Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет   
информатики и радиоэлектроники»

Военный факультет

Кафедра электронных вычислительных машин

Дисциплина: Арифметические и логические основы цифровых устройств

К ЗАЩИТЕ ДОПУСТИТЬ

И. В. Лукьянова

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

на тему

ПРОЕКТИРОВАНИЕ И ЛОГИЧЕСКИЙ СИНТЕЗ СУММАТОРА- УМНОЖИТЕЛЯ ДВОИЧНО-ЧЕТВЕРИЧНЫХ ЧИСЕЛ

БГУИР КП 1-40 02 01 105 ПЗ

Студент: Р.С Кочеров

Руководитель: И.В. Лукьянова

Минск 2023

Учреждение образования

«Белорусский государственный университет информатики   
и радиоэлектроники»

Факультет компьютерных систем и сетей

УТВЕРЖДАЮ

Заведующий кафедрой ЭВМ

Б. В. Никульшин

« » 20 г.

ЗАДАНИЕ

по курсовой работе студента

Кочерова Романа Сергеевича

1. Тема работы: «Проектирование и логический синтез сумматора-умножителя двоично-четвертичных чисел».
2. Срок сдачи студентом законченной работы: до 20 мая 2022 г.
3. Исходные данные к работе:
   1. исходные сомножители: Мн = 27, 32; Мт = 68, 15;
   2. алгоритм умножения: Г;
   3. метод умножения: умножение закодированного двоично-четверичного множимого на два разряда двоичного множителя одновременно в дополнительных кодах;
   4. коды четверичных цифр множимого для перехода к двоично-четвертичной системе кодирования; 04 – 00, 14 – 10, 24 – 01, 34 – 11;
   5. тип синтезируемого умножителя: 1;
   6. логический базис для реализации ОЧС(A2): И, Исключающее ИЛИ, константная “1”, метод минимизации –карты Вейча и алгоритм Рота;
   7. логический базис для реализации ОЧУ(A5): ИЛИ, ИЛИ-НЕ; метод минимизации – карты Карно.
4. Содержание пояснительной записки (перечень подлежащих разработке вопросов):

Введение. 1. Разработка алгоритм умножения. 2.Разработка структурной схемы сумматора-умножителя. 3.Разработка функциональных схем основных узлов сумматора-умножителя. 4. Оценка эффективности минимизации переключательных функций. 5.Синтез комбинационных схем устройств на основе мультиплексоров. 6.Логический синтез преобразователя множителя (ПМ) 7. 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. Одноразрядный четвертичный умножитель. Схема электрическая функциональная.
   4. Одноразрядный четвертичный сумматор. Реализация на мультиплексорах. Схема электрическая функциональная.
   5. Преобразователь множителя. Схема электрическая функциональная.

КАЛЕНДАРНЫЙ ПЛАН

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование этапов курсовой работы | Объём этапа, % | Срок выполнения этапа | Примечания |
| Разработка алгоритма умножения | 10 | 10.02-20.02 |  |
| Разработка структурной схемы сумматора-умножителя | 10 | 21.02-09.03 | С выполнением чертежа |
| Разработка функциональных схем основных узлов сумматора-умножителя | 50 | 10.03-30.04 | С выполнением чертежей |
| Синтез комбинационных схем устройств на основе мультиплексоров | 10 | 01.05-15.05 | С выполнением чертежа |
| Завершение оформления пояснительной записки | 20 | 15.05-20.05 |  |

Дата выдачи задания: 10 февраля 2022 г.

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_ И.В. Лукьянова

ЗАДАНИЕ ПРИНЯЛ К ИСПОЛНЕНИЮ \_\_\_\_\_\_\_\_\_\_\_\_\_\_Р.С. Кочеров

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ 5**](#_Toc132450754)

[**1. РАЗРАБОТКА АЛГОРИТМА УМНОЖЕНИЯ 6**](#_Toc132450755)

[**2. РАЗРАБОТКА СТРУКТУРНОЙ СХЕМЫ СУММАТОРА-УМНОЖИТЕЛЯ 9**](#_Toc132450756)

[**3. РАЗРАБОТКА ФУНКЦИОНАЛЬНЫХ СХЕМ ОСНОВНЫХ УЗЛОВ СУММАТОРА-УМНОЖИТЕЛЯ 12**](#_Toc132450757)

[**3.1. Логический синтез одноразрядного четверичного умножителя 12**](#_Toc132450758)

[**3.2Логический синтез одноразрядного четверичного сумматора 15**](#_Toc132450759)

[**4. ОЦЕНКА ЭФФЕКТИВНОСТИ МИНИМИЗАЦИИ ПЕРЕКЛЮЧАТЕЛЬНЫХ ФУНКЦИЙ 21**](#_Toc132450760)

[**5. СИНТЕЗ ОЧС НА ОСНОВЕ МУЛЬТИПЛЕКСОРА 22**](#_Toc132450761)

[**6. ЛОГИЧЕСКИЙ СИНТЕЗ ПРЕОБРАЗОВАТЕЛЯ МНОЖИТЕЛЯ**](#_Toc132450762)[**(ПМ) 24**](#_Toc132450763)

[**7. ОЦЕНКА РЕЗУЛЬТАТОВ РАЗРАБОТКИ 25**](#_Toc132450764)

[**ЗАКЛЮЧЕНИЕ 26**](#_Toc132450765)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 27**](#_Toc132450766)

[**ПРИЛОЖЕНИЕ А 28**](#_Toc132450767)

[**ПРИЛОЖЕНИЕ Б 29**](#_Toc132450768)

[**ПРИЛОЖЕНИЕ В 30**](#_Toc132450769)

[**ПРИЛОЖЕНИЕ Г 31**](#_Toc132450770)

[**ПРИЛОЖЕНИЕ Д 32**](#_Toc132450771)

[**ПРИЛОЖЕНИЕ Е 33**](#_Toc132450772)

