**Содержание:**

1. Выбор варианта задания
   1. Определение параметров задания
   2. Постановка задачи
2. Математическая часть
   1. Структура вычислительного процесса
   2. Алгоритм обработки А2-7
   3. Алгоритм обработки B2-14
   4. Алгоритм обработки С
3. Структурная часть
   1. Обозначения операторов, используемых в однородной вычислительной среде (ОВС) и обозначения для временных диаграмм
   2. Обработка числа А
   3. Обработка числа В
   4. Обработка числа С
   5. Общие вычисления
4. Программная часть
   1. Описание используемых кодов
   2. Программирование общей ОВС
   3. Схема цепей программирование
   4. Программа на ОВС
5. Заключение
6. Список используемой литературы
7. **Выбор варианта задания**
   1. **Определение параметров задания**
      1. Номер на потоке

К = (3-1) \* 30 + 8 = 68

* + 1. Формула вычислений

X = 68 mod 11 = 2

C = |A-((5B)/4)|

* + 1. Система счисления для A

Y = 68 mod 3 = 2

A2-7

* + 1. Система счисления для B

Z = 68 mod 7 = 5

B2-14

* + 1. Тип кода для вычислений

Вычисления в дополнительном коде, т.к. К – чётное

* + 1. Направление получения и выдачи чисел

V = 68 mod 64 = 4 = 1002 =00 01 002 , где 00 – С, 01 – B, 00 – A

00

Ю

С

10

11 В

01 З

**A** и **B** поступают с **Юга** и **Запада**, а **С** выдаётся на **Юг**

* + 1. Вычисление времени программирования в тактах

Tпр < (2 + 2 + 5 + 6) \* 16

Tпр < 240

* 1. **Постановка задачи**

Ставится задача вычислений по формуле:

**C = |A – 5B/4|**

Где числа **А** и **В** представляются в **(2-7)** и **(2-14)** системах счисления, т.е. **А = А2-7**, **В = В2-14**, обрабатываются в **дополнительном коде** и поступают с **Юга** и **Запада** соответственно, а результат **С** выдаётся на **Юг**. Программирование ОВС необходимо выполнить за время, не превышающее **240 тактов**.

При выполнении работы необходимо стремиться к выполнению работы с минимальным количеством микросхем и наименьшем временем программирования, которое возможно при минимальном количестве входов программирования.

1. **Математическая часть**
   1. **Структура вычислительного процесса**

Вычисления в ОВС производятся как последовательно, так и параллельно. Для решаемой задачи удобно выделить три алгоритма: обработка числа А, обработка числа В и их совместная обработка для получения результата С.

**С**

**А2-7**

**В2-14**

**С= |(А\* + B\*)пр|**

**В\* = (-(((В2-14)2 \* 5) \ 4))дп**

**А\* = ((А2-7)2)дп**

**A2-7**

**B2-14**

**A\***

**B\***

*Рисунок 1. Структура вычислительного процесса*

* 1. **Алгоритм обработки А2-7**
     1. Представления алгоритма **А** в виде блок-схемы

Ввод числа А

1

2

3

4

5

6

7

Начало

Конец

Начало алгоритма

Конец алгоритма

A2-7

LA = A2-7 & 0F16

HA = (A2-7 & F016) 2-4

A\*

A2 = HA + 2HA + 4HA + LA

Вычисление младшей тетрады

Вычисление старшей тетрады

Перевод числа А в двоичный код

Вывод А

*Рисунок 2. Алгоритм обработки А*

* + 1. Описание алгоритма **А**

П1. Начало алгоритма

П2. Вводим число A2-7

П3. Вычисляем младшую тетраду, логически умножая введённое число на OF16

а8а7а6а5а4а3а2а1 – А2-7

0 0 0 0 1 1 1 1 – 0F16

0 0 0 0 а4а3а2а1 – LA

П4. Вычисляем старшую тетраду, логически умножая на F016

а8а7а6а5а4а3а2а1 – А2-7

1 1 1 1 0 0 0 0 – F016

а8а7а6а5 0 0 0 0 – H'A

Полученный результат H'A не является старшей тетрадой, так как полученной число старше на 4 порядка, нам необходимо сдвинуть его вправо, для этого необходимо умножить на 2-4, отсюда получаем:

