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 01 207 ПЗ

Студент

Руководитель

М.В. Жук

Ю. А. Луцик

МИНСК 2020

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

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_2020 г.

ЗАДАНИЕ

по курсовой работе студента  
 Жук Максим Владимировича

1. Тема работы: Проектирование и логический синтез сумматора-умножителя двоично-четверичных чисел.
2. Срок сдачи студентом законченной работы: 25 мая 2019 г.
3. Исходные данные к работе:
   1. исходные сомножители: Мн = 46,75; Мт = 58,12.
   2. алгоритм умножения: А.
   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 – 01, 14 – 00, 24 – 11, 34 – 10.
   5. тип синтезируемого умножителя: структурные схемы приведены для умножителя 1-ого типа (ОЧУ, ОЧС, аккумулятор).

.

* 1. логический базис для реализации ОЧС: А3; метод минимизации – карты Карно-Вейча.
  2. логический базис для реализации ОЧУ: А7; метод минимизации – алгоритм Рота.

1. Содержание пояснительной записки (перечень подлежащих разработке вопросов):

1. Разработка алгоритм умножения. 2. Разработка структурной схемы сумматора-умножителя. 3. Разработка функциональных схем основных узлов сумматора-умножителя. 4. Синтез комбинационных схем устройств на основе мультиплексоров. 5. Оценка результатов разработки. Заключение. Список литературы.

1. Перечень графического материала:
   1. Сумматор-умножитель первого типа. Схема электрическая структурная.
   2. Одноразрядный четвертичный сумматор. Схема электрическая функциональная.
   3. Регистр-аккумулятор. Схема электрическая функциональная.
   4. Одноразрядный четвертичных сумматор. Реализация на мультиплексорах. Схема электрическая функциональная.

Дата выдачи задания: 26 февраля 2020 г.

Руководитель Ю. А. Луцик

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

**СОДЕРЖАНИЕ**

Введение ……………………………………………………………………… 5

1. Разработка алгоритма умножения ………………………………………… 6

2. Разработка структурной схемы сумматора-умножителя …………………. 9

3. Логический синтез одноразрядного четверичного сумматора ………….. 10

4. Логический синтез одноразрядного четверичного умножителя………..…15

5. Логический синтез одноразрядного четвертичного сумматора на

основе мультиплексора …………………..………………………………… 23

6. Логический синтез преобразователя множителя ………………………. 25

7. Временные затраты на умножение .……………………………………... 26

Заключение ………………………………………………………………….. 29

Список использованных источников ...………………………………….… 30

Приложение А ………………………………………………………………. 31

Приложение Б ……………………………………………………………….. 32

Приложение В ………………………………………………………………. 33

Приложение Г ………………………………………………………………. 34

Приложение Д ………………………………………………………………. 35

**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).

# 

**1. РАЗРАБОТКА АЛГОРИТМА УМНОЖЕНИЯ**

* 1. **Перевод сомножителей из десятичной системы счисления в четверичную.**

Множимое

46 | 4 0.75 Мн4 = 232,300

44 11| 4 4 в соответствии с заданной

2 8 2 3.00 кодировкой множимого

3 Мн2/4 = 111011,100000

Множитель

58 | 4 0.12 Мт4 = 322,013

56 14 | 4 4

2 12 3 0.48 Мт2/4 = 101111,010010

2 4 *множитель представляется*

1.92 *обычным весомозначным*

4 *кодом:* 04 - 01, 14 - 00, 24 - 11, 34 - 10

3.68 для всех вариантов

Запишем сомножители в форме с плавающей запятой в прямом коде:

Мн = 0,111011100000 РМн = 0.0010 + 0310 – закодировано по заданию

Мт = 0,101111010010 РМт = 0.0011 + 0310 – 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 03

РМт = 0.0011 03

РМн∙Мт = 0.0110 12

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 прямых кодах диада 11(34) заменяется на триаду 1. Преобразованный множитель имеет вид: Мтп4 = 2202 или Мтп2= 0111110011. Перемножение мантисс по алгоритму “А” приведено в табл. 1.1.

