Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет   
информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Дисциплина: Арифметические и логические основы вычислительной техники

К ЗАЩИТЕ ДОПУСТИТЬ

\_\_\_\_\_\_\_\_\_\_Ю. А. Луцик

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе  
на тему

ПРОЕКТИРОВАНИЕ И ЛОГИЧЕСКИЙ СИНТЕЗ СУММАТОРА-УМНОЖИТЕЛЯ ДВОИЧНО-ЧЕТВЕРИЧНЫХ ЧИСЕЛ

БГУИР КР 1-40 02 01 302 ПЗ

Студент

Руководитель

К. А. Балталина

И. В. Лукьянова

МИНСК 2021

Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет   
информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Дисциплина: Арифметические и логические основы вычислительной техники

УТВЕРЖДАЮ

Заведующий кафедрой ЭВМ

\_\_\_\_\_\_\_\_\_ Б. В. Никульшин

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_2021 г.

ЗАДАНИЕ

по курсовой работе студентки  
 Балталиной Карины Алексеевны

1. Тема работы: Проектирование и логический синтез сумматора-умножителя двоично-четверичных чисел.
2. Срок сдачи студентом законченной работы: 20 мая 2021 г.
3. Исходные данные к работе:
   1. исходные сомножители: Мн = 78,12 ; Мт = 39, 97.
   2. алгоритм умножения: Г.
   3. метод умножения: умножение закодированного двоично-четверичного множимого на два разряда двоичного множителя одновременно в прямых кодах.
   4. коды четверичных цифр множимого для перехода к двоично-четверичной системе кодирования; 04 – 00, 14 – 01, 24 – 11, 34 – 10.
   5. тип синтезируемого умножителя: 2.
   6. логический базис для реализации ОЧС: А1; метод минимизации – Алгоритм Рота.
   7. логический базис для реализации ОЧУ: А4; метод минимизации – карты Карно-Вейча.
4. Содержание пояснительной записки (перечень подлежащих разработке вопросов):

Введение. 1. Разработка алгоритма умножения. 2. Разработка структурной схемы сумматора-умножителя 3. Логический синтез одноразрядного четверичного умножителя. 4. Логический синтез одноразрядного четверичного сумматора. 5. Логический синтез одноразрядного четвертичного сумматора на основе мультиплексора. 6. Логический синтез преобразователя множителя. 7. Временные затраты на умножение. Заключение. Список литературы.

1. Перечень графического материала:
   1. Сумматор-умножитель второго типа. Схема электрическая структурная.
   2. Одноразрядный четвертичный сумматор. Схема электрическая функциональная.
   3. Одноразрядный четверичный умножитель. Схема электрическая функциональная.
   4. Преобразователь множителя.Схема электрическая функциональная.
   5. Одноразрядный четвертичных сумматор. Реализация на мультиплексорах. Схема электрическая функциональная.

КАЛЕНДАРНЫЙ ПЛАН

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование этапов курсовой работы | Объём этапа, % | Срок выполнения этапа | Примечания |
| Разработка алгоритма умножения | 10 | 15.02-20.02 |  |
| Разработка структурной схемы сумматора-умножителя | 10 | 21.02-10.03 | С выполнением чертежа |
| Разработка функциональных схем основных узлов сумматора-умножителя | 50 | 10.03-27.03 | С выполнением чертежей |
| Синтез комбинационных схем устройств на основе мультиплексоров | 10 | 27.03-11.04 | С выполнением чертежа |
| Завершение оформления пояснительной записки | 20 | 11.04-16.04 |  |

Дата выдачи задания: 28 февраля 2020 г.

Руководитель И. В. Лукьянова

ЗАДАНИЕ ПРИНЯЛ К ИСПОЛНЕНИЮ \_\_\_\_\_\_\_\_\_\_

Оглавление

[1. Разработка алгоритма умножения 8](#_Toc72804691)

[2. разработка структурной схемы сумматора-умножителя 12](#_Toc72804692)

[5 Логический синтез преобразователя множителя (ПМ) 29](#_Toc72804693)

[6 оценка результатов разработки 30](#_Toc72804694)

[Заключение 31](#_Toc72804695)

[Приложение А 33](#_Toc72804696)

[Приложение Б 34](#_Toc72804697)

[Приложение В 35](#_Toc72804698)

[Приложение Г 36](#_Toc72804699)

[Приложение Д 37](#_Toc72804700)

**ВВЕДЕНИЕ**

Данная курсовая работа посвящена разработке алгоритмов выполнения операций умножения и сложения. На основе полученных алгоритмов требуется разработать и синтезировать следующие устройства: одноразрядный четвертичный сумматор (ОЧС), одноразрядный четвертичный умножитель (ОЧУ), а также переключательные функции ОЧС на мультиплексорах. Минимизация перечисленных устройств осуществляется с помощью карт Карно-Вейча и алгоритма извлечения Рота. На основе полученных данных требуется построить схемы этих устройств и проанализировать результаты (эффективность минимизации и время выполнения операций).

# Разработка алгоритма умножения

1. Перевод сомножителей из десятичной системы счисления в четверичную.

