001

000

-

Z2

R,W2, W3

110

Z5

**-**

X2

011

010

*Рисунок 1.10- Граф автомата Мура*

**1.2.8 Обробка порядків і нормалізація**

Нормалізація мантиси.

MZ= ,0101101111010111 🡨 ,

,101101111010111

Знак мантиси:

**1.2.9 Форма запису нормалізованого результату з плаваючою комою в пам’ять**

Зн.Р P=+1210 Зн.М M

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0. | 0 | 0 | 0 | 1 | 1 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0, | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |

**1.3 Третій спосіб множення**

**1.3.1 Теоретичне обгрунтування**

Числа множаться у прямих кодах, знакові та основні розряди обробляються окремо. Визначення знака добутку здійснюють підсумування по модулю 2 цифр, що розміщуються в знакових розрядах співмножників.

Множення мантис третім способом здійснюється зі старших розрядів множника, сума часткових добутків і множник зсуваються вліво, а множене нерухоме.

Вираз

*подамо у вигляді*

*.*

Отже, сума часткових добутків в *i*-му циклі, де , зводиться до обчислення виразу:

.

з початковими умовами *Z0=0, i=1.*

**1.3.2 Операційна схема**

***RG2(n-1)***

***RG2***

***0***

***n-1***

***RG1***

***2n***

***0***

***2n+1***

***CT***

***0***

***s***

***CT=0***

***2n***

***2n***

***0***

***0***

***SM***

***n+1***

***2n+1***

***n***

***n-1***

***0***

***RG3***

**0**

*Рисунок 1.11-Операційна схема*

**1.3.3 Змістовний мікроалгоритм**

RG1:=0; RG2:=X;

RG3:=Y; CT:=15

Початок

RG2[n-1]

RG1:=RG1+RG3

RG1:=l(RG1).0

RG2:=l(RG2).0

CT:=CT-1

1

CT=0

Кінець

1

*Рисунок 1.12-Змістовний мікроалгоритм*

**1.3.4 Таблиця станів регістрів**

*Таблиця 1.5- Таблиця станів регістрів*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № ц. | RG1 | RG3 | RG2 | СТ |
| П.С. | 0000000000000000000000000000000 | 101000011000101 | **1**00100011000110 | 1111 |
| 1  🡨 | +0000000000000000101000011000101  0000000000000000101000011000101 |  | **0**01000110001100 | 1110 |
| 0000000000000001010000110001010 |
| 2 | 0000000000000010100001100010100 |  | **0**10001100011000 | 1101 |
| 3 | 0000000000000101000011000101000 |  | **1**00011000110000 | 1100 |
| 4  🡨 | +0000000000000000101000011000101  0000000000000101101011011101101 |  | **0**00110001100000 | 1011 |
| 0000000000001011010110111011010 |
| 5 | 0000000000010110101101110110100 |  | **0**01100011000000 | 1010 |
| 6 | 0000000000101101011011101101000 |  | **0**11000110000000 | 1001 |
| 7 | 0000000001011010110111011010000 |  | **1**10001100000000 | 1000 |
| 8  🡨 | +0000000000000000101000011000101  0000000001011011011111110010101 |  | **1**00011000000000 | 0111 |
| 0000000010110110111111100101010 |
| 9  🡨 | +0000000000000000101000011000101  0000000010110111100111111101111 |  | **0**00110000000000 | 0110 |
| 0000000101101111001111111011110 |
| 10 | 0000001011011110011111110111100 |  | **0**01100000000000 | 0101 |
| 11 | 0000010110111100111111101111000 |  | **0**11000000000000 | 0100 |
| 12 | 0000101101111001111111011110000 |  | **1**10000000000000 | 0011 |
| 13  🡨 | +0000000000000000101000011000101  0000101101111010100111110110101  0001011011110101001111101101010 |  | **1**00000000000000 | 0010 |
| 14  🡨 | +0000000000000000101000011000101  0001011011110101111000000101111 |  | **0**00000000000000 | 0001 |
| 0010110111101011110000001011110 |
| 15 | **,0101101111010111100000010111100** |  | 000000000000000 | **0000** |