Таблица 1.1 - Перемножение мантисс

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Четверичная с/c | | | Двоично-четверичная с/с | | | Комментарии |
| **1** | | | **2** | | | **3** |
| 0. | 000000 |  | 01. | 010101010101 |  | ∑0 ч |
| 3. | 101100 |  | 10. | 000100000101 |  | П1Ч=[-Мн] |
| 3. | 101100 |  | 10. | 000100000101 |  | ∑1 ч |
| 3. | 310110 | 0 | 10. | 100001000001 | 01 | ∑1 ч∙ 2-2 |
| 1. | 131200 |  | 10. | 001000110101 |  | П2Ч =Мн∙2 |
| 1. | 101310 | 0 | 10. | 000100100001 | 01 | ∑2 ч |
| 0. | 110131 | 00 | 01. | 010001001000 | 0101 | ∑2 ч∙ 2-2 |
| 0. | 011013 | 100 | 01. | 010000010010 | 000101 | ∑3 ч∙ 2-2 |
| 1. | 131200 |  | 00. | 001000110101 |  | П4Ч= Мн∙2 |
| 1. | 202213 |  | 00. | 110111110010 | 000101 | ∑4 ч |
| 0. | 120221 | 3100 | 01. | 001000110101 | 10000101 | ∑4 ч∙ 2-2 |
| 1. | 131200 |  | 00. | 001000110101 |  | П5Ч= Мн∙2 |
| 1. | 312021 | 3100 |  | 100111011100 | 10000101 | ∑5 ч |
| 0. | 131202 | 13100 | 01. | 001000110111 | 0010000101 | ∑5 ч∙ 2-2 |
| 3. | 101100 |  | 10. | 000100000101 |  | П6Ч= [-Мн] |
| 3. | 232302 | 13100 | 10. | 111011100111 | 0010000101 | ∑6 ч |
| 3. | 323230 | 213100 | 10. | 101110111001 | 110010000101 | ∑6 ч∙ 2-2 |
| 0. | 232300 |  | 01. | 111011100101 |  | П7Ч =Мн∙1 |
| 0. | 222130 | 213100 | 01. | 111111001001 | 110010000101 | ∑7 ч |
| 0. | 022213 | 0213100 | 01. | 011111110010 | 01110010000101 | ∑7ч∙2-2=Мн∙Mт |

После окончания умножения необходимо оценить погрешность вычислений. Для этого полученное произведение (Мн∙Мт4 = 0, 0222130213100, РМн∙Мт = 6) приводится к нулевому порядку, а затем переводится в десятичную систему счисления:

Мн∙Мт4 = 222130.213100 РМн∙Мт = 0;

Мн∙Мт10 = 2716.61328125

Результат прямого перемножения операндов дает следующее значение:

Мн10 · Мт10 = 46,75 · 58,12 = 2 717,11.

Абсолютная погрешность:

Δ = 2 717,11 – 2716.61328125= 0,49671875.

Относительная погрешность:

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

**2 РАЗРАБОТКА СТРУКТУРНОЙ СХЕМЫ**

**СУММАТОРА-УМНОЖИТЕЛЯ**

Структурная схема сумматора-умножителя первого типа для алгоритма умножения «A» представлена на схеме 1.

Структура первого типа строится на базе заданных узлов ОЧУ, ОЧС, формирователя дополнительного кода и регистра результата. Управление режимами работы схемы осуществляется внешним сигналом *Mul/sum,* который определяет вид текущей арифметической операции (умножение или суммирование).

Принцип работы ФДК в зависимости от управляющих сигналов приведён в таблице 2.1.

Таблица 2.1 – Режимы работы формирователя дополнительного кода

|  |  |  |
| --- | --- | --- |
| **Сигналы на входах ФДК** | | **Результат на выходах ФДК** |
| **F1** | **F2** |
| 0 | 0 | Дополнительный код множимого |
| 0 | 1 | Дополнительный код слагаемого |
| 1 | 0 | Меняется знак множимого |
| 1 | 1 | Дополнительный код слагаемого |

**3. РАЗРАБОТКА ФУНКЦИОНАЛЬНЫХ СХЕМ ОСНОВНЫХ УЗЛОВ СУММАТОРА-УМНОЖИТЕЛЯ**

**3.2. Логический синтез одноразрядного четверичного сумматора**

Одноразрядный четверичный сумматор – это комбинационное устройство, имеющее 5 входов (2 разряда одного слагаемого, 2 разряда второго слагаемого и вход переноса) и 3 выхода. Принцип работы ОЧС представлен с помощью таблицы истинности (табл. 3.2.1).