0 0 0 0 а8а7а6а5 – HA

П5. Переводим А в двоичный код, так как вводимое А в 2-7 системе счисления, то необходимо H \* 7 + L, так как умножение – это сложение со степенью двойки, то мы имеем 4H + 2H + H + L

П6. Вывод преобразованного для дальнейших вычислений числа А

П7. Конец алгоритма

* 1. **Алгоритм обработки В2-14**
     1. Представления алгоритма **В** в виде блок-схемы

Умножение на 5

Вывод В

Деление на 4

Перевод числа В в двоичный код

Начало

Конец

Начало алгоритма

Конец алгоритма

B2-14

B\*

Вычисление старшей тетрады

HB = (B2-14 & F016) 2-4

1

2

3

4

5

6

7

8

9

10

B2 = B2-14 – 2HB

B' = 4B2 + B2

B'' = B' & FC16

B'''= B''+ FF16

B\* = ˥( B''' & B''')

*Рисунок 3. Алгоритм обработки В*

Ввод числа B

Перевод B в обратный код

Перевод В в дополнительный код

* + 1. Описание алгоритма **В**

П1. Начало алгоритма

П2. Вводим число B2-14

П3. Вычисляем старшую тетраду, логически умножая на F016

b8b7b6b5b4b3b2b1 – B2-14

1 1 1 1 0 0 0 0 – F016

b8b7b6b5 0 0 0 0 – H'B

Полученный результат H'B не является старшей тетрадой, так как полученной число старше на 4 порядка, нам необходимо сдвинуть его вправо, для этого необходимо умножить на 2-4, отсюда получаем:

0 0 0 0 b8b7b6b5 – HB

П4. Переводим B в двоичный код. Т.к. B2-14 = 16HB + LB, а B2 = 14HB + LB, то мы можем упростить наши вычисления представив перевод как

B2 = B2-14­ – 2HB

П5. Умножаем В на 5, так как умножение а двоичном коде – это сложение со степенью двойки, то мы имеем 4Н + Н

П6. Делим В на 4 логически умножая его на константу FC16

П7. Переводим В в обратный код, инвертируя число

П8. Переводим В в дополнительный код, логически умножая число само на себя и затем инвертируя его

П9. Вывод преобразованного для дальнейших вычислений числа B

П10. Конец алгоритма

* 1. **Алгоритм обработки С**
     1. Представления алгоритма **С** в виде блок-схемы

1

2

3

4

5

6

7

Начало

Конец

Начало алгоритма

Конец алгоритма

A\*, B\*

Ввод чисел А и В

Вывод С

C

Cдп = A\* + B\*

C' = Cдп + ЗнЗн

С = С' ⊕ ЗнЗн

Сложение А и В в дополнительном коде

Проверка знака числа C'

Вычитание знака числа

*Рисунок 4. Алгоритм обработки С*

* + 1. Описание алгоритма **С**

П1. Начало алгоритма

П2. Вводим преобразованные числа A\* и В\*

П3. Вычисляем Сдп складывая преобразованные А\* и В\*

П4. Вычитаем знак числа из полученного Сдп­

П5. Складываем С' по модулю 2 с его знаками, если полученное С' отрицательное, то мы получим результат в прямом коде. Если же C' положительное – то мы получим число без изменений.