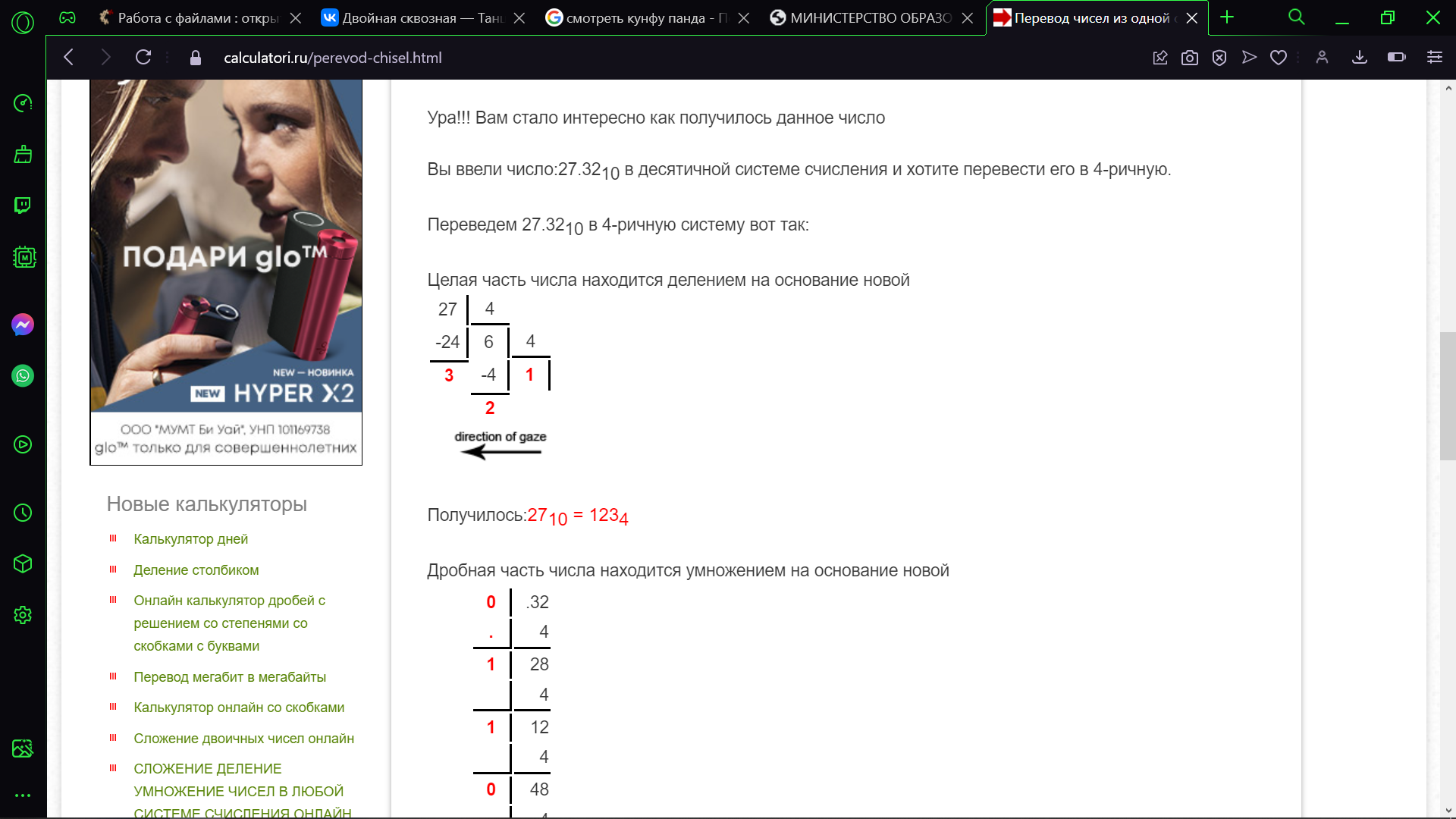
# **ВВЕДЕНИЕ**

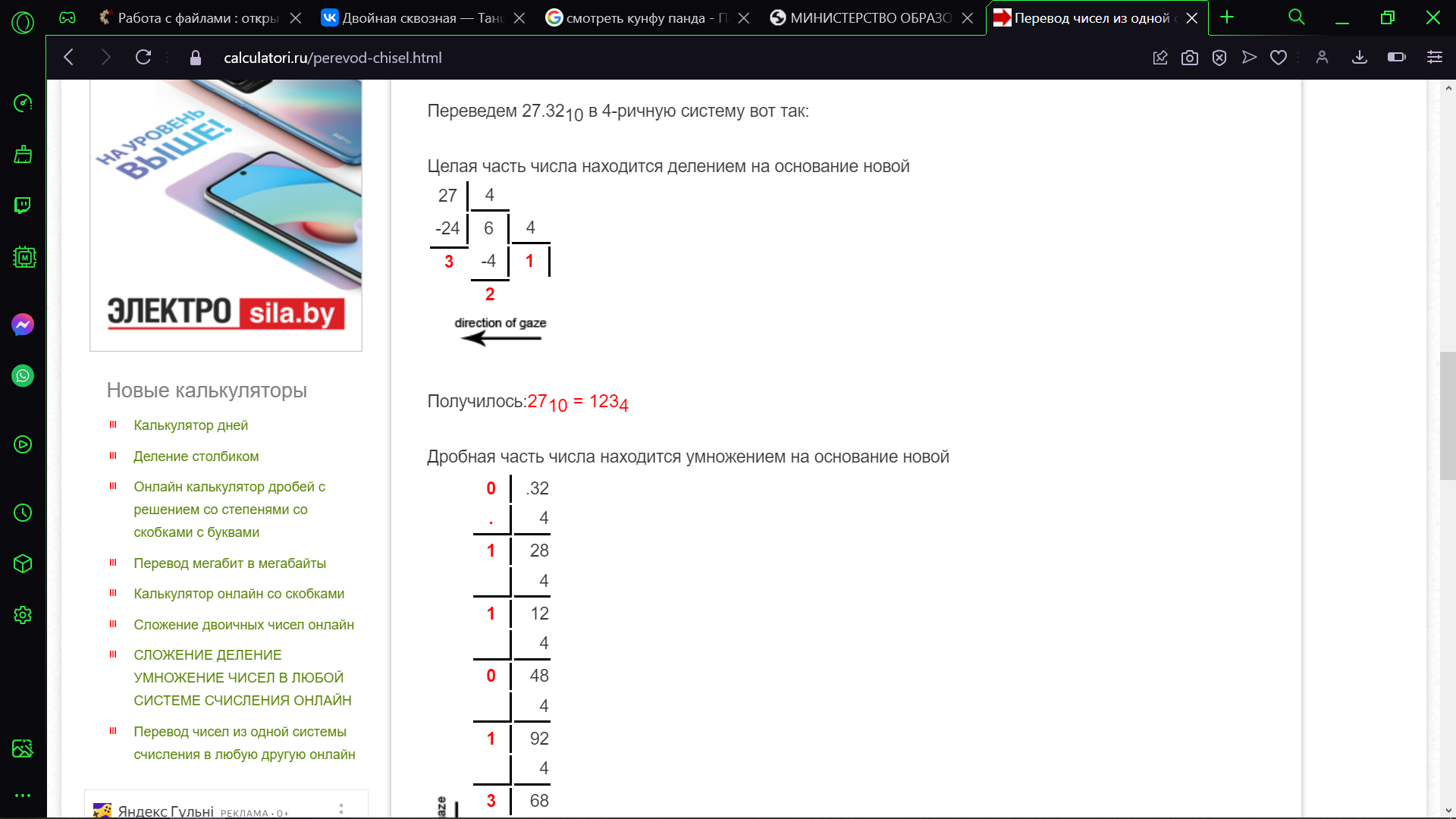
В курсовой работе поставлена цель разработать устройство одноразрядного четвертичного умножителя-сумматора. Арифметические операции сложения двоично-четверичных чисел с разными знаками в дополнительных кодах и умножения на 2 разряда множителя в прямых кодах должны выполняться одним цифровым устройством, именуемым **сумматор-умножитель**. Учитывая то, что суммирующие узлы обязательно входят в состав умножителя, начинать синтез нужно с разработки алгоритма умножения. Начать нужно с перевода чисел из одной системы счисления в другую и представить их в кодировке, данной по условию. Также поставлены задачи научиться минимизировать функции картами и алгоритмом Рота, проанализировать и выяснить преимущества и недостатки каждого из методов минимизации. Научиться понимать принципы работы таких устройств, как одноразрядный четверичный сумматор(ОЧС) и одноразрядный четверичный умножитель(ОЧУ). Поставлена задача выработать навыки проектирования схем для мультиплексоров, научиться оценивать время работы схемы и делать оценку эффективности работы переключательных функций.