Мн10 = 78,12; Мт10 = 39,97.

**Множимое**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| – | 78 | 4 |  | – | 19 | 4 |  | – | 4 | 4 |  | × | 0,12 |  | × | 0,48 |
| 4 | 19 |  | 16 | 4 |  | 4 | 1 |  | 4 |  | 4 |
| – | 38 |  |  |  | 3 |  |  |  | 0 |  |  |  | 0,48 |  |  | 1,92 |
| 36 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Мн4 = 1032,01.

В соответствии с заданной кодировкой множимого:

Мн2/4 = 01001011,0001.

**Множитель**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| – | 39 | 4 |  | – | 9 | 4 |  | × | 0,97 |  | × | 0,88 |
| 36 | 9 |  | 8 | 2 |  | 4 |  | 4 |
|  | 3 |  |  |  | 1 |  |  |  | 3,88 |  |  | 3,52 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |
| --- | --- |
| × | 0,52 |
| 4 |
|  | 2,08 |

Мт4 = 213,332.

В соответствии с обычной весомозначной кодировкой множимого:

Мт2/4 = 110110,101011.

2. Запишем сомножители в форме с плавающей запятой в дополнительно коде:

Мн = 0,01001011000110; PМн = 0.0010 (+310) – закодировано по заданию;

Мт = 0,110110101011; PМт = 0.0100 (+410) – закодировано традиционно.

3. Умножение двух чисел с плавающей запятой на два разряда множителя одновременно в дополнительных кодах сводится к сложению порядков, преобразованию разрядов множителя согласно алгоритму и перемножению мантисс сомножителей.

Порядок произведения будет следующим:

Запишем сомножители в форме с плавающей запятой в прямом коде:  
Мн = 0,01001011000110 РМн = 0,0010 + 0310 – закодировано по заданию  
Мт = 0, 110110101011 РМт = 0,0100 + 0410 – закодировано традиционно

Умножение двух чисел с плавающей запятой на два разряда множителя одновременно в прямых кодах. Это сводится к сложению порядков, формированию знака произведения, преобразованию разрядов множителя согласно алгоритму, и перемножению мантисс сомножителей.

Порядок произведения будет равен:

РМн = 0.0010 034

РМт = 0.0100 104

РМн∙Мт = 0.0110 134

Результат закодирован в соответствии с заданием на кодировку множимого.

Знак произведения определяется суммой по модулю “два” знаков сомножителей:

зн Мн ⊕ зн Мт = 0 ⊕ 0 = 0

Для умножения мантисс необходимо предварительно преобразовать множитель. При умножении чисел в дополнительных кодах диада 11(34) заменяется на триаду 1, диада 10(24) заменяется на триаду 10().

Преобразованный множитель имеет вид:

;

Для выполнения операции умножения нам понадобятся заготовки в виде ;

;

;

;

Перемножение мантисс по алгоритму «Г» приведено в таблице 1.1.

Таблица 1.1 - Перемножение мантисс

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Четверичная с/c | | Двоично-четверичная с/с | | Комментарии |
| **1** | | **2** | | **3** |
| 0. | 0000000000000 | 00 | 00000000000000000000000000 | ∑0 ч |
| 0. | 0103201000000 | 00 | 00010010110001000000000000 | П1ч |
| 0. | 0103201000000 | 00 | 00010010110001000000000000 | ∑1 ч |
| 3. | 3312033200000 | 10 | 10100111001010110000000000 | П2ч |
| 0. | 0021300200000 | 00 | 00001101100000110000000000 | ∑2 ч |
| 0. | 0002130020000 | 00 | 00000011011000001100000000 | П4ч |
| 0. | 0030030220000 | 00 | 00001000001000111100000000 | ∑4 ч |
| 0. | 0000000000000 | 00 | 00000000000000000000000000 | П5ч |
| 0. | 0030030220000 | 00 | 00001000001000111100000000 | ∑5 ч |
| 0. | 0000000000000 | 00 | 00000000000000000000000000 | П6ч |
| 0. | 0030030220000 | 00 | 00001000001000111100000000 | ∑6 ч |
| 3. | 3333333120332 | 10 | 10101010101010011100101011 | П7ч |
| 0. | 0030030000332 | 00 | 00001000001000000000101011 | ∑7ч |

Во избежание переполнения увеличиваем порядок Мн \* Мт на единицу . Таким образом порядок равен 8 .

После окончания умножения необходимо оценить погрешность вычислений.

Для этого полученное произведение (Мн ∙ Мт4 = 0, 0030030000332, (\* 8) приводится к нулевому порядку, а затем переводится в десятичную систему счисления:

= 300300,00332

= 3120,06

Результат прямого перемножения операндов дает следующее значение:

= 3122,46

Абсолютная погрешность:

Δ = 3122,46 - 3120,06 = 2,4

Относительная погрешность:

δ; δ =

Эта погрешность получена за счёт приближенного перевода из десятичной системы счисления в четверичную обоих сомножителей, а также за счёт округления полученного результата произведения.

# разработка структурной схемы сумматора-умножителя

Структурная схема сумматора-умножителя второго типа для алгоритма умножения «Г» приведена на рисунке приложения А.