П6. Вывод результата С

П7. Конец алгоритма

1. **Структурная часть**
   1. **Обозначения операторов, используемых в однородной вычислительной среде (ОВС) и обозначения для временных диаграмм**

Операторы, используемые в ОВС:

GC – генератор константы

& – логическое умножение

+ – арифметическое сложение

DD – цифровая задержка

&˥ – логическое умножение с инверсией

⊕ – сумма по модулю 2

Краткое описание ОВС:

Стрелками показано направление движение данных, цифровые значения, например 1т, указывают задержку во времени. Всё время является дискретным и изменяется в тактах (т). Пунктирной линий обозначен транзит данных через операционные элементы (ОЭ). Рядом с линиями (или над ними), соединяющими ОЭ присутствуют символьные обозначения – т.н. метка данных, для понимания происходящего процесса. Нумерация рядом с ОЭ идёт по порядку их следования. В Общей ОВС нумерация идёт почипово – чип – это квадратная матрица 3х3, состоящая, соответственно, из 9 ОЭ, расположенных на одинаковом расстоянии относительно друг друга. Во временных диаграммах (ВД), в свою очередь, первый столбец – # - столбец с адресом ячейки, второй столбец – символьный идентификатор, первая строка – дискретное время – такты. Стрелками во ВД обозначаются задержки на 1 или 9 тактов.

Значения, используемые во временных диаграммах:

an + an = x

an + x = y

an + y = z

˥bn = i

bn + i = m

bn + 1 = n

n + n = k

m + n = s

m + m = p

k + 1 = e

s + 1 = w

p + 1 = q

e + 1 = u

w + 1 = o

q + 1 = v

u + u = j

o + o = d

v + v = r

e + e = h

j + x = t

d + y = f

d + z = g

r + z = l

r + x = ъ

d + an = э

t + зн = ы

f + зн = ч

g + зн = м

l + зн = т

ъ + зн = ь

э + зн = б

d + зн = г

h + зн = з

ы ⊕ зн = с1

ч ⊕ зн = с2

м ⊕ зн = с3

т ⊕ зн = с4

ь ⊕ зн = с5

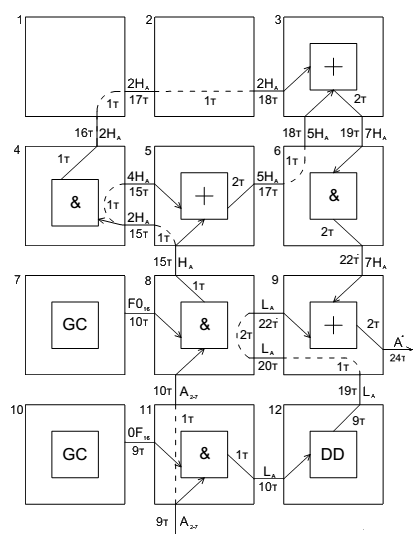
о ⊕ зн = с6

г ⊕ зн = с7

з ⊕ зн = с8

* 1. **Обработка числа А**
     1. ОВС обработки числа А

На рис. 5 представлена ОВС для обработки числа А.

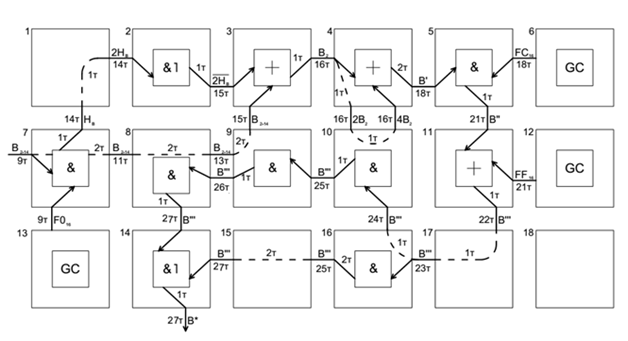
****

*Рисунок 5. ОВС для обработки числа А*

Данные поступают на 9 такте в 11 ОЭ с юга, откуда и начинаются вычисления. С 10 ОЭ мы получаем константу 0F16 и логически умножаем её с поступившим числом А2-7, полученный результат – младшую тетраду LА выдаём на 12 ОЭ, через 11 ОЭ передаём транзитом А2-7 на 8 ОЭ, где логически умножаем его на константу F016 генерируемую в соседнем – 7 ОЭ, результат выдаём с задержкой 5 тактов (1 такт тратится на вычисление и 4 дополнительно для помещение смещение старшей тетрады на место младшей (см. пояснение к рис. 2). Далее мы получаем 7HA постепенно складывая HA с самим собой, задержанным на определённое количество тактов (1 для получения 2HA и 2 для получения 4HA). Потом в 9 ОЭ мы складываем 7HA с LA и получаем желаемое А\*, которое выдаём на 24 такте на восток.