**1. РАЗРАБОТКА АЛГОРИТМА УМНОЖЕНИЯ**

1. Перевод сомножителей из десятичной системы счисления в четвертичную (мантисса должна содержать шесть четвертичных разрядов).

**Множимое**



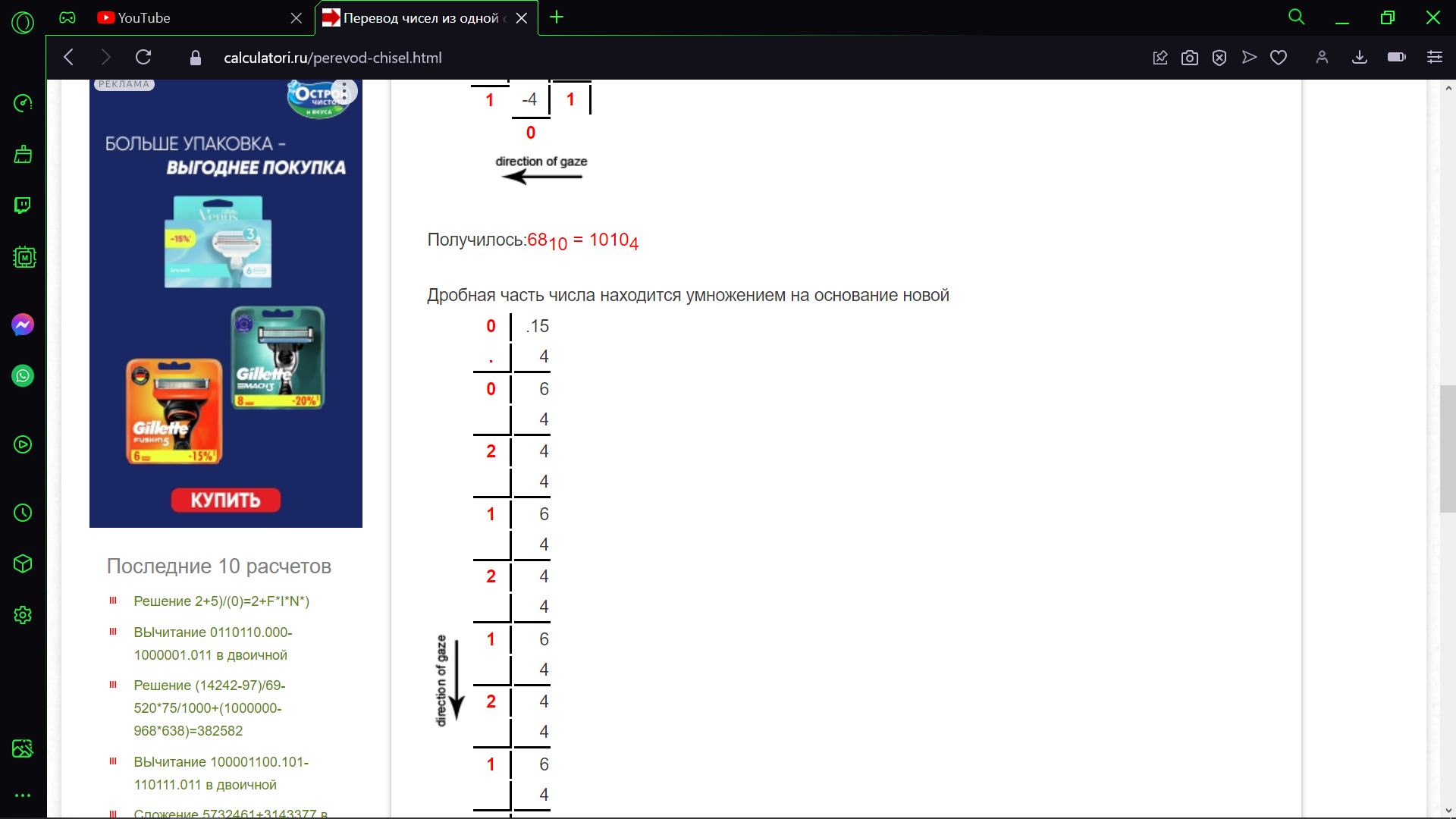


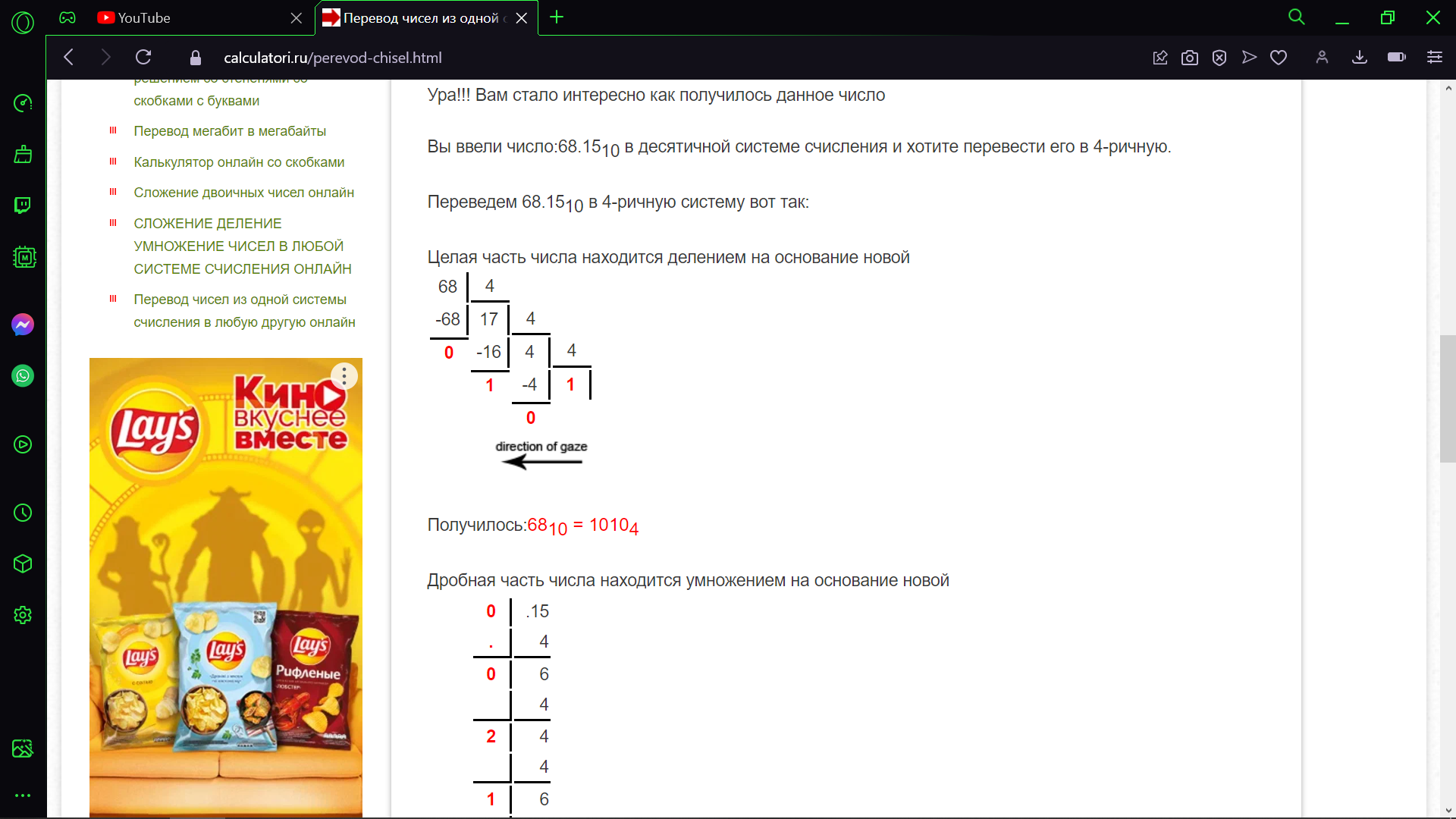
Мн4=123,110.