**1.3.5 Функціональна схема з відображенням управляючих сигналів**

***2n+1***

***2n+1***

***0***

***0***

***SM***

***n***

***2n+1***

***n+1***

**0**

***RG2***

***0***

***n-1***

***RG1***

***2n***

***0***

***2n+1***

***RG3***

***0***

***n-1***

***R***

***W1***

***ShL1***

***DL=’0’***

***W2***

***SL2***

***DL=’0’***

***W3***

***n***

***X***

***n***

***Y***

***CT***

***s***

***WCT***

***dec***

***x2***

*Рисунок 1.13-Функціональна схема*

**1.3.6 Закодований мікроалгоритм**

*Таблиця 1.6- Таблиця кодування мікрооперацій*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблиця кодування мікрооперацій | |  | Таблиця кодування логічних умов | |
| МО | УС |  | ЛУ | Позначення |
| RG1:=0  RG2:=X  RG3:=Y  CT:=15  RG1:=RG1+RG3  RG1:=l(RG1).0  RG2:=l(RG2).0  CT:=CT-1 | R  W2  W3  WCT  W1  ShL1  ShL2  dec |  | RG2[n-1]  CT=0 | X1  X2 |
| Початок  Кінець  R,W2,W3,WCT  W1  ShL1,ShL2,dec  X1  X2  1  1  Z1  Z2  Z3  Z4  Z5 |

*Рисунок 1.14- Закодований мікроалгоритм*

**1.3.7 Граф управляючого автомата Мура з кодами вершин**

Z1

-

Q1Q2

01

00

-

X1

-

Z1

-

Z3

W1

Z2

Y1,Y2,Y3

001

000

Q3Q2Q1

-

Z2

R,W2, W3,WCT

110

Z5

**-**

X2

Z4

ShL1, ShL2,dec

010

011

*Рисунок 1.15- Граф автомата Мура*

**1.3.8 Обробка порядків і нормалізація**

.

Нормалізація мантиси.

MZ= ,0101101111010111 🡨 , ,

,101101111010111 .

Знак мантиси: .

**1.3.9 Форма запису нормалізованого результату з плаваючою комою в пам’ять**

Зн.Р P=+1210 Зн.М M

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0. | 0 | 0 | 0 | 1 | 1 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0, | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |

**1.4 Четвертий спосіб множення**

**1.4.1 Теоретичне обґрунтування**

Числа множаться у прямих кодах, знакові та основні розряди обробляються окремо. Визначення знака добутку здійснюють підсумування по модулю 2 цифр, що розміщуються в знакових розрядах співмножників.

Множення здійснюється зі старших розрядів множника, сума часткових добутків залишається нерухомою, множене зсувається праворуч, множник ліворуч.

Вираз

*подамо у вигляді*

*.*

Отже, сума часткових добутків в *i*-му циклі, де , зводиться до обчислення виразу:

.

з початковими значеннями i=1, Y0=2-1Y, Z0=0.

**1.4.2 Операційна схема**

***RG1***

***0***

***2n-1***

***2n-1***

***2n-1***

***0***

***0***

***SM***

***2n***

***2n***

***RG2(n-1)***

***RG2***

***0***

***n-1***

***2n***

***RG3***

***0***

***2n-1***

*Рисунок 1.16-Операційна схема*

**1.4.3 Змістовний мікроалгоритм**

Початок

RG1:=0

RG2:=X

RG3:=Y

RG3:=0.r(RG3)

RG2[n-1]

1

RG1:=RG1+RG3

RG3:=0.r(RG3)

RG2:=l(RG2).0

RG2=0

1

Кінець

*Рисунок 1.17-Змістовний мікроалгоритм*

**1.4.4 Таблиця станів регістрів**

*Таблиця 1.7- Таблиця станів регістрів*

|  |  |  |  |
| --- | --- | --- | --- |
| № ц. | RG1 | RG3 | RG2 |
| П.С. | 000000000000000000000000000000 | 010100001100010100000000000000 | **1**00100011000110 |
| 1 | +010100001100010100000000000000  010100001100010100000000000000 | 001010000110001010000000000000 | **0**01000110001100 |
| 2 | 010100001100010100000000000000 | 000101000011000101000000000000 | **0**10001100011000 |
| 3 | 010100001100010100000000000000 | 000010100001100010100000000000 | **1**00011000110000 |
| 4 | +000010100001100010100000000000  010110101101110110100000000000 | 000001010000110001010000000000 | **0**00110001100000 |
| 5 | 010110101101110110100000000000 | 000000101000011000101000000000 | **0**01100011000000 |
| 6 | 010110101101110110100000000000 | 000000010100001100010100000000 | **0**11000110000000 |
| 7 | 010110101101110110100000000000 | 000000001010000110001010000000 | **1**10001100000000 |
| 8 | +000000001010000110001010000000  010110110111111100101010000000 | 000000000101000011000101000000 | **1**00011000000000 |
| 9 | +000000000101000011000101000000  010110111100111111101111000000 | 000000000010100001100010100000 | **0**00110000000000 |
| 10 | 010110111100111111101111000000 | 000000000001010000110001010000 | **0**01100000000000 |
| 11 | 010110111100111111101111000000 | 000000000000101000011000101000 | **0**11000000000000 |
| 12 | 010110111100111111101111000000 | 000000000000010100001100010100 | **1**10000000000000 |
| 13 | +000000000000010100001100010100  010110111101010011111011010100 | 000000000000001010000110001010 | **1**00000000000000 |
| 14 | +000000000000001010000110001010  **,010110111101011110000001011110** | 000000000000000101000011000101 | **000000000000000** |