* 1. **Обработка числа В**
     1. ОВС обработки числа В

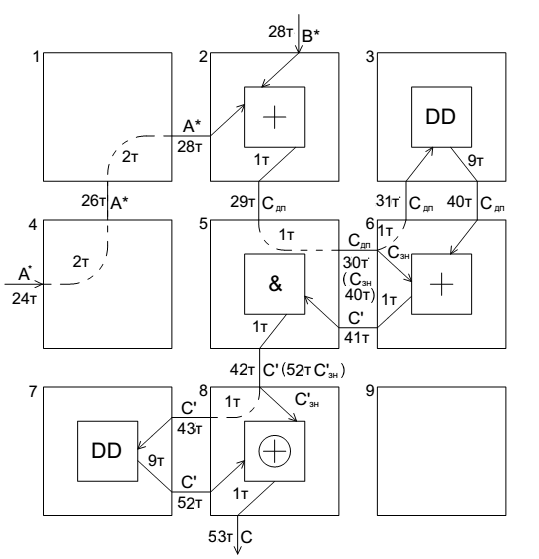
На рис. 6 предоставлена ОВС для вычисления числа В.



*Рисунок 6. ОВС для обработки числа В*

* 1. **Обработка числа С**
     1. ОВС обработки числа С

На рис. 7 предоставлена ОВС для вычисления числа С.



*Рисунок 7. ОВС для обработки числа С*

1. **Программная часть**
   1. **Описание используемых кодов**

Первый столбец – т.н. адрес ОЭ, первая строка – регистр команды. Регистр команды занимает 16-бит: КОП – 3 бита, Y – 1 бит, А1 – А6 – по 2 бита.

Коды выдачи данных:

002 – юг

012 – запад

102 – север

112 – восток

Код операции (КОП):

0002 – логическое умножение (&)

0012 – логическое умножение с инверсией (&˥)

0112 – сумма по модулю 2 (⊕)

1002 – арифметическое сложение (+)

1102 – при Y = 0 – операция генерации константы (GC). В процессе программирования в регистр команды на место А2 – А5 записывается 8-битовая константа. В Процессе обработки константа циклически сдвигается от А2 к А5.

1102 – при Y = 1 – операция цифровой задержки (DD) на 9 тактов. Код А1 задаёт направление приёма операнда, а код А6 направление выдачи. Коды программирования А2 – А5 задаются произвольно

Другие коды:

Y – определяет задержку на 1 или 2 такта, принимая соответственно значения 0 или 1

А1, А2 – коды управления входа чисел

А3 – код управления входа числа для транзита

А4 – код управления выхода числа

А5, А6 – коды управления выхода транзита при задержке на 2 или 1 такт соответственно