Если устройство работает как сумматор, то оба слагаемых последовательно заносятся в регистр множимого, а на управляющий вход формирователя дополнительного кода поступает «1». В ОЧС первое слагаемое складывается с нулём, записанным в регистре результата, и переписывается без изменений в регистр результата. На втором такте второе слагаемое из регистра множимого через цепочку ОЧУС попадает на входы ОЧС и складывается с первым слагаемым, хранящимся в регистре результата.

Если устройство работает как умножитель, то множимое и множитель помещаются в соответствующие регистры, а на управляющий вход ФДК поступает «0». Диада множителя поступает на входы преобразователя множителя. Единица переноса в следующую диаду, если она возникает, должна быть добавлена к следующей диаде множителя.

В регистре множителя после каждого такта умножения содержимое сдвигается на два двоичных разряда, и в конце умножения регистр обнуляется. В случае когда выход 2ПМ переходит в единичное состояние инициализируется управляющий вход F1 формирователя дополнительного кода, и на выходах ФДК формируется дополнительный код множимого с обратным знаком. На выходах 3 и 4 ПМ формируются диады преобразованного множителя, которые поступают на входы ОЧУС вместе с диадами множимого. На трёх выходах ОЧУС формируется результат умножения диад Мн•Мт плюс перенос из предыдущего ОЧУС. Максимальной цифрой в диаде преобразованного множителя является двойка, поэтому перенос, формируемый ОЧУС, может быть только двоичным («0» или «1»). Частичные произведения, получаемые на выходах ОЧУС, складываются с накапливаемой частичной суммой из регистра результата с помощью цепочки. Количество тактов умножения определяется разрядностью Мт.

1. **РАЗАРАБОТКА ФУНКЦИОНАЛЬНЫХ СХЕМ ОСНОВНЫХ**

**УЗЛОВ СУММАТОРА-УМНОЖИТЕЛЯ**

* 1. **Логический синтез одноразрядного четвертичного умножителя-сумматора.**

ОЧУС – это комбинационное устройство, имеющее шесть входов (два разряда из регистра множимого, два разряда из регистра множителя, вход переноса и управляющий вход h) и три выхода.

Принцип работы ОЧУС представлен с помощью таблицы истинности (таблица 3.1).

Разряды множителя закодированы: 0 – 00, 1 – 01, 2 – 10, 3 – 11.

Разряды множимого закодированы: 0 – 00, 1 – 01, 2 – 11, 3 – 10.

Управляющий вход h определяет тип операции:

«0» – умножение закодированных цифр, поступивших на информационные входы;

«1» – вывод на выходы без изменения значения разрядов, поступивших из регистра множимого.

Таблица 3.1

| **Пер.** | **Мн** | | **Мт** | | **Упр.** | **Перенос** | **Результат** | | **Результаты вычисления в четверичной с/с** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Р1*** | ***x1*** | ***x2*** | ***y1*** | ***y2*** | ***h*** | ***P*** | ***Q1*** | ***Q2*** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0\*0+0=00 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Выход-код “00” |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0\*1\*+0=00 |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | Выход-код “00” |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0\*2+0=00 |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | Выход-код “00” |
| 0 | 0 | 0 | 1 | 1 | 0 | X | X | Х | 0\*3+0=00 |
| 0 | 0 | 0 | 1 | 1 | 1 | X | X | Х | Выход “00” |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1\*0+0=00 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | Выход “01” |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1\*1+0=01 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | Выход “01” |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1\*2+0=02 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | Выход “01” |
| 0 | 0 | 1 | 1 | 1 | 0 | X | X | Х | 1\*3+0=03 |
| 0 | 0 | 1 | 1 | 1 | 1 | X | X | Х | Выход “01” |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3\*0+0=00 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | Выход “03” |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 3\*1+0=03 |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | Выход “03” |
| 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 3\*2+0=12 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | Выход “03” |
| 0 | 1 | 0 | 1 | 1 | 0 | X | X | Х | 3\*3+0=21 |
| 0 | 1 | 0 | 1 | 1 | 1 | X | X | Х | Выход “03” |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 2\*0+0=00 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | Выход “02” |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 2\*1+0=02 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | Выход “02” |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 2\*2+0=10 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | Выход “02” |
| 0 | 1 | 1 | 1 | 1 | 0 | X | X | Х | 2\*3+0=12 |
| 0 | 1 | 1 | 1 | 1 | 1 | X | X | Х | Выход “02” |
| 1 | 0 | 0 | 0 | 0 | 0 | Х | Х | Х | 0\*0+1=01 |
| 1 | 0 | 0 | 0 | 0 | 1 | Х | Х | Х | Выход “00” |
| 1 | 0 | 0 | 0 | 1 | 0 | Х | Х | Х | 0\*1+1=01 |
| 1 | 0 | 0 | 0 | 1 | 1 | Х | Х | Х | Выход “00” |
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0\*2+1=01 |
| 1 | 0 | 0 | 1 | 0 | 1 | Х | Х | Х | Выход “00” |
| 1 | 0 | 0 | 1 | 1 | 0 | X | X | Х | 0\*3+1=01 |
| 1 | 0 | 0 | 1 | 1 | 1 | X | X | Х | Выход “00” |
| 1 | 0 | 1 | 0 | 0 | 0 | Х | Х | Х | 1\*0+1=01 |
| 1 | 0 | 1 | 0 | 0 | 1 | Х | Х | Х | Выход “01” |
| 1 | 0 | 1 | 0 | 1 | 0 | Х | Х | Х | 1\*1+1=02 |
| 1 | 0 | 1 | 0 | 1 | 1 | Х | Х | Х | Выход “01” |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1\*2+1=03 |
| 1 | 0 | 1 | 1 | 0 | 1 | Х | Х | Х | Выход “01” |
| 1 | 0 | 1 | 1 | 1 | 0 | X | X | Х | 1\*3+1=10 |
| 1 | 0 | 1 | 1 | 1 | 1 | X | X | Х | Выход “01” |
| 1 | 1 | 0 | 0 | 0 | 0 | Х | Х | Х | 3\*0+1=01 |
| 1 | 1 | 0 | 0 | 0 | 1 | Х | Х | Х | Выход “03” |
| 1 | 1 | 0 | 0 | 1 | 0 | Х | Х | Х | 3\*1+1=10 |
| 1 | 1 | 0 | 0 | 1 | 1 | Х | Х | Х | Выход “03” |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 3\*2+1=13 |
| 1 | 1 | 0 | 1 | 0 | 1 | Х | Х | Х | Выход “03” |
| 1 | 1 | 0 | 1 | 1 | 0 | X | X | Х | 3\*3+1=22 |
| 1 | 1 | 0 | 1 | 1 | 1 | X | X | Х | Выход “03” |
| 1 | 1 | 1 | 0 | 0 | 0 | Х | Х | Х | 2\*0+1=01 |
| 1 | 1 | 1 | 0 | 0 | 1 | Х | Х | Х | Выход “02” |
| 1 | 1 | 1 | 0 | 1 | 0 | Х | Х | Х | 2\*1+1=03 |
| 1 | 1 | 1 | 0 | 1 | 1 | Х | Х | Х | Выход “02” |
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 2\*2+1=11 |
| 1 | 1 | 1 | 1 | 0 | 1 | Х | Х | Х | Выход “02” |
| 1 | 1 | 1 | 1 | 1 | 0 | X | X | Х | 2\*3+1=13 |
| 1 | 1 | 1 | 1 | 1 | 1 | X | X | Х | Выход “02” |