**1.4.5 Функціональна схема з відображенням управляючих сигналів**

***RG1***

***0***

***2n-1***

***2n-1***

***2n-1***

***0***

***0***

***SM***

***2n***

***2n+1***

***RG2(n-1)***

***RG2***

***0***

***n-1***

***2n***

***RG3***

***0***

***2n-1***

***CLR***

***W3***

***SL***

***W1***

***W2***

***SR***

*Рисунок 1.18-Функціональна схема*

**1.4.6 Закодований мікроалгоритм**

*Таблиця 1.8- Таблиця кодування мікрооперацій*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблиця кодування мікрооперацій | |  | Таблиця кодування логічних умов | |
| МО | УС |  | ЛУ | Позначення |
| RG1:=0  RG2:=X  RG3:=Y  RG1:=RG1+RG3  RG3:=0.r(RG3)  RG2:=l(RG2).0 | R  W2  W3  W1  ShR  ShL |  | RG2[n-1]  RG2=0 | X1  X2 |
|  |

Z1

Початок

R, W2, W3, ShR

Z2

X1

Z3

1

W1

ShR,ShL

Z4

X2

1

Z5

Кінець

*Рисунок 1.19- Закодований мікроалгоритм*

**1.4.7 Граф управляючого автомата Мура з кодами вершин**

Z1

-

Q1Q2

01

00

-

X1

-

Z1

-

Z3

W1

Z4

ShR,ShL

Z2

Y1,Y2,Y3

001

000

Q3Q2Q1

-

Z2

R, W2, W3, ShR

110

Z5

**-**

X2

010

011

*Рисунок 1.20- Граф автомата Мура*

**1.4.8 Обробка порядків і нормалізація**

.

Нормалізація мантиси.

MZ= ,0101101111010111 🡨 , ;

,101101111010111 .

Знак мантиси: .

**1.4.9 Форма запису нормалізованого результату з плаваючою комою в пам’ять**

Зн.Р P=+1210 Зн.М M

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0. | 0 | 0 | 0 | 1 | 1 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0, | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |

**2. Операція ділення чисел**

Z=X: Y, X-ділене, Y – дільник.

**2.1 Перший спосіб ділення (з відновленням від’ємного залишку)**

**2.1.1 Теоретичне обґрунтування**

Нехай ділене Х і дільник Y є n-розрядними правильними дробами, поданими в прямому коді. В цьому випадку знакові й основні розряди операндів обробляються окремо. Знак результату визначається шляхом підсумовування по модулю 2 цифр, записаних в знакових розрядах.

При реалізації ділення за першим методом здійснюється зсув вліво залишку при нерухомому дільнику. Черговий залишок формується в регістрі Р2 (у вихідному стані в цьому регістрі записаний Х). Виходи Р2 підключені до входів СМ безпосередньо, тобто ланцюги видачі коду з Р2 не потрібні. Час для підключення n+1 цифри частки визначається виразом t=(n+1)(tt+tc), де tt - тривалість виконання мікрооперації додавання-віднімання; tc - тривалість виконання мікрооперації зсуву.

**2.1.2 Операційна схема**

***RG3***

***n***

***0***

***RG2***

***n+1***

***0***

***n***

***SM***

***n+1***

***n+1***

***n+1***

***0***

***0***

***n+1***

***0***

***RG1***

*Рисунок 2.1-Операційна схема*

**2.1.3 Змістовний мікроалгоритм**

Початок

RG3:=0

RG2:=X;

RG1:=Y;

RG3:=l(RG3).

RG2:=l(RG2).0

1

RG2[n+1]

RG2:=RG2++1

RG2:=RG2+RG1

RG3[n+1]