Q16 – общий код команды в шестнадцатеричной системе счисления

* 1. **Программирование общей ОВС**

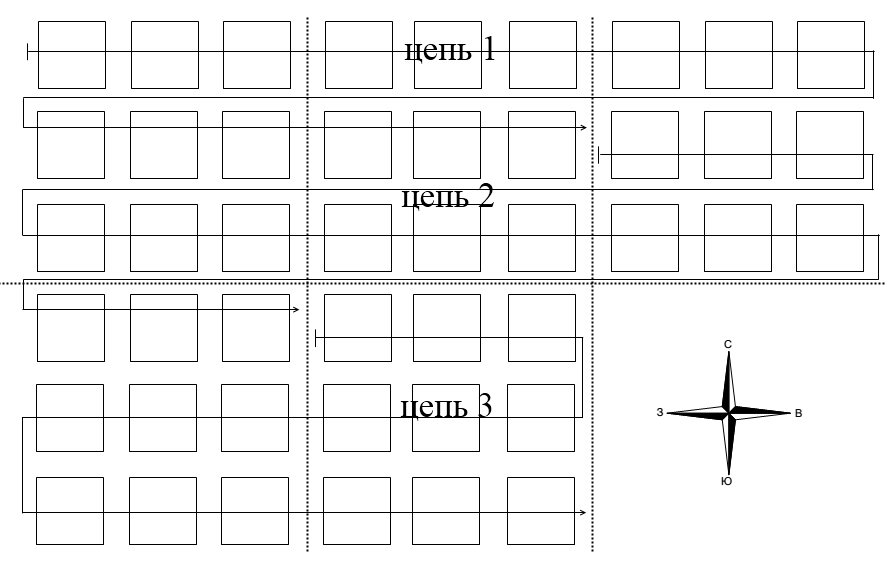
В таблице 5 показано программирование общей ОВС

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ОЭ** | **К** | **О** | **П** | **Y** | **А** | **1** | **А** | **2** | **А** | **3** | **А** | **4** | **А** | **5** | **А** | **6** | **Q16** |
| **1.1** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000016 |
| **1.2** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000016 |
| **1.3** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000016 |
| **1.4** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000016 |
| **1.5** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000016 |
| **1.6** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000016 |
| **1.7** | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 050316 |
| **1.8** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 004316 |
| **1.9** | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 948A16 |
| **2.1** | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0A0316 |
| **ОЭ** | **К** | **О** | **П** | **Y** | **А** | **1** | **А** | **2** | **А** | **3** | **А** | **4** | **А** | **5** | **А** | **6** | **Q16** |
| **2.2** | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 24B016 |
| **2.3** | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 847016 |
| **2.4** | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 146C16 |
| **2.5** | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1C4D16 |
| **2.6** | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1C5816 |
| **2.7** | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | C1E216 |
| **2.8** | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 2B0516 |
| **2.9** | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 10C416 |
| **3.1** | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 947416 |
| **3.2** | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 070516 |
| **3.3** | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | C0FD16 |
| **3.4** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 019216 |
| **3.5** | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 8B0A16 |
| **3.6** | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | C3FD16 |
| **3.7** | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1CD216 |
| **3.8** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 008116 |
| **3.9** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000016 |
| **4.1** | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0CE316 |
| **4.2** | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 913116 |
| **4.3** | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 184616 |
| **4.4** | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | C3C316 |
| **4.5** | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 11E316 |
| **4.6** | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 963116 |
| **4.7** | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | C21F16 |
| **4.8** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 013216 |
| **4.9** | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | D40216 |
| **5.1** | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 10D116 |
| **5.2** | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 860516 |
| **5.3** | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | D15416 |
| **5.4** | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 104816 |
| **5.5** | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0C8716 |
| **5.6** | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 865216 |
| **5.7** | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | DC0316 |
| **5.8** | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 668916 |
| **5.9** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 000016 |

*Таблица 5. Программирование ОВС*

* 1. **Схема цепей программирования**

Микросхемы имеют три входа и три выхода программирования, через которые последовательно программируются ОЭ строк. Последовательные соединения регистров команды всех ОЭ ОВС позволяет программировать ОВС через 1 вход программирования, однако требует для этого максимальное количество тактов. При меньшем времени программирования выполняется несколько последовательных цепей, которые программируются параллельно через несколько входов программирования. Наименьшее время программирования достигается при равномерном распределении звеньев между цепями программирования. Программирование одного ОЭ занимает 16 тактов. На рис. 9 представлена схема цепей программирования для данной ОВС.