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

В таблице 3.2 выделено 24 безразличных наборов, т.к. со старших выходов ( ОЧС?) не могут прийти коды «2» и «3».

Таблица 3.2.1 – Таблица истинности ОЧС

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

*Продолжение таблицы 3.2.1*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 2+1+1=10 |
| 1 | 1 | 1 | 0 | 0 | x | x | x | 2+3+0=11 |
| 1 | 1 | 1 | 0 | 1 | x | x | x | 2+3+1=12 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 2+0+0=02 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 2+0+1=03 |
| 1 | 1 | 1 | 1 | 0 | x | x | x | 2+2+0=10 |
| 1 | 1 | 1 | 1 | 1 | x | x | x | 2+2+1=11 |

Минимизацию функций *П, S1* и *S2*проведем при помощи карт Карно.

Для функции *П*:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
| 00 | 0 | 0 | 0 | 0 | x | x | x | x |
| 01 | 0 | 0 | 0 | 0 | x | x | x | x |
| 11 | 0 | 1 | 0 | 0 | x | x | x | x |
| 10 | 1 | 1 | 1 | 0 | x | x | x | x |

Рисунок 3.2.1 – Минимизация функции *П*при помощи карты Карно.

*p =*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
| 00 | 0 | 0 | 0 | 0 | x | x | x | x |
| 01 | 0 | 0 | 0 | 0 | x | x | x | x |
| 11 | 0 | 1 | 0 | 0 | x | x | x | x |
| 10 | 1 | 1 | 1 | 0 | x | x | x | x |

Рисунок 3.2.1 – Минимизация функции *П*при помощи карты Карно.

*.*

Для функции *S1*:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
| 00 | 1 | 1 | 1 | 0 | x | x | x | x |
| 01 | 0 | 1 | 0 | 0 | x | x | x | x |
| 11 | 1 | 0 | 1 | 1 | x | x | x | x |
| 10 | 0 | 0 | 0 | 1 | x | x | x | x |

Рисунок 3.2.3 – Минимизация функции *S1* при помощи карты Карно.

*=*

*= .*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
| 00 | 1 | 1 | 1 | 0 | x | x | x | x |
| 01 | 0 | 1 | 0 | 0 | x | x | x | x |
| 11 | 1 | 0 | 1 | 1 | x | x | x | x |
| 10 | 0 | 0 | 0 | 1 | x | x | x | x |

Рисунок 3.2.4 – Минимизация функции *S1* при помощи карты Карно.

*)*

Для функции *S2*:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
| 00 | 1 | 0 | 1 | 0 | x | x | x | x |
| 01 | 0 | 1 | 0 | 1 | x | x | x | x |
| 11 | 0 | 1 | 0 | 1 | x | x | x | x |
| 10 | 1 | 0 | 1 | 0 | x | x | x | x |

Рисунок 3.2.5 – Минимизация функции *S2* при помощи карты Карно.

*=*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
| 00 | 1 | 0 | 1 | 0 | x | x | x | x |
| 01 | 0 | 1 | 0 | 1 | x | x | x | x |
| 11 | 0 | 1 | 0 | 1 | x | x | x | x |
| 10 | 1 | 0 | 1 | 0 | x | x | x | x |

Рисунок 3.2.6 – Минимизация функции *S2* при помощи карты Карно.

*)*

**3.1. Логический синтез одноразрядного четверичного умножителя**

Одноразрядный четверичный умножитель – это комбинационное устройство, имеющее 5 двоичных входов (2 разряда из регистра Мн, 2 разряда из регистра Мт и управляющий вход h) и 4 двоичных выхода. Принцип работы ОЧУ описывается с помощью таблицы истинности (табл.3.1.1).