1

Кінець

*Рисунок 2.2-Змістовний мікроалгоритм*

**2.1.4 Таблиця станів регістрів**

*Таблиця 2.1- Таблиця станів регістрів*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № ц. | RG3 | RG2 | RG1 | МО |
| П.С. | 0000000000000000 | 00100100011000110 | 00101000011000101пк  11010111100111011дк | ПС |
| 1 | 000000000000000**1** | **0**1001000110001100  +11010111100111011  00100000011000111 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 2 | 00000000000000**1**1 | **0**1000000110001110  +11010111100111011  00011000011001001 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 3 | 0000000000000**1**11 | **0**0110000110010010  +11010111100111011  00001000011001101 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 4 | 000000000000**1**111 | **0**0010000110011010  +11010111100111011  11101000011010101 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 5 | 00000000000**1**1110 | **1**1010000110101010  +00101000011000101  11111001001101111 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 6 | 0000000000**1**11100 | **1**1110010011011110  +00101000011000101  00011010110100011 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 7 | 000000000**1**111001 | **0**0,110101101000110  +11,010111100111011  00,001101010000010 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 8 | 00000000**1**1110011 | **0**0,011010100000100  +11,010111100111011  11,110010000111111 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 9 | 0000000**1**11100110 | **1**1,100100001111110  +00,101000011000101  00,001100101000011 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 10 | 000000**1**111001101 | **0**0,011001010000110  +11,010111100111011  11,110000111000001 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 11 | 00000**1**1110011010 | **1**1,100001110000010  +00,101000011000101  00,001010001000111 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 12 | 0000**1**11100110101 | **0**0,010100010001110  +11,010111100111011  11,101011111001001 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 13 | 000**1**111001101010 | **1**1,010111110010010  +00,101000011000101  00,000000001010111 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 14 | 00**1**1110011010101 | **0**0,000000010101110  +11,010111100111011  11,010111111101001 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 15 | 0**1**11100110101010 | **1**0,101111111010010  +00,101000011000101  11,011000010010111 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |
| 16 | **1**,**111001101010100** | **1**0,110000100101110  +00,101000011000101  11,011000111110011 |  | 🡨 RG2🡨RG3  RG2:=RG2-RG1 |

**2.1.5 Функціональна схема з відображенням управляючих сигналів**

***RG3***

***n+1***

***0***

***RZ***

***1***

***n+1***

***DL***

***W3***

***ShL1***

***0***

***n+1***

***RG2***

***1***

***n+2***

***RX***

***n+2***

***W2***

***W4***

***ShL2***

***W5***

***DL=’0’***

***SM***

***n+1***

***n+1***

***0***

***0***

***RG1***

***0***

***n+1***

***W1***

*Рисунок 2.3-Функціональна схема*

**2.1.6 Закодований мікроалгоритм**

*Таблиця 2.2- Таблиця кодування мікрооперацій*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Таблиця кодування мікрооперацій | |  | Таблиця кодування логічних умов | | |
| МО | УС |  | ЛУ | | Позначення |
| RG3:=0  RG2:=X;  RG1:=Y;  RG3:=l(RG3).  RG2:=l(RG2).0  RG2:=RG2++1  RG2:=RG2+RG1 | W3  W2  W1  ShL1  ShL2  W4  W5 |  | RG2[n+1]  RG3[n+1] | | X1  X2 |
| Z1  Початок | |

W3, W2, W1

Z2

Z3

ShL1, ShL2

1

X1

Z4

W4

Z5

W5

X2

Z6

Кінець

1

*Рисунок 2.4- Закодований мікроалгоритм*

**2.1.7 Граф управляючого автомата Мура з кодами вершин**

Q3Q2Q1

000

Z1

-

-

001

111

Z6

-

Z2

Y1,Y2,Y3

X2

-

Z5

**-**

Z3

Y4

X2

Z4

Y5,Y6

110

011

010

*Рисунок 2.5- Граф автомата Мура*

**2.1.8 Обробка порядків і нормалізація**

.

Нормалізація мантиси не потрібна.

MZ= ,111001101010100.

Знак мантиси: .

**2.1.9 Форма запису нормалізованого результату з плаваючою комою в пам’ять**

Зн.Р P=+1210 Зн.М M

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0. | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1, | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |

**2.2 Другий спосіб ділення(без відновлення остачі)**

**2.2.1 Теоретичне обґрунтування**

Нехай ділене Х і дільник Y є n-розрядними правильними дробами, поданими в прямому коді. В цьому випадку знакові й основні розряди операндів обробляються окремо. Знак результату визначається шляхом підсумовування по модулю 2 цифр, записаних в знакових розрядах.

Остача нерухома, дільник зсувається праворуч. Як і при множенні з нерухомою сумою часткових добутків можна водночас виконувати підсумування і віднімання, зсув в регістрах Y,Z. Тобто 1 цикл може складатися з 1 такту, це дає прискорення відносно 1-го способу.