Управляющий вход h определяет тип операции: 0 – умножение закодированных цифр, поступивших на информационные входы, и добавление переноса; 1 – вывод на выходы без изменения значений разрядов, поступивших из регистра множимого.

В таблице 3.1 выделено 36 безразличных наборов, т. к. на входы ОЧУС из разрядов множителя не может поступить код «11», при работе ОЧУС как сумматора на вход переноса не может поступить единица, а при умножении на ноль или единицу на вход переноса также не может поступить единица.

Минимизацию функций P,Q1 и Q2  проведем при помощи карт Вейча/

Для функции P :

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | x | x | x | x | x | x | x | x |  |  |  |
|  |  | x | x | x | x | x | x | x | x |  |  |  |
| p |  | x | x | x | x | x | x | x | x |  |  |  |
|  |  | 1 | x | x | 1 |  | x | x |  |  |  |  |
|  |  | 1 |  |  | 1 |  |  |  |  |  |  |  |
|  |  | x | x | x | x | x | x | x | x |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

h h

Для функции Q1 :

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | x | x | x | x | x | x | x | x |  |  |  |
|  |  | x | x | x | x | x | x | x | x |  |  |  |
| p |  | x | x | x | x | x | x | x | x |  |  |  |
|  |  | 1 | x | x |  | 1 | x | x |  |  |  |  |
|  |  | 1 | 1 | 1 |  | 1 |  |  |  |  |  |  |
|  |  | x | x | x | x | x | x | x | x |  |  |  |
|  |  | 1 | 1 | 1 | 1 |  |  |  |  |  |  |  |
|  |  |  | 1 | 1 |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

h h

Для функции Q2 :

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | x | x | x | x | x | x | x | x |  |  |  |
|  |  | x | x | x |  | x | x | x | x |  |  |  |
| p |  | x | x | x | x | x | x | x | x |  |  |  |
|  |  | 0 | x | x |  | 0 | x | x |  |  |  |  |
|  |  |  | 0 |  | 0 | 0 |  | 0 | 0 |  |  |  |
|  |  | x | x | x | x | x | x | x | x |  |  |  |
|  |  | 0 | 0 |  |  |  |  | 0 | 0 |  |  |  |
|  |  | 0 | 0 |  | 0 | 0 |  | 0 | 0 |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

h h

Следовательно :

*.*

.

Функциональная схема ОЧУC в заданном базисе представлена в приложении Б.

**3.2.Логический синтез одноразрядного четвертичного сумматора**

Одноразрядный четверичный сумматор – это комбинационное устройство, имеющее 5 двоичных входов (2 разряда одного слагаемого, 2 разряда второго слагаемого и вход переноса) и 3 двоичных выхода.