**ВХОД 2**

**ВХОД 3**

**ВХОД 1**

*Рисунок 9. Схема цепей программирования*

* 1. **Программа на ОВС**

вычислительный алгоритм технологический операционный

Программа для ОВС представляется двухмерным массивом бит с количеством строк равным количеству входов программирования. В каждой строке записываются коды команд ОЭ в порядке их очередности в цепи программирования. Коды команд ОЭ удобно представлять в 16-ричной системе счисления. В таблице 6 представлена программа для данной ОВС

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Цепь 1** | **ОЭ** | **1.1** | **1.2** | **1.3** | **2.1** | **2.2** | **2.3** | **3.1** | **3.2** | **3.3** |
| **Команда** | 000016 | 000016 | 000016 | 0A0316 | 24B016 | 847016 | 947416 | 070516 | C0FD16 |
|  | **ОЭ** | **1.4** | **1.5** | **1.6** | **2.4** | **2.5** | **2.6** |  |  |  |
| **Команда** | 000016 | 000016 | 000016 | 146C16 | 1C4D16 | 1C5816 |  |  |  |
| **Цепь 2** | **ОЭ** | **3.4** | **3.5** | **3.6** | **1.7** | **1.8** | **1.9** | **2.7** | **2.8** | **2.9** |
| **Команда** | 019216 | 8B0A16 | C3FD16 | 050316 | 004316 | 948A16 | C1E216 | 2B0516 | 10C416 |
|  | **ОЭ** | **3.7** | **3.8** | **3.9** | **4.1** | **4.2** | **4.3** |  |  |  |
| **Команда** | 1CD216 | 008116 | 000016 | 0CE316 | 913116 | 184616 |  |  |  |
| **Цепь 3** | **ОЭ** | **5.1** | **5.2** | **5.3** | **4.4** | **4.5** | **4.6** | **5.4** | **5.5** | **5.6** |
| **Команда** | 10D116 | 860516 | D15416 | C3C316 | 11E316 | 963116 | 104816 | 0C8716 | 865216 |
|  | **ОЭ** | **4.7** | **4.8** | **4.9** | **5.7** | **5.8** | **5.9** |  |  |  |
|  | **Команда** | C21F16 | 013216 | D40216 | DC0316 | 668916 | 000016 |  |  |  |

*Таблица 6. Программа для ОВС*

1. **Заключение**

Во время выполнения данной работы была разработана ОВС, которая проводит вычисления по формуле:

**C = |A – 5B/4|**

Где числа **А** и **В** представляются в **(2-7)** и **(2-14)** системах счисления, т.е. **А = А2-7**, **В = В2-14**, обрабатываются в **дополнительном коде** и поступают с **Юга** и **Запада** соответственно на **9** такте, а результат **С** выдаётся на **Юг** с **53** по **60** такт включительно.

Вычисления в ОВС производятся последовательно-параллельно. Параллельное вычисление **А\*** и **В\***, затем вычисление С. Для вычисления А\*, В\* и С разработаны алгоритмы и отдельные ОВС. ОВС объединены в одну общую, которая занимает 5 чипов и состоит из 45 ОЭ. Для каждой ОВС построена временная диаграмма. ОВС запрограммирована, и программа переведена в шестнадцатеричный код. Программа внесена в ОВС **тремя** параллельными **цепями программирования**. Все программирование ОВС произведено за **240 тактов**.

1. **Список используемой литературы**
2. Дрозд А. А. – «Методические указания к расчётно-графической работе по теме выполнение арифметико-логических операций на однородной вычислительной среде» – Одесса: ОНПУ, 2011
3. Евреинов Э. В. – «Однородные вычислительные системы, структуры и среды» – М.: Радио и связь, 1981
4. Рабинович З. Л., Раманаускас В. А. – «Типовые операции в вычислительных машинах» – Киев: Техника, 1980
5. Савельев А. Я. – «Прикладная теория цифровых автоматов» – М.: Высшая школа, 1987
6. Самофалов К. Г., Романкевич А. М., Валуйский В. Н., Каневский Ю. С., Пиневич М. М. – «Прикладная теория цифровых автоматов» – Киев: Высшая школа, 1987