**2.2.2 Операційна схема**

***0***

***n***

***RG3***

***2n***

***0***

***RG2***

***p***

***SM***

***2n***

***0***

***2n***

***0***

***2n***

***0***

***RG1***

*Рисунок 2.6-Операційна схема*

**2.2.3 Змістовний мікроалгоритм**

Початок

Кінець

RG2:=RG2+RG1

RG1:=0.r(RG1)

RG3:=l(RG3).SM(p)

RG2:=RG2++1

RG1:=0.r(RG1)

RG3:=l(RG3).SM(p)

RG2[2n+1]

RG3[n]

1

1

RG3:=0;

RG1:=Y;

RG2:=X

*Рисунок 2.7-Змістовний мікроалгоритм*

**2.2.4 Таблиця станів регістрів**

*Таблиця 2.3- Таблиця станів регістрів*

|  |  |  |  |
| --- | --- | --- | --- |
| № ц. | RG3 | RG2 | RG1 |
| П.С. | 000000000000000**1** | **0**,10010001100011000000000000000 | 00,1010000110001010000000000000пк  11,0101111001110110000000000000дк |
| 1 | 00000000000000**1**1 | +110101111001110110000000000000  **0**01000000110001110000000000000 | 000101000011000101000000000000 |
| 2 | 0000000000000**1**11 | +111010111100111011000000000000  **0**00011000011001001000000000000 | 000010100001100010100000000000 |
| 3 | 000000000000**1**111 | +111101011110011101100000000000  **0**00000100001100110100000000000 | 000001010000110001010000000000 |
| 4 | 00000000000**1**1110 | +111110101111001110110000000000  **1**11111010000110101010000000000 | 000000101000011000101000000000 |
| 5 | 0000000000**1**11100 | +000000101000011000101000000000  **1**11111111001001101111000000000 | 000000010100001100010100000000 |
| 6 | 000000000**1**111001 | +000000010100001100010100000000  **0**00000001101011010001100000000 | 000000001010000110001010000000 |
| 7 | 00000000**1**1110011 | +111111110101111001110110000000  **0**00000000011010100000010000000 | 000000000101000011000101000000 |
| 8 | 0000000**1**11100110 | +111111111010111100111011000000  **1**11111111110010000111101000000 | 000000000010100001100010100000 |
| 9 | 000000**1**111001101 | +000000000010100001100010100000  **0**00000000000110010011111100000 | 000000000001010000110001010000 |
| 10 | 00000**1**1110011010 | +111111111110101111001110110000  **1**11111111111100001101110010000 | 000000000000101000011000101000 |
| 11 | 0000**1**11100110101 | +000000000000101000011000101000  **0**00000000000001010000110111000 | 000000000000010100001100010100 |
| 12 | 000**1**111001101010 | +111111111111101011110011101100  **1**11111111111110101111010100100 | 000000000000001010000110001010 |
| 13 | 00**1**1110011010101 | +000000000000001010000110001010  **0**00000000000000000000000101110 | 000000000000000101000011000101 |
| 14 | 0**1**11100110101010 | +111111111111111010111100111011  **1**11111111111111010111101101001 | 000000000000000010100001100010 |
| 15 | **1,111001101010100** | +000000000000000010100001100010  111111111111111101011111001011 | 000000000000000001010000110001 |

**2.2.5 Функціональна схема з відображенням управляючих сигналів**

***W2***

***W3***

***W4***

***R***

***ShL***

***n***

***RG3***

***2n***

***0***

***RG2***

***2n***

***0***

***RG1***

***SM***

***0***

***2n***

***0***

***2n***

***SM***

***1***

***0***

***DL=p***

***W1***

***ShR***

***DR=’0’***

*Рисунок 2.8-Функціональна схема*

**2.2.6 Закодований мікроалгоритм**

*Таблиця 2.4- Таблиця кодування мікрооперацій*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблиця кодування мікрооперацій | |  | Таблиця кодування логічних умов | |
| МО | УС |  | ЛУ | Позначення |
| RG3:=0;  RG1:=Y;  RG2:=X  RG2:=RG2+RG1  RG1:=0.r(RG1)  RG3:=l(RG3).SM(p)  RG2:=RG2++1 | R  W1  W2  W3  ShR  ShL  W4 |  | RG2[2n+1]  RG3[n] | X1  X2 |
|  |

Початок

Z1

Z2

R,W1,W2

1

X1

Z4

Z3

W4,ShR,ShL

W3,ShR,ShL

X2

1

Z5

Кінець

*Рисунок 2.9- Закодований мікроалгоритм*

**2.2.7 Граф управляючого автомата Мура з кодами вершин**

000

Z1

-

Z4

W4,ShR,ShL

Z3

W3,ShR,ShL

Z2

R,W1,W2

-

110

Q3Q2Q1

010

011

001

X2

X2

Z5

**-**

*Рисунок 2.10- Граф автомата Мура*

**2.2.8 Обробка порядків і нормалізація**

Нормалізація мантиси не потрібна.