В соответствии с заданной кодировкой множимого:

Мн2/4=100111,101000.

**Множитель**





Мт4=1010, 02.

В соответствии с обычной весомозначной кодировкой множителя:

Мт2/4=01000100, 0010.

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

Мн = 0,100111101000

Мт = 0,010001000010

РМн = 0.0011 +034

РМт = 0.0100 +104

закодировано по заданию,

закодировано традиционно.

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

Порядок произведения будет следующим:

РМн  = 0.0011 +034

РМт  = 0.0100 +104

РМн\* Мт = 0.1011 +134

Результат закодирован в соответствии с заданием на кодировку множимого.

Знак произведения определяется суммой по модулю «два» знаков сомножителей, т. е.:

знМн знМт = 0 0 = 0.

Для умножения мантисс необходимо предварительно преобразовать множитель, чтобы исключить диаду 11 (34), заменив ее на триаду 101 и диаду 10 (24).

Преобразованный множитель имеет вид Перемножение мантисс по алгоритму «Г» приведено в таблице 1.

Таблица 1 – Перемножение мантисс

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Четвертичная с/с** | | | **Двоично-четвертичная с/с** | | | **Комментарии** |
| **1** | | | **2** | | | **3** |
| 0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  3.  0.  0.  0. | 000000  012311  012311  000000  012311  000123  013100  000000  013100  000001  013102  333333  013101  000000  013101 | 0  0  00  00  110  110  0000  1100  23110  00110  021120  022220  0000000  0222200 | 0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  1.  0.  0.  0. | 00 00 00 00 00 00  00 10 01 11 10 10  00 10 01 11 10 10  00 00 00 00 00 00  00 10 01 11 10 10  00 00 00 10 01 11  00 10 11 10 00 00  00 00 00 00 00 00  00 10 11 10 00 00  00 00 00 00 00 10  00 10 11 10 00 01  11 11 11 11 11 11  00 10 11 10 00 10  00 00 00 00 00 00  00 10 11 10 00 10 | 00  00  00 00  00 00  10 10 00  10 10 00  00 00 00 00  10 10 00 00  01 11 10 10 00  00 00 10 10 00  00 01 10 10 01 00  00 01 01 01 01 00  00 00 00 00 00 00 00  00 01 01 01 01 00 00 |  |

После окончания умножения необходимо оценить погрешность вычислений. Для этого полученное произведение (Мн ∙ Мт4 = 0, 0131010222200, РМн ∙ Мт = 7) приводится к нулевому порядку, а затем переводится в десятичную систему счисления:

Мн ∙ Мт4 = 131010, 2222, РМн ∙ Мт = 0;

Мн ∙ Мт10 = 1860, 664.

Результат прямого перемножения операндов даёт следующее значение:

Мн10 ∙ Мт10 = 1858, 858.

Абсолютная погрешность:

.

Относительная погрешность:

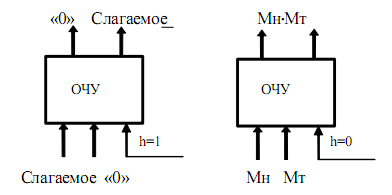
.

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

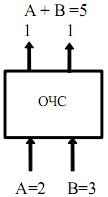
# **2. рАЗРАБОТКА СТРУКТУРНОЙ СХЕМЫ СУММАТОРА-УМНОЖИТЕЛЯ**

Структурная схема сумматора-умножителя 1-го типа для алгоритма умножения “Г” приведена на схеме 1.

Если устройство работает как сумматор (на входе mul/sum – “1”), то оба слагаемых последовательно (за 2 такта) заносятся в регистр множимого, а на управляющий вход формирователя дополнительного кода (ФДК) F2 поступает «1». Следует учесть, что числа представлены в форме с плавающей запятой. Поэтому, прежде чем складывать мантиссы, необходимо выровнять порядки. В блоке порядков необходимо обеспечить сравнение порядков, используя сумматор порядков, и в зависимости от знака результата сдвигать первое или второе слагаемое. Реализация сдвига мантиссы числа с меньшим порядком будет зависеть от используемого алгоритма умножения. Этим будет определяться порядок подачи слагаемых на операцию и то, где будет сдвигаться мантисса (в регистре множимого или в регистре результата). На выходах ФДК формируется дополнительный код одного из слагаемых с учетом знака. Это слагаемое может быть записано в регистр результата, при этом управляющие сигналы, поступающие на входы «h» всех ОЧУ, дают возможность переписать на выходы ОЧУ разряды слагаемого без изменений. При необходимости выравнивания порядков в регистре-аккумуляторе может выполняться сдвиг мантиссы первого слагаемого. Если на вход «h» поступает «0», то ОЧУ перемножает разряды Мн и Мт.



Одноразрядный четверичный сумматор предназначен для сложения двух двоично-четверичных цифр, подаваемых на его входы.



В ОЧС первое слагаемое складывается с нулем, так как на старших выходах ОЧУ будут формироваться только коды нуля. Затем первое слагаемое попадает в регистр-аккумулятор, который изначально обнулен. На втором такте второе слагаемое из регистра множимого через цепочку ОЧУ и ОЧС попадает в аккумулятор, где складывается с первым слагаемым. Таким образом, аккумулятор (накапливающий сумматор) складывает операнды и хранит результат. Разрядность аккумулятора должна быть на единицу больше, чем разрядность исходных слагаемых, чтобы предусмотреть возможность возникновения при суммировании переноса.