Принцип работы ОЧС представлен с помощью таблицы истинности (таблица 3.2).

Разряды обоих слагаемых закодированы: 0 – 00; 1 – 01; 2 – 11; 3 – 10.

Так как ОЧС синтезируется для схемы второго типа, то безразличные наборы в таблице истинности отсутствуют.

*Таблица 3.2*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***a1*** | ***a2*** | ***b1*** | ***b2*** | ***p*** | **П** | ***S1*** | ***S2*** | **Операции**  **в четверичной с/с** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0+0+0=00 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0+0+1=01 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0+1+0=01 |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0+1+1=02 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0+3+0=03 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0+3+1=10 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0+2+0=02 |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0+2+1=03 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1+0+0=01 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1+0+1=02 |
| 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1+1+0=02 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1+1+1=03 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1+3+0=10 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1+3+1=11 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1+2+0=03 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1+2+1=10 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 3+0+0=03 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 3+0+1=10 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 3+1+0=10 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 3+1+1=11 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 3+3+0=12 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 3+3+1=13 |
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 3+2+0=11 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 3+2+1=12 |
| 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 2+0+0=02 |

Продолжение таблицы 3.2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1+3+1=11 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1+0+0=01 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1+0+1=02 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1+2+0=03 |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1+2+1=10 |
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1+1+0=02 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1+1+1=03 |

Минимизацию переключательной функций *П* проведём с помощью алгоритма Рота, а функции *S1 и S2* с помощью карты Карно.

Для функции :

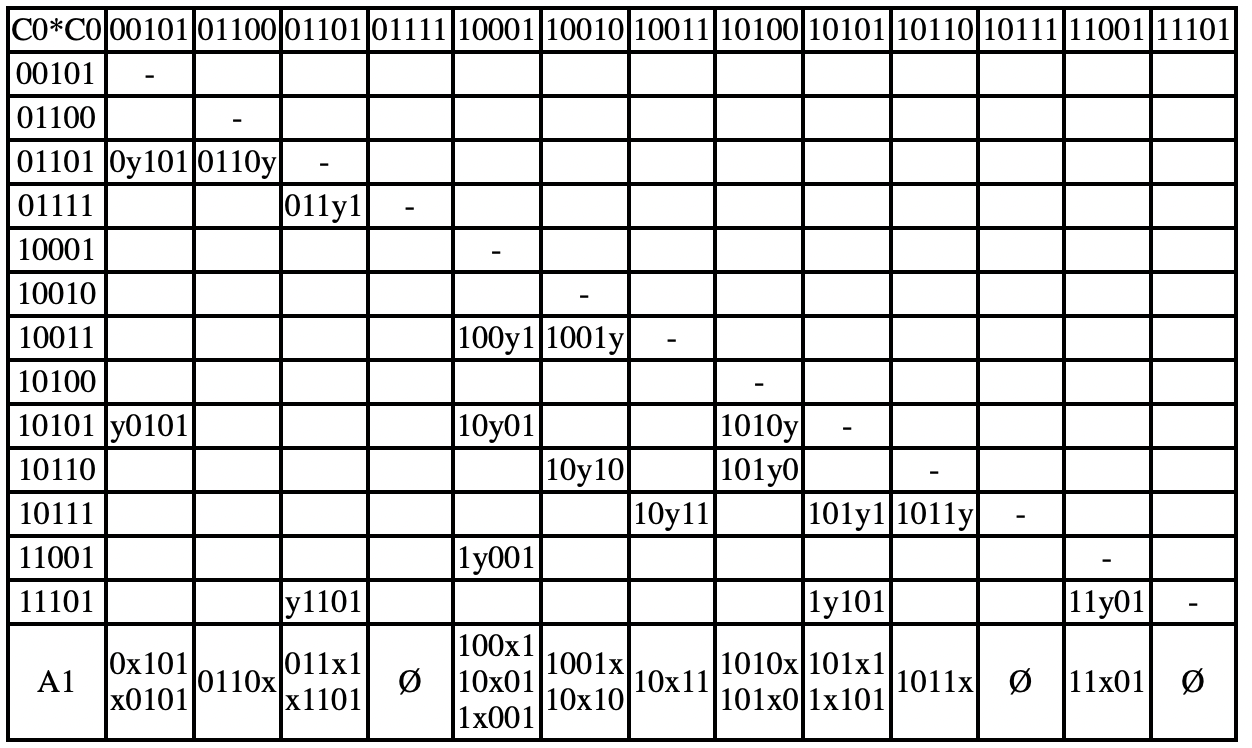
определим множество единичных кубов:

Множество безразличных кубов:

N=

Сформируем множество :