MZ= ,111001101010100

Знак мантиси:

**2.2.9 Форма запису нормалізованого результату з плаваючою комою в пам’ять**

Зн.Р P=+310 Зн.М M

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0. | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1, | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |

**3. Операція додавання чисел**

**Z=X+Y.**

**3.1 Теоретичне обґрунтування способу**

В пам’яті числа зберігаються у ПК. На першому етапі додавання чисел з плаваючою комою виконують вирівнювання порядків до числа із старшим порядком. На другому етапі виконують додавання мантис. Додавання мантис виконується у доповню вальних кодах, при необхідності числа у ДК переводяться в АЛП. Додавання виконується порозрядно на n-розрядному суматорі з переносом. Останній етап – нормалізація результату. Виконується за допомогою зсуву мантиси результату і коригування порядку результату. Порушення нормалізації можливо вліво і вправо, на 1 розряд вліво і на n розрядів вправо.

**Виконання етапів вирівнювання порядків і додавання мантис:**

1. Порівняння порядків.

,

.

1. Вирівнювання порядків.

Робимо зсув вправо мантиси числа Y, зменшуючи на кожному кроці, доки стане 0.

*Таблиця 3.1- Таблиця зсуву мантиси на етапі вирівнювання порядків*

|  |  |  |
| --- | --- | --- |
|  |  | Мікрооперація |
| 0.101000011000101 | 11 | П.С. |
| 0.010100001100010 | 10 | 🡪 |
| 0.001010000110001 | 01 | 🡪 |
| 0.000101000011000 | 00 | 🡪 |

1. Додавання мантис у модифікованому ДК.

Таблиця 3.2-Додавання мантис

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1, | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 0 | 0, | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 1, | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |

1. Нормалізація результату (В ПК).

**3.2 Операційна схема**

m-кількість розрядів мантиси

n-кількість розрядів порядку

q=]log2m[

R

L

*КС*

CT=0

m+1 RGZ 0

q CT 1

n+1 RGPZ 0

Рисунок 3.1-Операційна схема

Виконаємо синтез КС для визначення порушення нормалізації.

*Таблиця 3.3-Визначення порушення нормалізації*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Розряди регістру RGZ | | | Значення функцій | |
| Z’0 | Z0 | Z1 | L | R |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 |

L= Z0, R=.

Результат беремо по модулю, знак встановлюємо за Z’0 до нормалізації.

**3.3 Змістовний мікроалгоритм**

Початок

CT:=m; RGZ:=Z;

1

0

RGZ:=RGZ(m+2).r(RGZ)

RGPZ:=RGPZ+1

1

RGZ:=l(RGZ).0

RGPZ:=RGPZ-1

CT:=CT-1;

1

Кінець

*Рисунок 3.2-Змістовний мікроалгоритм*

**3.4 Таблиця станів регістрів**

*Таблиця 3.4- Таблиця станів регістрів*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№ такту** | **RGPZ** | **RGZ** | **ЛПН(L)** | **ППН(R)** | **СT** | **Мікрооперація** |
| **ПС** | 001000 | **1**1,011111010101110 | 0 | 1 | 100 |  |
| **1** | 000111 | 0**0**,**0**11111010101110  0**0**,**1**11110101011100 | 0 | 0 | 011 | Z’0 Z0:=  RGZ:=l(RGZ).0  RGPZ:=RGPZ-1  CT:=CT-1 |

**3.5 Функціональна схема з відображенням управляючих сигналів**

R

L

Z0

ShL

ShR

W1,W2

m+1 m m-1 RGZ 0

q CT 1

CT=0

inc

dec

W

W

dec

n+1 RGPZ 0

*Рисунок 3.3-Функціональна схема*

**3.6 Закодований мікроалгоритм**

*Таблиця 3.5- Таблиця кодування мікрооперацій*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблиця кодування мікрооперацій | |  | Таблиця кодування логічних умов | |
| МО | УС |  | ЛУ | Позначення |
| CT:=m;  RGZ:=Z;  Z’0 Z0:=  RGZ:=RGZ(m+2).r(RGZ)  RGPZ:=RGPZ+1  RGZ:=l(RGZ).0  RGPZ:=RGPZ-1  CT:=CT-1; | W  W1  W2  ShR  inc  ShL  dec  dec |  | Z’0 =0  0 | X1  X2  X3  X4 |
|  |

Z1

Початок

Z2

W,W1

X1

1

Z3

W2

1

ShR,inc

Z4

1

Z5

ShL,dec,dec

X4

1

Кінець

Z6

*Рисунок 3.4- Закодований мікроалгоритм*

**3.7 Граф управляючого автомата Мура з кодами вершин**

Z1

-

Q3Q2Q1

000

-

001

111

Z6

-

Z2

W,W1

X4

X2

Z5

ShL,dec,dec

-

Z3

W2

110

011

Z4

ShR,inc

X2

010

*Рисунок 3.5- Граф автомата Мура*

**3.8 Обробка порядків**

PZ=111.