**Если устройство работает как умножитель** (на входе mul/sum – “0”), то множимое и множитель помещаются в соответствующие регистры, а на управляющий вход ФДК F2 поступает «0». Диада множителя поступает на входы преобразователя множителя (ПМ). Задачей ПМ является преобразование диады множителя в соответствии с алгоритмом преобразования. При этом в случае образования единицы переноса в старшую диаду множителя она должна быть учтена при преобразовании этой старшей диады (выход 1 ПМ). Выход 2 ПМ переходит в единичное состояние, если текущая диада содержит отрицание. В этом случае инициализируется управляющий вход F1 формирователя дополнительного кода (ФДК), и на выходах ФДК формируется дополнительный код множимого с обратным знаком (умножение на -1). Принцип работы ФДК в зависимости от управляющих сигналов приведен в табл.2.

Табл.2 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 Мн |
| 1 | 1 | Меняется знак слагаемого |

На выходах 3,4 ПМ формируются диады преобразованного множителя, которые поступают на входы ОЧУ вместе с диадами множимого.

ОЧУ предназначен лишь для умножения двух четверичных цифр. Если в процессе умножения возникает перенос в следующий разряд, необходимо предусмотреть возможность его прибавления. Для суммирования результата умножения текущей диады Мн · Мт с переносом из предыдущей диады предназначены ОЧС. Следовательно, чтобы полностью сформировать частичное произведение четверичных сомножителей, необходима комбинация цепочек ОЧУ и ОЧС. Частичные суммы формируются в аккумуляторе. На первом этапе он обнулен, и первая частичная сумма получается за счет сложения первого частичного произведения (сформированного на выходах ОЧС) и нулевой частичной суммы (хранящейся в аккумуляторе). Далее в аккумуляторе происходит сложение i-й частичной суммы с (i+1)-м частичным произведением, результат сложения сохраняется.

На четырех выходах ОЧУ формируется результат умножения диад

Мн·Мт. Максимальной цифрой в диаде преобразованного множителя является двойка, поэтому в старшем разряде произведения максимальной цифрой может оказаться только «1» :

3 · 2 = 1 2.

max maх

Мн Мт

Это означает, что на младшие входы ОЧС никогда не поступят диады цифр, соответствующие кодам «2» и «3», следовательно, в таблице истинности работы ОЧС будут содержаться 16 безразличных входных наборов. Количество тактов умножения определяется разрядностью Мт.

Структурная схема сумматора-умножителя представлена в приложении А.

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

**3.1. Логический синтез одноразрядного четверичного умножителя**

ОЧУ - это комбинационное устройство, имеющее 5 входов и 3 выхода:

* 2 разряда из регистра Мн, преобразованные в ФДК;
* 2 разряда из регистра Мт, преобразованные в ПМ;
* управляющий вход h.

Разряды множителя закодированы в обыкновенной двоичной с/с:

0 - 00; 1 - 01; 2 - 10; 3 - 11.

Разряды множимого закодированы в соответствии с заданием:

0 - 00; 1 - 10; 2 - 01; 3 - 11.

Управляющий вход h определяет тип операции:

0 - умножение закодированных цифр, поступивших на информационные входы ОЧУ.

1 - вывод на выходы без изменения значения разрядов, поступивших из регистра множимого.

Принцип работы ОЧУ описывается с помощью таблицы истинности.

**Таблица истинности ОЧУ.**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Мн** | | **Мт** | | **Упр.** | **Старшие разряды** | | **Младшие разряды** | | **Пример операции в четвертичной с/с** |
| ***x1*** | ***x2*** | ***y1*** | ***y2*** | ***h*** | ***P1*** | ***P2*** | ***P3*** | ***P4*** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 ∙ 0 = 00 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | Выход – код «02» |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 2 ∙ 1 = 02 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | Выход – код «02» |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 2 ∙ 2 = 10 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | Выход – код «02» |
| 0 | 1 | 1 | 1 | 0 | x | x | x | x | 2 ∙ 3 = 12 |
| 0 | 1 | 1 | 1 | 1 | x | x | x | x | Выход – код «02» |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 ∙ 0 = 00 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | Выход – код «01» |
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 ∙ 1 = 01 |
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | Выход – код «01» |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 ∙ 2 = 02 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | Выход – код «01» |
| 1 | 0 | 1 | 1 | 0 | x | x | x | x | 1 ∙ 3 = 03 |
| 1 | 0 | 1 | 1 | 1 | x | x | x | x | Выход – код «01» |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 ∙ 0 = 00 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | Выход – код «03» |
| 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 3 ∙ 1 = 03 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | Выход – код «03» |
| 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 3 ∙ 2 = 12 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | Выход – код «03» |
| 1 | 1 | 1 | 1 | 0 | x | x | x | x | 3 ∙ 3 = 21 |
| 1 | 1 | 1 | 1 | 1 | x | x | x | x | Выход – код «03» |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 ∙ 0 = 00 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | Выход – код «00» |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 ∙ 1 = 00 |
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | Выход – код «00» |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 ∙ 2 = 00 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | Выход – код «00» |
| 0 | 0 | 1 | 1 | 0 | x | x | x | x | 0 ∙ 3 = 00 |
| 0 | 0 | 1 | 1 | 1 | x | x | x | x | Выход – код «00» |

h +

+

+ +

+

В таблице выделено 8 безразличных набора, т.к. на входы ОЧУ из разрядов множителя не может поступить код 11.

Проведём минимизацию переключательных функций картами Вейча. Для проведения минимизации переключательных функций необходимо выделить те наборы переменных, где функция принимает единичное значение. Также необходимо включить и все безразличные наборы, то есть наборы, которые не могут прийти на вход схемы ОЧУ.

**Минимизация функции картами Карно:**

y1 y2 h

000

100

101

111

110

010

011

001

x1 x2

10

11

01

00

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

Минимизировав функцию, получим:

**Минимизация функции картами Карно:**

y1 y2 h

000

100

101

111

110

010

011

001

x1 x2

10

11

01

00

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | x | x |  |  |
|  |  |  |  | x | x |  |  |
|  |  |  |  | x | x |  |  |
|  |  |  |  | x | x |  |  |

Минимизировав функцию, получим:

**Минимизация функции картами Карно:**

y1 y2 h

000

100

101

111

110

010

011

001

x1 x2

10

11

01

00

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

Минимизировав функцию, получим:

**Минимизация функции картами Карно:**

y1 y2 h

000

100

101

111

110

010

011

001

x1 x2

10

11

01

00

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

Минимизировав функцию, получим:

Функциональная схема ОЧУ представлена в приложении Б.