Первый шаг умножения () приведён в таблице 3.2



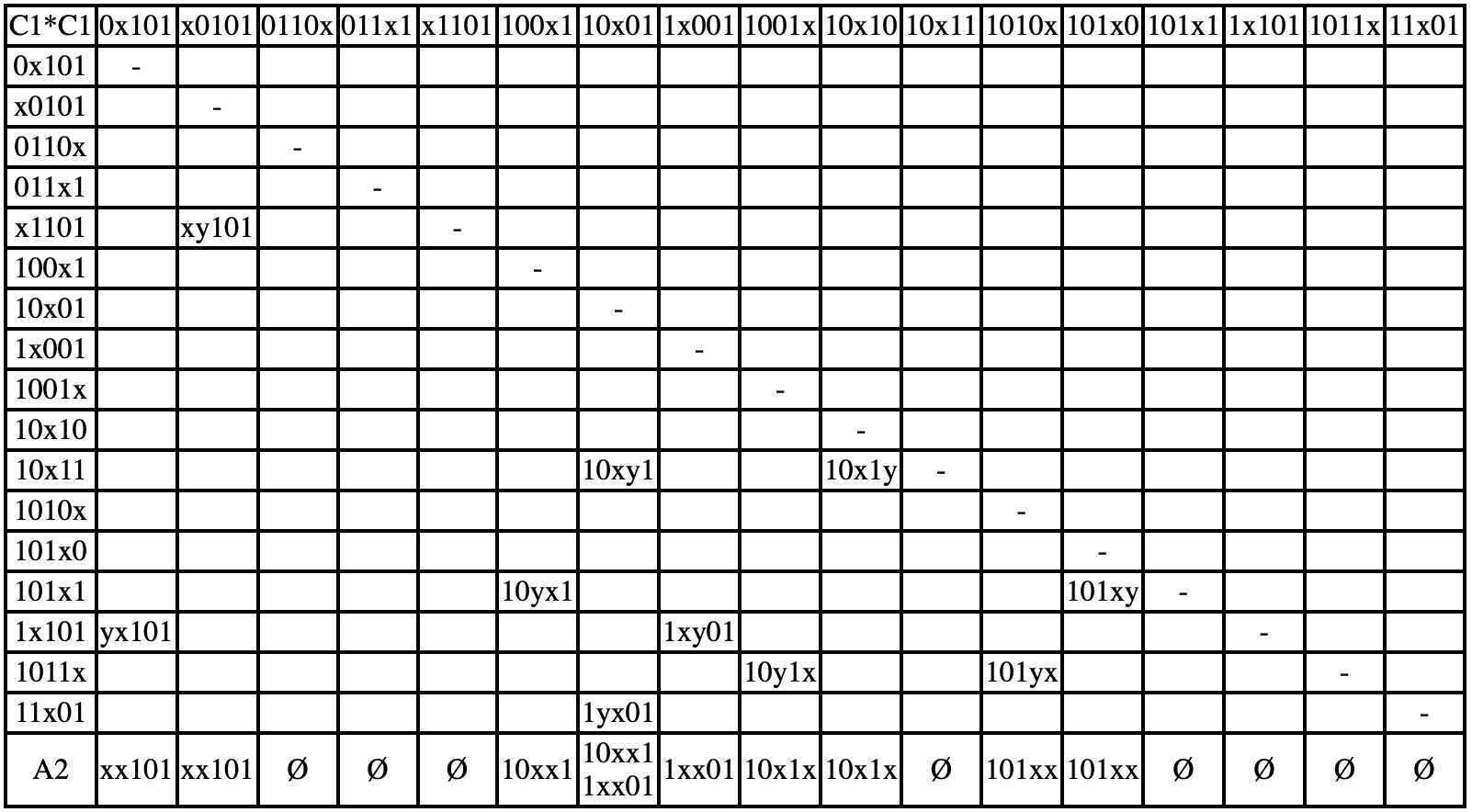
*Таблица 3.2*

В результате этой операции сформируется новое множество кубов:

##### C1 = { 0x101; x0101; 0110x; 011x1; x1101; 100x1; 10x01; 1x001; 1001x; 10x10; 10x11; 1010x; 101x0; 101x1; 1x101; 1011x; 11x01 }

Множество *Z0* пустое.

В таблице 3.3 приведён следующий шаг поиска простых импликант с помощью операции .



*Таблица 3.3*

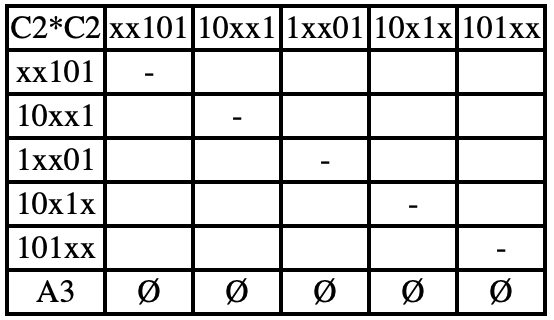
В результате этой операции сформируется новое множество кубов:

##### C2 = { xx101; 10xx1; 1xx01; 10x1x; 101xx }.

Множество *Z1* = { 0110x; 011x1 } .

В таблице 3.4 приведён следующий шаг поиска простых импликант с помощью операции .