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

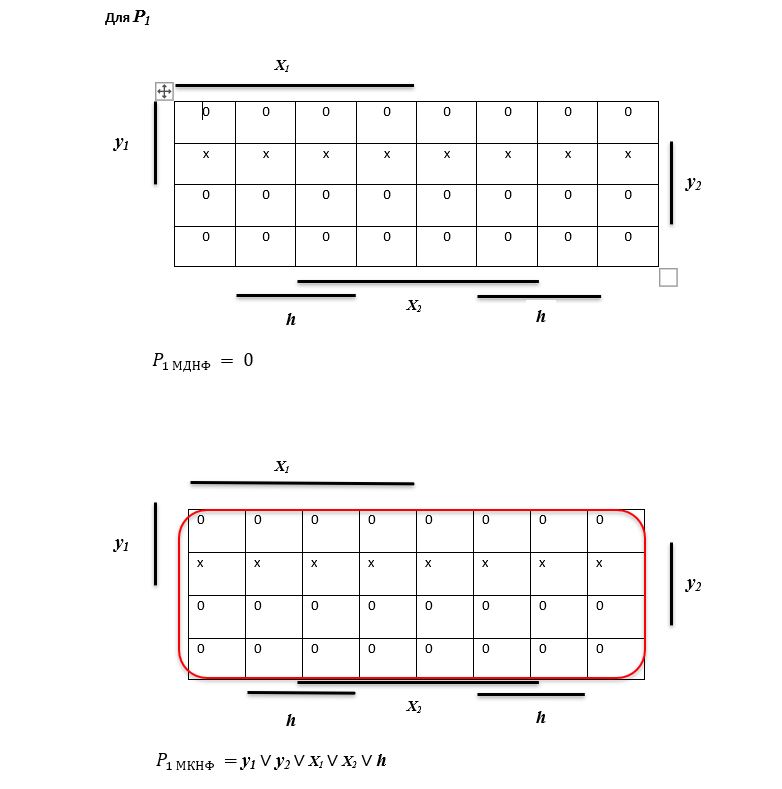
Разряды множимого закодированы: 0 – 01, 1 – 00, 2 – 11, 3 – 10.