**3.2Логический синтез одноразрядного четверичного сумматора**

ОЧС - это комбинационное устройство, имеющее 5 входов и 3 выхода:

* 2 разряда одного слагаемого (множимого);
* 2 разряда второго слагаемого (регистр результата);
* вход переноса из младшего ОЧС.

**Таблица истинности ОЧC.**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***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 | 0 | 0 | 0+0+0=00 |
| 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0+0+1=01 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0+1+0=01 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0+1+1=02 |
| 0 | 0 | 0 | 1 | 0 | х | х | х | 0+2+0=02 |
| 0 | 0 | 0 | 1 | 1 | х | х | х | 0+2+1=03 |
| 0 | 0 | 1 | 1 | 0 | х | х | х | 0+3+0=03 |
| 0 | 0 | 1 | 1 | 1 | х | х | х | 0+3+1=10 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1+0+0=01 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1+0+1=02 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1+1+0=02 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1+1+1=03 |
| 1 | 0 | 0 | 1 | 0 | х | х | х | 1+2+0=03 |
| 1 | 0 | 0 | 1 | 1 | х | х | х | 1+2+1=10 |
| 1 | 0 | 1 | 1 | 0 | х | х | х | 1+3+0=10 |
| 1 | 0 | 1 | 1 | 1 | х | х | х | 1+3+1=11 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 2+0+0=02 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 2+0+1=03 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 2+1+0=03 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 2+1+1=10 |
| 0 | 1 | 0 | 1 | 0 | х | х | х | 2+2+0=10 |
| 0 | 1 | 0 | 1 | 1 | х | х | х | 2+2+1=11 |
| 0 | 1 | 1 | 1 | 0 | х | х | х | 2+3+0=11 |
| 0 | 1 | 1 | 1 | 1 | х | х | х | 2+3+1=12 |
| 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 3+0+0=03 |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 3+0+1=10 |
| 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 3+1+0=10 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 3+1+1=11 |
| 1 | 1 | 0 | 1 | 0 | х | х | х | 3+2+0=11 |
| 1 | 1 | 0 | 1 | 1 | х | х | х | 3+2+1=12 |
| 1 | 1 | 1 | 1 | 0 | х | х | х | 3+3+0=12 |
| 1 | 1 | 1 | 1 | 1 | х | х | х | 3+3+1=13 |

+

+

+

Проведём минимизацию переключательных функций картами Вейча. Для проведения минимизации переключательных функций необходимо выделить те наборы переменных, где функция принимает единичное значение. Также необходимо включить и все безразличные наборы, то есть наборы, которые не могут прийти на вход схемы ОЧУ.

**Минимизация функции П картами Вейча:**

***a2***

***b2***

***a1***

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

***p***

***p***

***b1***

Минимизировав функцию, получим:

**Минимизация функции S1 картами Вейча:**

***a2***

***b2***

***a1***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 |  | 1 |  |  | 1 |  | 1 |
| x | х | x | x | x | x | х | х |
| x | х | x | x | x | x | х | х |
|  | 1 |  | 1 | 1 |  | 1 |  |

***p***

***p***

***b1***

Минимизировав функцию, получим:

=

**Минимизация функции алгоритмом Рота:**

Определим множество единичных кубов:

Для облегчения минимизации алгоритмом Рота, минимизируем множество безразличных кубов.

Используем карту Вейча:

***x2***

***x4***

***x1***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 |  |  |  |  |  |  |  |
| x | х | x | x | x | x | х | х |
| x | х | x | x | x | x | х | х |
|  |  |  |  |  |  |  |  |

***x5***

***x5***

***x3***

После минимизации получаем множество безразличных кубов:

.

Сформируем множество :

Первый шаг умножения () приведён в таблице:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| C0\*C0 | 00101 | 10001 | 10100 | 10101 | 01000 | 01001 | 01100 | 11000 |
| 00101 | - |  |  |  |  |  |  |  |
| 10001 |  | - |  |  |  |  |  |  |
| 10100 |  |  | - |  |  |  |  |  |
| 10101 | y0101 | 10y01 | 1010y | - |  |  |  |  |
| 01000 |  |  |  |  | - |  |  |  |
| 01001 |  |  |  |  | 0100y | - |  |  |
| 01100 |  |  |  |  | 01y00 |  | - |  |
| 11000 |  |  |  |  | y1000 |  |  | - |
| xxx1x | 001y1 | 100y1 | 101y0 | 101y1 | 010y0 | 010y1 | 011y0 | 110y0 |

В результате этой операции сформируется новое множество кубов:

Множество *Z0* кубов, не участвовавших в образовании новых кубов, пустое.

Выполним следующий шаг поиска простых импликант, выполнив \*:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| C1\*C1 | x0101 | 001x1 | 10x01 | 100x1 | 1010x | 101x0 | 101x1 | 0100x | 01x00 | x1000 | 010x0 | 010x1 | 011x0 | 110x0 |
| x0101 | - |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 001x1 |  | - |  |  |  |  |  |  |  |  |  |  |  |  |
| 10x01 |  |  | - |  |  |  |  |  |  |  |  |  |  |  |
| 100x1 |  |  |  | - |  |  |  |  |  |  |  |  |  |  |
| 1010x |  |  |  |  | - |  |  |  |  |  |  |  |  |  |
| 101x0 |  |  |  |  |  | - |  |  |  |  |  |  |  |  |
| 101x1 |  | y01x1 |  | 10yx1 |  | 101xy | - |  |  |  |  |  |  |  |
| 0100x |  |  |  |  |  |  |  | - |  |  |  |  |  |  |
| 01x00 |  |  |  |  |  |  |  |  | - |  |  |  |  |  |
| x1000 |  |  |  |  |  |  |  |  |  | - |  |  |  |  |
| 010x0 |  |  |  |  |  |  |  |  |  |  | - |  |  |  |
| 010x1 |  |  |  |  |  |  |  |  |  |  | 010xy | - |  |  |
| 011x0 |  |  |  |  |  |  |  |  |  |  | 01yx0 |  | - |  |
| 110x0 |  |  |  |  |  |  |  |  |  |  | y10x0 |  |  | - |
| xxx1x | x01y1 |  | 10xy1 |  | 101yx |  |  | 010yx | 01xy0 | x10y0 |  |  |  |  |

В результате образовалось множество кубов:

Множество кубов, не участвовавших в образовании новых кубов, пустое.

Выполним следующий шаг поиска простых импликант, выполнив \*:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| C2\*C2 | x01x1 | 10xx1 | 101xx | 010xx | 01xx0 | x10x0 |
| x01x1 | - |  |  |  |  |  |
| 10xx1 |  | - |  |  |  |  |
| 101xx |  |  | - |  |  |  |
| 010xx |  |  |  | - |  |  |
| 01xx0 |  |  |  |  | - |  |
| x10x0 |  |  |  |  |  | - |
| xxx1x |  |  |  |  |  |  |
| A3 | Ø | Ø | Ø | Ø | Ø | Ø |

Новых кубов (третей размерности) не образовалось.