**3.9 Форма запису нормалізованого результату з плаваючою комою в пам’ять**

Зн.Р P=+710 Зн.М M

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0. | 0 | 0 | 0 | 0 | 1 | 1 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1, | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |

**4. Операція добування кореня**

**Z=**

**4.1 Теоретичне обґрунтування способу**

Аргумент вводиться зі старших розрядів. Порядок результату дорівнює поділеному на два порядку аргумента. З мантиси добувається корінь завдяки нерівностям:

;

;

.

Виконання операції зводиться до послідовності дій:

1. Одержання остачі.

;

2. Якщо , то .

3. Якщо , то .

Відновлення остачі додає зайвий такт, але можна зробити інакше:

, тоді корінь добувається без відновлення залишку.

Для цього зсувається на 2 розряди ліворуч, а - на 1 розряд ліворуч, і формується як при діленні.

**4.2 Операційна схема**

n+1 RR 2 0

n-1 RZ 1 0

n-1 RX 2 0

СТ

*Рисунок 4.1-Операційна схема*

**4.3 Змістовний мікроалгоритм**

Початок

RX:=X;

RR:=0;

RZ:=0

CT:=15

RR[n+1]

1

RR:=RR+.11

RR:=RR+RZ.11

RR=LL(RR).RX(n-1;n-2)

RX:=LL(RX).00

RZ:=L(RZ).

CT:=CT-1

RZ[n]

1

Кінець

*Рисунок 4.2-Змістовний мікроалгоритм*

**4.4 Таблиця станів регістрів**

*Таблиця 4.1- Таблиця станів регістрів*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № ц. | RZ | RR | RX | CT |
| П.С.  Пзсув | 000000000000000 | 00000000000000000  **0**0000000000000010 | 100100011000110  010001100011000 | 1111 |
| 1 | 000000000000001 | +11111111111111111  00000000000000001  **0**0000000000000101 | 000110001100000 | 1110 |
| 2 | 000000000000011 | +11111111111111011  00000000000000000  **0**0000000000000000 | 011000110000000 | 1101 |
| 3 | 000000000000110 | +11111111111110011  11111111111110011  **1**1111111111001101 | 100011000000000 | 1100 |
| 4 | 000000000001100 | +00000000000011011  11111111111101000  **1**1111111110100010 | 001100000000000 | 1011 |
| 5 | 000000000011000 | +00000000000110011  11111111111010101  **1**1111111101010100 | 110000000000000 | 1010 |
| 6 | 000000000110000 | +00000000001100011  11111111110110111  **1**1111111011011111 | 000000000000000 | 1001 |
| 7 | 000000001100000 | +00000000011000011  11111111110100010  **1**1111111010001000 | 000000000000000 | 1000 |
| 8 | 000000011000001 | +00000000110000011  00000000000001011  **0**0000000000101100 | 000000000000000 | 0111 |
| 9 | 000000110000010 | +11111110011111011  11111110100100111  **1**1111010010011100 | 000000000000000 | 0110 |
| 10 | 000001100000100 | +00000011000001011  11111101010100111  **1**1110101010011100 | 000000000000000 | 0101 |
| 11 | 000011000001000 | +00000110000010011  11111011011000011  **1**1101101100001100 | 000000000000000 | 0100 |
| 12 | 000110000010000 | +00001100000100011  11111001100101111  **1**1100110010111100 | 000000000000000 | 0011 |
| 13 | 001100000100000 | +00011000001000011  11111110011111111  **1**1111001111111100 | 000000000000000 | 0010 |
| 14 | 011000001000001 | +00110000010000011  00101010001111111  **1**0101000111111100 | 000000000000000 | 0001 |
| 15 | 110000010000011 | +01100000100000111  00001001100000011 | 000000000000000 | 0000 |

**4.5 Функціональна схема з відображенням управляючих сигналів**

n+1 RR 2 0

n-1 RZ 1 0

n-1 RX 2 0

СТ

R

ShLL0

R

W1

W2

ShLL

WCT

dec

R1

ShL

*Рисунок 4.3-Функціональна схема*

**4.6 Закодований мікроалгоритм**

*Таблиця 4.2- Таблиця кодування мікрооперацій*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблиця кодування мікрооперацій | |  | Таблиця кодування логічних умов | |
| МО | УС |  | ЛУ | Позначення |
| RX:=X;  RR:=0;  RZ:=0  CT:=15  RR:=RR+RZ.11  RR:=RR+.11  RR=LL(RR).RX(n-1;n-2)  RX:=LL(RX).00  RZ:=L(RZ).  CT:=CT-1 | WX  R  R1  WCT  W1  W2  ShLL  ShLL0  ShL  dec |  | RR[n+1]  RZ[n] | X1  X2 |
|  |