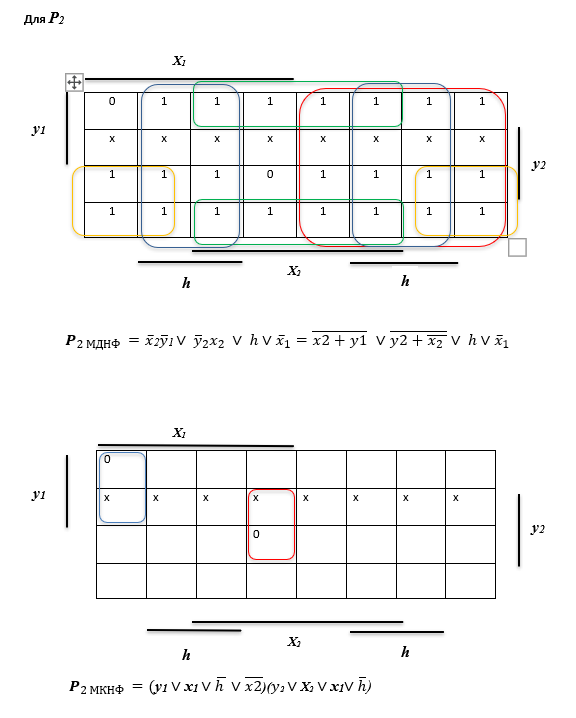
Таблица 3.1.1 - Таблица истинности ОЧУ

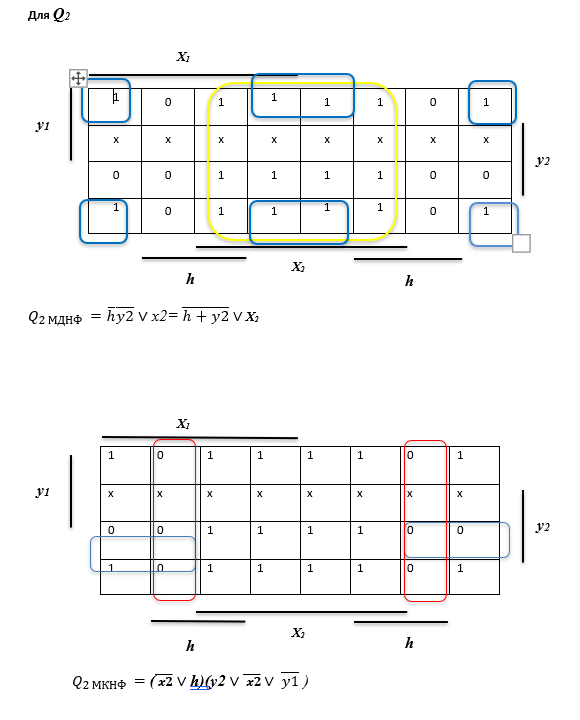
| **Мн**  **Мт** | | **Мт**  **Перенос** | | **Упр.** | **Ст. разряды** | | **Мл. разряды**  **Мт** | | **Результат операции**  **в четверичной с/c** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***X1*** | ***X2*** | ***y1*** | ***y2*** | ***h*** | ***P1*** | ***P2*** | ***Q1*** | ***Q2*** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 ∙ 0 = 00 |
| 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | Выход 01 |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 ∙ 1 = 01 |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | Выход 01 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 ∙ 2 = 02 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | Выход 01 |
| 0 | 0 | 1 | 1 | 0 | x | x | x | x | 1 ∙ 3 = 03 |
| 0 | 0 | 1 | 1 | 1 | x | x | x | x | Выход 01 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 3 ∙ 0 = 00 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | Выход 03 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 3 ∙ 1 = 03 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | Выход 03 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 3 ∙ 2 = 12 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | Выход 03 |
| 1 | 0 | 1 | 1 | 0 | x | x | x | x | 3 ∙ 3 = 21 |
| 1 | 0 | 1 | 1 | 1 | x | x | x | x | Выход 03 |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 ∙ 0 = 00 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | Выход 00 |
| 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 ∙ 1 = 00 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | Выход 00 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 ∙ 2 = 00 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | Выход 00 |
| 0 | 1 | 1 | 1 | 0 | x | x | x | x | 0 ∙ 3 = 00 |

*Продолжение таблицы 3.1.1*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| 0 | 1 | 1 | 1 | 1 | x | x | x | x | Выход 00 |
| 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 2 ∙ 0 = 00 |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | Выход 02 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 2 ∙ 1 = 02 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | Выход 02 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 2 ∙ 2 = 10 |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | Выход 02 |
| 1 | 1 | 1 | 1 | 0 | x | x | x | x | 2 ∙ 3 = 12 |
| 1 | 1 | 1 | 1 | 1 | x | x | x | x | Выход 02 |







Минимизацию функции Q1 проведём алгоритмом Рота:

|  |  |
| --- | --- |
| *L* | *N* |
| 00100 10001 10010 10011 10100 10101 11001 11010 11011 11101 | 00110 00111 10110 10111 11111 11110 01111 01110 |

В исходных множествах произвели склеивание кубов и получили:

|  |  |
| --- | --- |
| L | N |
| x0100 1x010 1x101 1x0x1 | xx11x |

Поиск простых импликант:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| C0\*C0 | x0100 | 1x010 | 1x101 | 1x0x1 | xx11x |
| x0100 | - |  |  |  |  |
| 1x010 | 10yy0 | - |  |  |  |
| 1x101 | 1010y | 1xyyy | - |  |  |
| 1x0x1 | 10y0y | 1x01y | 1xy01 | - |  |
| xx11x | x01y0 | 1xy10 | 1x1y1 | 1xy11 | - |
| A1 | 1010x x01x0 | 1x01x 1xx10 | 1xx01 1x1x1 | 1xx11 | Ø |