Получено множество:

На этом заканчивается этап поиска простых импликант, т.к.. Множество простых импликант:

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

Поиск *L-*экстремалей представлен в таблице:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| z#(Z-z) | x01x1 | 10xx1 | 101xx | 010xx | 01xx0 | x10x0 |
| x01x1 | - | 100x1 | 101x0 | 010xx | 01xx0 | x10x0 |
| 10xx1 | 001x1 | - | 101x0 | 010xx | 01xx0 | x10x0 |
| 101xx | 001x1 | 100x1 | - | 010xx | 01xx0 | x10x0 |
| 010xx | 001x1 | 100x1 | 101x0 | - | 011x0 | 110x0 |
| 01xx0 | 001x1 | 100x1 | 101x0 | 010x1 | - | 110x0 |
| x10x0 | 001x1 | 100x1 | 101x0 | 010x1 | 011x0 | - |
| xxx1x | 00101 | 10001 | 10100 | 01001 | 01100 | 11000 |
| Остаток | 00101 | 10001 | 10100 | 01001 | 01100 | 11000 |

*L*-экстремалями стали следующие простые импликанты:

.

Необходимо проверить, нет ли среди полученных *L-*экстремалей таких, которые стали *L-*экстремалями за счёт безразличных кубов. Для этого в таблице кубы множества *L* пересекают с остатками простых импликант:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| z#(Z-z) n L | 00101 | 10001 | 10100 | 10101 | 01000 | 01001 | 01100 | 11000 |
| 00101 | 00101 | y0y01 Ø | y010y Ø | y0101 Ø | 0yy0y Ø | 0yy01 Ø | 0y10y Ø | yyy0y Ø |
| 10001 | y0y01 Ø | 10001 | 10y0y Ø | 10y01 Ø | yy00y Ø | yy001 Ø | yyy0y Ø | 1y00y Ø |
| 10100 | y010y Ø | 10y0y Ø | 10100 | 1010y Ø | yyy00 Ø | yyy0y Ø | yy100 Ø | 1yy00 Ø |
| 01001 | 0yy01 Ø | yy001 Ø | yyy0y Ø | yyy01 Ø | 0100y Ø | 01001 | 01y0y Ø | y100y Ø |
| 01100 | 0y10y Ø | yyy0y Ø | yy100 Ø | yy10y Ø | 01y00 Ø | 01y0y Ø | 01100 | y1y00 Ø |
| 11000 | yyy0y Ø | 1y00y Ø | 1yy00 Ø | 1yy0y Ø | y1000 Ø | y100y Ø | y1y00 Ø | 11000 |

По результатам таблицы *L-*экстремалями, не связанными с безразличными наборами, являются. Эти кубы обязательно должны войти в минимальное покрытие.

Далее необходимо проанализировать, какие из исходных единичных кубов не покрыты найденными *L-*экстремалями. Этот анализ осуществляется с помощью следующей таблицы:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| L#E | 00101 | 10001 | 10100 | 10101 | 01000 | 01001 | 01100 | 11000 |
| x01x1 | Ø | 10001 | 10100 | Ø | 01000 | 01001 | 01100 | 11000 |
| 10xx1 | Ø | Ø | 10100 | Ø | 01000 | 01001 | 01100 | 11000 |
| 101xx | Ø | Ø | Ø | Ø | 01000 | 01001 | 01100 | 11000 |
| 010xx | Ø | Ø | Ø | Ø | Ø | Ø | 01100 | 11000 |
| 01xx0 | Ø | Ø | Ø | Ø | Ø | Ø | Ø | 11000 |
| x10x0 | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |
| Остаток | Ø | Ø | Ø | Ø | Ø | Ø | Ø | Ø |

Минимальное покрытие - множество L-экстремалей

E ={x01x1; 10xx1; 101xx; 010xx; 01xx0; x10x0 }.

=

Функциональная схема ОЧС представлена в приложении В.

**4. 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:

Все рассчитанные данные сведены в таблицу 3 и 4

Табл. 3 Эффективность минимизации ОЧУ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вых.  схемы | Рассчитанная цена схемы | | | Эфф.  мин. k |
| До минимизации | | После минимизации |
| P1 | с=++=15 |  | с = +=4 | 3,75 |
| P2 | с= |  | с= | 0 |
| P3 | с=++=53 |  | с=+=6 | 8,83 |
| P4 | с=++=89 |  | с=++=23 | 3,87 |

Табл.4 Эффективность минимизации ОЧС

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вых.  схемы | Рассчитанная цена схемы | | | Эфф.  мин. k |
| До минимизации | | После минимизации |
|  | с=++=28 |  | с = +=12 | 2,33 |
|  | с=++=53 |  | с=++=19 | 2,79 |
|  | с=++=65 |  | с=++=11 | 5,9 |

# **5. 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-ми). Входы – это информационные входы мультиплексора. Входы – управляющие входы.