*Таблица 3.4*



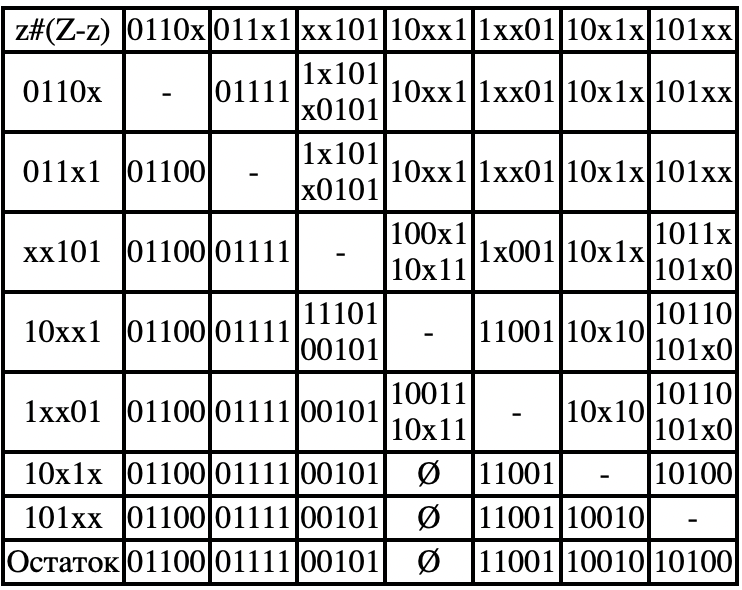
На этом поиск простых импликант заканчивается, кубы из множества С2 являются кубами множества Z2

Множество простых импликант:

##### Z = { 0110x; 011x1; xx101; 10xx1; 1xx01; 10x1x; 101xx }.

Следующий этап – поиск *L-*экстремалей на множестве простых импликант (таблица 3.5). Для этого используется операция # (решетчатое вычитание).

*Таблица 3.5 – Поиск L-экстремалей*

****

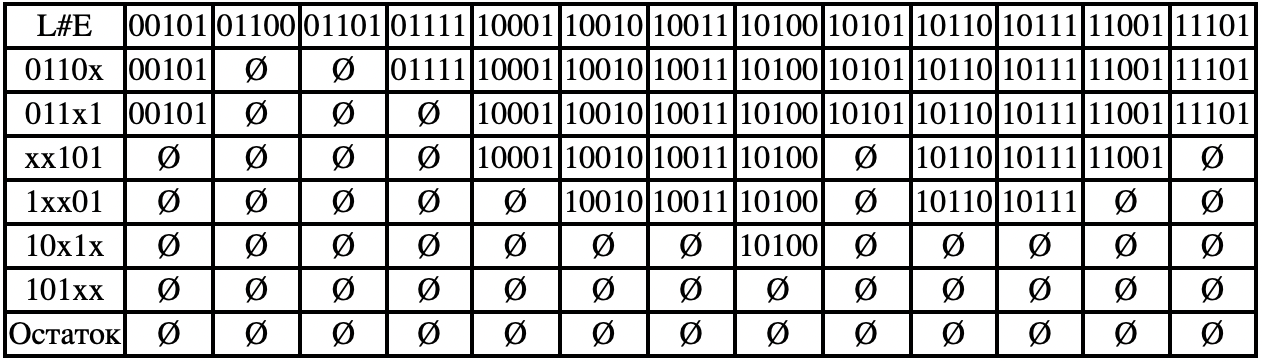
*L*-экстремалями стали следующие простые импликанты:

##### E = { 0110x; 011x1; xx101; 1xx01; 10x1x; 101xx }.

Проверка *L*-экстремалей.

Находим множество кубов , не покрываемых L-экстремалями (Таблица 3.6).Для этого используется операция # (решетчатое вычитание).

*Таблица 3.6*

****

Следовательно множество кубов, непокрываемых L-экстремалями,

L'=L#E:

10xx1.

В результате :

##### Минимальное покрытие - множество L-экстремалей

##### E = { 0110x; 011x1; xx101; 1xx01; 10x1x; 101xx };

Следовательно :

.

Для функции S1:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 001 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
| 00 |  |  | 1 |  | 1 | 1 |  | 1 |
| 01 |  | 1 | 1 | 1 | 1 |  |  |  |
| 11 | 1 | 1 |  | 1 |  |  | 1 |  |
| 10 | 1 |  |  |  |  | 1 | 1 | 1 |

*Таблица 3.3*

Следовательно.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 001 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
| 00 |  | 1 | 1 | 1 | 1 |  |  |  |
| 01 | 1 | 1 |  | 1 |  |  | 1 |  |
| 11 | 1 |  |  |  |  | 1 | 1 | 1 |
| 10 |  |  | 1 |  | 1 | 1 |  | 1 |

Для функции S2:

Следовательно.

Эффективность минимизаций можно оценить отношением числа входов схем, реализующих переключательную функцию до и после минимизации:

𝐾1=;

𝐾2=; 𝐾3=;

Функциональная схема ОЧC в заданном базисе представлена в приложении В.

1. **СИНТЕЗ ОЧС НА ОСНОВЕ МУЛЬТИПЛЕКСОРА**

Мультиплексор – это логическая схема, имеющая n входов,m управляющих входов и один выход. При этом должно выполняться равенство . На выход мультиплексора может быть пропущен без изменений любой (один) логический сигнал, поступающий на информационные входы. Порядковый номер информационного входа, значение с которого в данный момент должно быть передано на выход, должно быть передано на выход, определяется двоичным кодам на управляющих входах. Для синтеза ОЧС будем использовать мультиплексор “один из восьми” (1 из 8).