A1 = { 1010x; x01x0; 1x01x; 1xx10; 1xx01; 1x1x1; 1xx11 }

Z0 = { Ø }

B1 = { x0100; 1x010; 1x101; 1x0x1; xx11x }

C1 = { 1010x; x01x0; 1x01x; 1xx10; 1xx01; 1x1x1; 1xx11; 1x0x1; xx11x }

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| C1\*C1 | 1010x | x01x0 | 1x01x | 1xx10 | 1xx01 | 1x1x1 | 1xx11 |
| 1010x | - |  |  |  |  |  |  |
| x01x0 | 10100 | - |  |  |  |  |  |
| 1x01x | 10yyx | 10y10 | - |  |  |  |  |
| 1xx10 | 101y0 | 10110 | 1x010 | - |  |  |  |
| 1xx01 | 10101 | 1010y | 1x0y1 | 1xxyy | - |  |  |
| 1x1x1 | 10101 | 101xy | 1xy11 | 1x11y | 1x101 | - |  |
| 1xx11 | 101y1 | 1011y | 1x011 | 1xx1y | 1xxy1 | 1x111 | - |
| 1x0x1 | 10y01 | 10yxy | 1x011 | 1x01y | 1x001 | 1xyx1 | 1x011 |
| xx11x | 101yx | x0110 | 1xy1x | 1x110 | 1x1y1 | 1x111 | 1x111 |
| A2 | 101xx | 101xx | 1x0x1 1xx11 1xx1x | 1x11x 1xx1x 1x01x | 1xxx1 1x1x1 | 1xxx1 | Ø |

A2 = { 101xx; 1xx1x; 1xxx1 }

Z1 = { Ø }

B2 = { 1010x; x01x0; 1x01x; 1xx10; 1xx01; 1x1x1; 1xx11; 1x0x1; xx11x }

C2 = { 101xx; 1xx1x; 1xxx1; x01x0; xx11x }

|  |  |  |  |
| --- | --- | --- | --- |
| C2\*C2 | 101xx | 1xx1x | 1xxx1 |
| 101xx | - |  |  |
| 1xx1x | 1011x | - |  |
| 1xxx1 | 101x1 | 1xx11 | - |
| x01x0 | 101x0 | 10110 | 101xy |
| xx11x | 1011x | 1x11x | 1x111 |
| A3 | Ø | Ø | Ø |

**A3 = { Ø }**

Конечное множество простых импликант Z = { 101xx; 1xx1x; 1xxx1; x01x0; xx11x }

**Поиск L-экстремалей.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| z#(Z-z) | 101xx | 1xx1x | 1xxx1 | x01x0 | xx11x |
| 101xx | - | 11x1x 1x01x | 11xx1 1x0x1 | 001x0 | 0x11x x111x |
| 1xx1x | 1010x | - | 11x01 1x001 | 001x0 | 0x11x 0111x |
| 1xxx1 | 10100 | 11x10 1x010 | - | 001x0 | 0x11x 0111x |
| x01x0 | Ø | 11x10 1x010 | 11x01 1x001 | - | 0111x 0x111 0111x |
| xx11x | Ø | 11010 1x010 | 11x01 1x001 | 00100 | - |
| Остаток | Ø | 11010 1x010 | 11x01 1x001 | 00100 | 0111x 0x111 0111x |

**Далее для расчетов используется исходное множество L (без произведенного склеивания).**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| z#(Z-z) n L | 00100 | 10001 | 10010 | 10011 | 10100 | 10101 | 11001 | 11010 | 11011 | 11101 |
| 11010 | Ø | Ø | Ø | Ø | Ø | Ø | Ø | 11010 | Ø | Ø |
| 1x010 | Ø | Ø | 10010 | Ø | Ø | Ø | Ø | 11010 | Ø | Ø |
| 11x01 | Ø | Ø | Ø | Ø | Ø | Ø | 11001 | Ø | Ø | 11101 |
| 1x001 | Ø | 10001 | Ø | Ø | Ø | Ø | 11001 | Ø | Ø | Ø |
| 00100 | 00100 | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| 0111x | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| 0x111 | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| 0111x | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |

**Множество L-экстремалей E = { 1xx1x; 1xxx1; x01x0 }**

**Z´ = Z - E = { 101xx; xx11x }**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| L#E | 00100 | 10001 | 10010 | 10011 | 10100 | 10101 | 11001 | 11010 | 11011 | 11101 |
| 1xx1x | 00100 | 10001 | Ø | Ø | 10100 | 10101 | 11001 | Ø | Ø | 11101 |
| 1xxx1 | 00100 | Ø | Ø | Ø | 10100 | Ø | Ø | Ø | Ø | Ø |
| x01x0 | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| Остаток | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |

**Минимальное покрытие - множество L-экстремалей E = {1xx1x; 1xxx1; x01x0}**

**Q1мднф**= х1у2+х1h+!х2у1h.

**4. ЛОГИЧЕСКИЙ СИНТЕЗ ОДНОРАЗРЯДНОГО ЧЕТВЕРИЧНОГО СУММАТОРА НА ОСНОВЕ МУЛЬТИПЛЕКСОРА**

Мультиплексор – это логическая схема, имеющая *n* информационных входов, *m* управляющих входов и один выход. При этом должно выполняться условие *n* = .

Принцип работы мультиплексора состоит в следующем. На выход мультиплексора может быть пропущен без изменений любой (один) логический сигнал, поступающий на один из информационных входов. Порядковый номер информационного входа, значение которого в данный момент должно быть передано на выход, определяется двоичным кодом, поданным на управляющие входы.

Функции ОЧС зависят от пяти переменных. Удобно взять мультиплексор с тремя адресными входами, это позволит упростить одну нашу большую функцию от пяти аргументов до восьми функций от двух переменных. Функции от двух переменных достаточно просты для того, чтобы самостоятельно заметить их минимальную форму.

Синтез дополнительных логических схем для ПФ ОЧC приведен в таблице 4.1.

Таблица 4.1. – Таблица истинности для ОЧС на мультиплексорах

*Таблица 4.1 – Таблица синтеза дополнительных логических схем для ПФ ОЧС*

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***№*** | ***a1*** | ***a2*** | ***b1*** | ***b2*** | ***p*** | **П** | **Выход** |  | **Выход** |  | **Выход** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| 0 | 0 | 0 | 0 | 0 | 0 | *0* | *«0»* | *1* | *!b2+p* | *1* |  |
| 1 | 0 | 0 | 0 | 0 | 1 | *0* | *1* | *0* |
| 2 | 0 | 0 | 0 | 1 | 0 | *0* | *0* | *0* |
| 3 | 0 | 0 | 0 | 1 | 1 | *0* | *1* | *1* |
| 4 | 0 | 0 | 1 | 0 | 0 | x | x | x |
| 5 | 0 | 0 | 1 | 0 | 1 | x | x | x |
| 6 | 0 | 0 | 1 | 1 | 0 | X | x | X |
| 7 | 0 | 0 | 1 | 1 | 1 | x | X | x |
| 8 | 0 | 1 | 0 | 0 | 0 | *0* | *«0»* | *0* | !b2\*p | *0* |  |
| 9 | 0 | 1 | 0 | 0 | 1 | *0* | *1* | *1* |
| 10 | 0 | 1 | 0 | 1 | 0 | *0* | *0* | *1* |
| 11 | 0 | 1 | 0 | 1 | 1 | *0* | *0* | *0* |
| 12 | 0 | 1 | 1 | 0 | 0 | X | x | X |
| 13 | 0 | 1 | 1 | 0 | 1 | X | X | x |
| 14 | 0 | 1 | 1 | 1 | 0 | X | x | X |
| 15 | 0 | 1 | 1 | 1 | 1 | X | X | x |
| 16 | 1 | 0 | 0 | 0 | 0 | *1* | *!b2+p* | *0* | *b2\*!p* | *1* |  |
| 17 | 1 | 0 | 0 | 0 | 1 | *1* | *0* | *0* |
| 18 | 1 | 0 | 0 | 1 | 0 | *0* | *1* | *0* |
| 19 | 1 | 0 | 0 | 1 | 1 | *1* | *0* | *1* |
| 20 | 1 | 0 | 1 | 0 | 0 | X | x | X |
| 21 | 1 | 0 | 1 | 0 | 1 | x | X | X |
| 22 | 1 | 0 | 1 | 1 | 0 | X | x | X |
| 23 | 1 | 0 | 1 | 1 | 1 | X | X | x |
| 24 | 1 | 1 | 0 | 0 | 0 | *0* | !b2\*p | *1* | b2+!p | *0* |  |
| 25 | 1 | 1 | 0 | 0 | 1 | *1* | *0* | *1* |
| 26 | 1 | 1 | 0 | 1 | 0 | *0* | *1* | *1* |
| 27 | 1 | 1 | 0 | 1 | 1 | *0* | *1* | *0* |
| 28 | 1 | 1 | 1 | 0 | 0 | X | X | X |
| 29 | 1 | 1 | 1 | 0 | 1 | x | x | x |
| 30 | 1 | 1 | 1 | 1 | 0 | X | X | X |
| 31 | 1 | 1 | 1 | 1 | 1 | X | X | X |