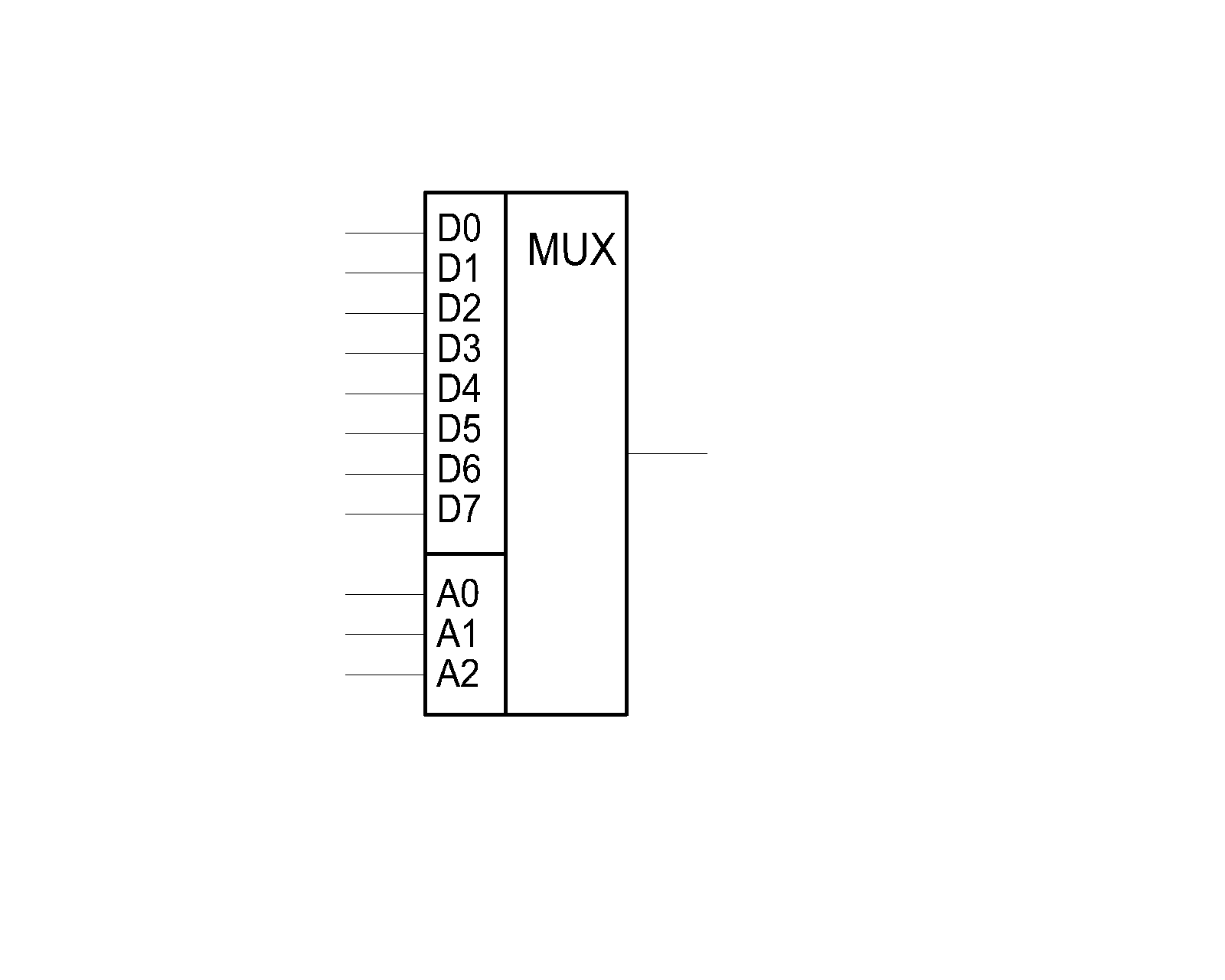


Рисунок 1 - Мультиплексор «один из восьми»

Используя таблицу истинности ОЧС, составим таблицу истинности для построения ОЧС на мультиплексорах.

Управление мультиплексором осуществляется тремя переменными: , а вход соответствующих значений функций на информационные входы обеспечивается реализацией этих функций на дополнительных логических элементах.

**Таблица истинности ОЧC на мультиплексорах.**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | p | П | Выход: |  | Выход: |  | Выход: |
| 0 | 0 | 0 | 0 | 0 | 0 | “0” | 0 | \*p | 0 | “0” |
| 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | x | x | x |
| 0 | 0 | 0 | 1 | 1 | x | x | x |
| 0 | 0 | 1 | 0 | 0 | 0 | “0” | 1 |  | 0 | \*p |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | х | х | х |
| 0 | 0 | 1 | 1 | 1 | х | х | х |
| 1 | 0 | 0 | 0 | 0 | 0 | “0” | 1 |  | 0 | \*p |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | x | x | x |
| 1 | 0 | 0 | 1 | 1 | x | x | x |
| 1 | 0 | 1 | 0 | 0 | 0 | “0” | 0 | \*p | 1 | “1” |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | х | х | х |
| 1 | 0 | 1 | 1 | 1 | х | х | х |
| 0 | 1 | 0 | 0 | 0 | 0 | “0” | 0 | \*p | 1 | “1” |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 0 | x | x | x |
| 0 | 1 | 0 | 1 | 1 | x | x | x |
| 0 | 1 | 1 | 0 | 0 | 0 | \*p | 1 |  | 1 |  |
| 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | х | х | х |
| 0 | 1 | 1 | 1 | 1 | х | х | х |
| 1 | 1 | 0 | 0 | 0 | 0 | \*p | 1 |  | 1 |  |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | x | x | x |
| 1 | 1 | 0 | 1 | 1 | x | x | x |
| 1 | 1 | 1 | 0 | 0 | 1 | “1” | 0 | \*p | 0 | “0” |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 | х | х | х |
| 1 | 1 | 1 | 1 | 1 | х | х | х |

Функциональная схема ОЧС на основе мультиплексора представлена в приложении Г.

**6. Логический синтез преобразователя множителя**

**(ПМ)**

Преобразователь множителя служит для исключения из множителя диад и , заменяя их на триады и соответственно.

Таблица истинности для ПМ. Таблица 8

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Входная диада | | Младший бит | Знак | Выходная диада | |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Минимизируем выходные функции , и .

Минимизацию проведём картами Карно.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Минимизация функции .   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | 00 | 01 | 11 | 010 | |  | | 0 |  |  |  |  | | 1 |  |  |  |  | | Минимизация функции .   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | 00 | 01 | 11 | 010 | |  | | 0 |  |  |  |  | | 1 |  |  |  |  | |
| Минимизация функции .   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | 00 | 01 | 11 | 10 | |  | | 0 |  |  |  |  | | 1 |  |  |  |  | |

# **7. ОЦЕНКА РЕЗУЛЬТАТОВ РАЗРАБОТКИ**

Формула расчёта временных затрат на умножение:

, где

– время сдвига частичной суммы;

– время формирования единицы переноса в ОЧУ;

– время формирования единицы переноса в ОЧС;

– время преобразования множителя;

– время формирования дополнительного кода множимого.

# **ЗАКЛЮЧЕНИЕ**

Во время выполнения проекта мы проделали огромную работу. Мы научились переводить числа из одной системы счисления в другую, умножать их алгоритмом Г, составлять структурные схемы, составлять таблицы истинности ОЧУ и ОЧС и минимизировать их. Познакомились с такими понятиями как коньюнктивная и дизьюнктивная нормальные формы записи функции. Научились минимизировать функции картами Карно\Вейча и алгоритмом Рота. У каждого метода минимизации есть свои плюсы и минусы. Алгоритм Рота очень громоздкий и неудобный для рассеянных людей, так как при каждом действии нужно быть предельно внимательным и аккуратным, чтобы не перепутать операции и не сбиться. С картами работать проще, но они не подойдут для большого количества переменных, так как в таком случае карты получатся очень большими и неудобными.

Данная работа была сделана для выработки полезных знаний и навыков, которые могут пригодиться нам в будущем, например в схемотехнике.

# 

# **список использованных источников**

1. Лысиков Б.Г. Арифметические и логические основы цифровых автома-

тов. Мн.: Вышейшая школа, 1980.

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

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

4. Луцик Ю.А., Лукьянова И.В.– Учебное пособие по курсу "Арифметические и логические основы вычислительной техники". -Мн.:ротапринт МРТИ ,2004 г.

**ПРИЛОЖЕНИЕ А**

Структурная схема сумматора-умножителя 1-го типа для алгоритма умножения «Г» на два разряда множителя одновременно

**ПРИЛОЖЕНИЕ Б**

Функциональная схема в заданном логическом базисе ОЧУ

**ПРИЛОЖЕНИЕ В**

Функциональная схема в заданном логическом базисе ОЧС

**ПРИЛОЖЕНИЕ Г**

Функциональная схема ОЧС на основе мультиплексора

**ПРИЛОЖЕНИЕ Д**

Функциональная схема в заданном логическом базисе ПМ

**ПРИЛОЖЕНИЕ Е**

Ведомость документов