*Таблица 4.1 – таблица истинности для синтеза ПФ ОЧС*

| ***a1*** | ***a2*** | ***b1*** | ***b2*** | ***p*** | **П** | **П** | ***S1*** | ***S1*** | ***S2*** | ***S2*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** |  | **7** |  | **8** |  |
| 0 | 0 | 0 | 0 | 0 | 1 | “0” | 1 |  | 0 |  |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 1 |  | 1 |  | 1 |  |
| 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | “0” | 0 |  | 0 |  |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 0 |  | 1 |  | 0 |  |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 |  | 1 |  | 1 |  |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 |  | 0 |  | 1 |  |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |

*Продолжение таблицы 4.1*

| 1 | 1 | 0 | 0 | 0 | 1 |  | 0 |  | 1 |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | “1” | 0 |  | 0 |  |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
|  | | | | | | | | | | |

Функциональная схема ОЧС на мультиплексорах представлена в приложении Г.

# Логический синтез преобразователя множителя (ПМ)

Преобразователь множителя (ПМ) служит для исключения из множителя диад 11, заменяя их на триады и диад 10, заменяя их на триады .

*Таблица истинности ПМ. Таблица 5*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Вх. диада | | Мл. бит | Зн. | Вых. диада | |
| Qn | Qn-1 | Qn-2 | P | S1 | S2 |
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 |

Проведём минимизацию **P** при помощи карты Карно:

Qn

Qn-1Qn-2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **P** | 00 | 01 | 11 | 10 |
| 0 |  |  |  |  |
| 1 | 1 | 1 | 1 | 1 |

P = Qn

Очевидно, что S1 не минимизируется, поэтому .

Проведём минимизацию **S2** при помощи карты Карно:

Qn-1Qn-2

Qn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S2** | 00 | 01 | 11 | 10 |
| 0 |  | 1 |  | 1 |
| 1 |  | 1 |  | 1 |

# оценка результатов разработки

Формула расчёта временных затрат на умножение:

, где

– время преобразования множителя;

– время формирования дополнительного кода множимого;

– время умножения на ОЧУС;

– время формирования единицы переноса в ОЧС;

– время сдвига частичной суммы;

n – количество разрядов на множителе.

**Заключение**

В процессе выполнения курсовой работы была разработана структурная схема сумматора-умножителя второго типа, а также функциональные схемы основных узлов данного устройства. Для уменьшения стоимости логических схем были выполнены минимизации переключательных функций различными способами. Такой подход позволил выявить достоинства и недостатки этих алгоритмов.

В качестве главного достоинства минимизации картами Карно (Вейче) можно выделить простоту и минимальные затраты времени. Однако применение данного способа для функций многих переменных будет затруднительно. Функциональные схемы были построены в различных логических базисах. Это позволило закрепить теоретические знания основных законов булевой алгебры, например, правило де Моргана. Также можно отметить, что необходимо сократить количество уровней в логической схеме для уменьшения времени работы данного устройства.

Реализация переключательных функций на основе мультиплексоров позволила облегчить процесс минимизации этих функций.

1. **Список использованных источников**

Единая система конструкторской документации (ЕСКД) : справ. пособие / С. С. Борушек [и др.]. – М. : Изд-во стандартов, 1989. – 352 с.

Искра, Н. А. Арифметические и логические основы вычислительной техники : пособие / Н. А. Искра, И. В. Лукьянова, Ю. А. Луцик. – Минск : БГУИР, 2016. – 75 с.

Луцик, Ю. А. Учебное пособие по курсу «Арифметические и логические основы вычислительной техники» / Ю. А. Луцик, И. В. Лукьянова, М. П. Ожигина. – Минск : МРТИ, 2001. – 77 с.

Лысиков, Б. Г. Арифметические и логические основы цифровых авто- матов / Б. Г. Лысиков. – Минск : Выш. шк., 1980. – 342 с.

Лысиков, Б. Г. Цифровая вычислительная техника / Б. Г. Лысиков. – Минск : Выш. шк., 2003. – 242 с.

Савельев, А. Я. Прикладная теория цифровых автоматов / А. Я. Са-вельев. – М. : Высш. шк., 1987. – 272 с.

Усатенко, С. Т. Выполнение электрических схем по ЕСКД : справочник / С. Т. Усатенко, Т. К. Каченюк, М. В. Терехова. – М. : Изд-во стандартов, 1989. – 325 с.

**Приложение А**

*(обязательное)*

Сумматор-умножитель второго типа. Схема электрическая структурная

**Приложение Б**

*(обязательное)*

Одноразрядный четверичный умножитель-сумматор. Схема электрическая функциональная

**Приложение В**

*(обязательное)*

Одноразрядный четверичный сумматор. Схема электрическая функциональная

**Приложение Г**

*(обязательное)*

Одноразрядный четверичный сумматор. Реализация на мультиплексорах. Схема электрическая функциональная

**Приложение Д**

*(обязательное)*

Преобразователь множителя. Схема электрическая функциональная