Функциональная схема реализации ОЧС на мультиплексорах приведена в приложении Г.

**5 ЛОГИЧЕСКИЙ СИНТЕЗ ПРЕОБРАЗОВАТЕЛЯ МНОЖИТЕЛЯ (ПМ)**

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

*Таблица 5.1 - Таблица истинности ПМ.*

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

Минимизируем выходные функции картами Карно

*Таблица 5.2 – Минимизация функции P*

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

*Таблица 5.3 – Минимизация функции*

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

*Таблица 5.4 – Минимизация функции*

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

1. **ОЦЕНКА ЭФФЕКТИВНОСТИ МИНИМИЗАЦИИ ПЕРЕКЛЮЧАТЕЛЬНЫХ ФУНКЦИЙ**

Для проведения оценки эффективности минимизации переключательных функций необходимо посчитать цену схемы до минимизации и цену схемы после минимизации. Эффективность минимизации *k* определяется как:

Таблица 6.1 – Эффективность минимизации ОЧУ

|  |  |  |  |
| --- | --- | --- | --- |
| Вых.  схемы | Рассчитанная цена схемы | | Эфф.  мин. k |
| До минимизации | После минимизации |
| P1 | с=127 | с=1 | 127 |
| P2 | с=22\*5+22+5+1=138 | с=6+4+2+1=13 | 10,6 |
| Q1 | с=10\*6+5+1=66 | с=7+1+3+1=12 | 5,5 |
| Q2 | с=16\*5+16+5+1=102 | c=2+3+1+1=7 | 14,57 |

Таблица 6.2 – Эффективность минимизации ОЧС

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вых.  схемы | Рассчитанная цена схемы | | | Эфф.  мин. k |
| До минимизации | | После минимизации |
|  | c=4\*5+5+4+1=30 |  | с=3\*3=15 | 2 |
|  | с=8\*5=54 |  | с=19 | 1,8 |
|  | с=8\*5=54 |  | с=12+4+3+1=20 | 2,7 |

1. **ОЦЕНКА РЕЗУЛЬТАТОВ РАЗРАБОТКИ**

**Временные затраты на умножение**

Формула расчёта временных затрат на умножение:

, где

– время сдвига частичной суммы;

– время умножения на ОЧУ;

– время формирования единицы переноса в ОЧС;

– время преобразования множителя;

– время формирования дополнительного кода множимого.

**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-Вейча 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.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

Луцик Ю.А., Лукьянова И.В. – Учебное пособие по курсу "Арифметические и логические основы вычислительной техники". – Минск: БГУИР, 2014 г.

Луцик Ю.А., Лукьянова И.В. – Методические указания к курсовому проекту по курсу “Арифметические и логические основы вычислительной техники”. – Мн.: БГУИР, 2004 г.

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

Лысиков Б.Г. Арифметические и логические основы цифровых автоматов. Мн.: Вышейшая школа, 1980.

Лысиков Б.Г. Цифровая вычислительная техника. Мн.: 2003 г.

**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

**ПРИЛОЖЕНИЕ Е**

*(обязательное)*

Ведомость документов