Z1

Початок

Z2

WX,R,R1,WCT

X1

1

Z3

Z4

W2

W1

Z5

ShLL,ShLL0,ShL,dec

X2

Z6

1

Кінець

*Рисунок 4.4- Закодований мікроалгоритм*

**4.7 Граф управляючого автомата Мура з кодами вершин**

Q3Q2Q1

000

Z1

-

Z6

-

001

-

111

Z2

Y1

X2

Z5

Y4

Z3

Y2

-

Z4

Y3

-

110

011

010

*Рисунок 4.5- Граф автомата Мура*

**4.8 Обробка порядків**

PZ=PX:2=8:2=410=1002.

**4.9 Форма запису нормалізованого результату з плаваючою комою в пам’ять**

Зн.Р P=+410 Зн.М M

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0. | 0 | 0 | 0 | 0 | 1 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0, | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

**5. Синтез управляючого автомату для операційного пристрою множення третім способом.**

x3x2x1 +1=011=3 – операція множення третім способом.

**5.1 Таблиця співвідношення управляючих входів операційного автомата і виходів управляючого автомата**

За закодованим мікроалгоритмом (Рис. 1.14) складемо таблицю:

*Таблиця 5.1 Таблиця кодування сигналів*

|  |  |
| --- | --- |
| Входи операційного автомата | Виходи управляючого автомата |
| R,W2,W3,WCT | Y1 |
| W1 | Y2 |
| ShL1,ShL2,dec | Y3 |

**5.2 Мікроалгоритм в термінах управляючого автомата**

Зробимо автомат Мура циклічним задля зменшення кількості вершин.

Початок

Z1

Y1

Z2

X1

1

Y2

Z3

Z4

Y3

X2

1

Кінець

Z1

*Рисунок 5.1- Закодований мікроалгоритм*

Будуємо граф автомата Мура

00

01

Q1Q2

-

Z1

-

Z2

Y1

X1

X2

-

Z4

Y3

Z3

Y2

10

11

*Рисунок 5.2- Граф автомата Мура*

**5.3 Структурна таблиця автомата**

За графом автомата мура складаємо структурну таблицю автомата. Значення функцій збудження тригерів визначаються відповідно до графічної схеми переходів JK-тригера.

*Таблиця 5.2-Структурна таблиця автомата*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Перехід | Q2Q1 | Q2Q1 | x1x2 | y1y2y3 | J2 | K2 | J1 | K1 |
| *z1z2* | 00 | 01 | -- | 000 | 0 | - | 1 | - |
| *z2 z3* | 01 | 11 | 1- | 100 | 1 | - | - | 0 |
| *z2 z4* | 01 | 10 | 0- | 100 | 1 | - | - | 1 |
| *z3 z4* | 11 | 10 | -- | 010 | - | 0 | - | 1 |
| *z4 z3* | 10 | 11 | 10 | 001 | - | 0 | 1 | - |
| *z4 z1* | 10 | 00 | -1 | 001 | - | 1 | 0 | - |
| *z4 z4* | 10 | 10 | 00 | 001 | - | 0 | 0 | - |

JK-тригер:

**5.4 Синтез функцій виходів і переходів**

0

1

0

0

Y3

1

0

0

0

Y2

0

0

1

0

Q2

Q1

Y1

1

1

-

-

1

1

-

-

0

0

-

-

1

1

-

-

Q2

Q1

X2

X1

K1

-

-

0

0

-

-

0

1

-

-

1

1

-

-

1

1

Q2

Q1

X2

X1

J1

0

0

1

0

0

0

-

1

0

-

-

-

-

-

-

-

-

Q2

Q1

X2

X1

K2

-

-

-

-

-

-

-

-

1

1

0

0

1

1

0

0

Q2

Q1

X2

X1

J2

*Рисунок 5.3- Діаграми Вейча*

**5.5 Функціональна схема пристрою (виходи управляючого автомата**

**підключені до входів операційного автомата)**

­­

K2

R

K

C

S

J

G

“1”

D2

R

K

C

S

J

J1

&

&

&

Y1

Y2

Y3

“1”

***2n+1***

***2n+1***

***0***

***0***

***SM***

***n***

***2n+1***

***n+1***

**0**

***RG2***

***0***

***n-1***

***RG1***

***2n***

***0***

***2n+1***

***RG3***

***0***

***n-1***

***Y1***

***Y2***

***Y3***

***Y1***

***Y3***

***Y1***

***n***

***X***

***n***

***Y***

***CT***

***s***

***Y1***

***Y3***

***x2***

&

J1

1

K2

K1

&

1

*Рисунок 5.5- Функціональна схема пристрою*

**Висновки**