**СОДЕРЖАНИЕ**

ВВЕДЕНИЕ ............................................................................................................ 5 1. РАЗРАБОТКА АЛГОРИТМА УМНОЖЕНИЯ .............................................. 6 2. РАЗРАБОТКА СТРУКТУРНОЙ СХЕМЫ СУММАТОРА-УМНОЖИТЕ-

ЛЯ ............................................................................................................................ 8

3. РАЗРАБОТКА ФУНКЦИОНАЛЬНЫХ СХЕМ ОСНОВНЫХ УЗЛОВ СУММАТОРА-УМНОЖИТЕЛЯ ......................................................................... 9 3.1. Логический синтез одноразрядного четверичного умножителя …......... 9 3.2. Логический синтез одноразрядного четверичного сумматора .................. 13

3.3. Минимизация функции *S1* алгоритмом Рота ……………...….…………… 17

3.4. Логический синтез преобразователя множителя …................................... 28 4. СИНТЕЗ КОМБИНАЦИОННЫХ СХЕМ УСТРОЙСТВ НА ОСНОВЕ МУЛЬТИПЛЕКСОРОВ …................................................................................... 30 5. ОЦЕНКА РЕЗУЛЬТАТОВ РАЗРАБОТКИ …................................................ 32 ЗАКЛЮЧЕНИЕ …................................................................................................ 33

СПИСОК ЛИТЕРАТУРЫ …................................................................................ 34 ПРИЛОЖЕНИЕ А …............................................................................................ 35 ПРИЛОЖЕНИЕ Б …............................................................................................. 36 ПРИЛОЖЕНИЕ В …………………………………………………………..….. 37

ПРИЛОЖЕНИЕ Г ................................................................................................. 38 ПРИЛОЖЕНИЕ Д ................................................................................................ 39

**ВВЕДЕНИЕ**

Предмет «Арифметические и логические основы вычислительной техники» является основополагающим в вопросах организации ЭВМ, а, следовательно, и неотъемлемой частью подготовки качественного специалиста в области информационных технологий.

Целью данной курсовой работы является проектирование такого устройства, как двоично-четверичный сумматор-умножитель (СУ). Сумматор является одним из центральных узлов арифметико-логического устройства (АЛУ), поэтому глубокое понимание принципов его работы критически важно для современного инженера. Чтобы спроектировать данное устройство, необходимо выполнить несколько последовательных этапов разработки:

– Разработка алгоритма умножения чисел, по которому работает СУ;

– Разработка структурной схемы СУ;

– Разработка функциональных схем основных узлов СУ;

– Оценка результатов проделанной работы;

– Оформление документации по проделанной работе.

В ходе выполнения курсовой работы автором были пройдены все эти этапы. В настоящей пояснительной записке изложено краткое описание процесса проектирования и приведена графическая документация по структурной схеме устройства и функциональным схемам основных её узлов.

**1. РАЗРАБОТКА АЛГОРИТМА УМНОЖЕНИЯ**

Перевод сомножителей из десятичной системы счисления в четверичную:

Множимое:

|  |  |
| --- | --- |
| × | 0,34  4 |
| × | 1,36  4 |
| × | 1,44  4 |
|  | 1,76 |

22 | 4

20 5 | 4

2 4 1

1

= 112,111

= 010100,010101

Множитель:

|  |  |
| --- | --- |
| × | 0,44  4 |
| × | 1,76  4 |
| × | 3,04  4 |
|  | 0,16 |

55 | 4

4 13 | 4

15 12 3

12 1

3

= 313,130

= 110111,011110

Запишем сомножители в форме с плавающей запятой в прямом коде:

Мн = 0, 010100010101 = 0.0011

Мт = 0, 110111011110 = 0.0011

Умножение двух чисел с плавающей запятой на 2 разряда множителя одновременно в прямых кодах сводится к сложению порядков, формированию знака произведения, преобразованию разрядов множителя согласно алгоритму и перемножению мантисс сомножителей.

|  |  |
| --- | --- |
| = 0.0011 | +34 |
| = 0.0011 | +34 |
| P = 0.0100 | +124 |

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, чтобы исключить диаду 11 (34), заменив ее на триаду 101.

Преобразованный множитель имеет вид: Мтп = 1121210

[Мн]д = 0,112111; [-Мн]д = 3,221223; [2Мн]д = 0,230222

*Умножение по алгоритму A:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Четверичная с/с | | Двоично-четверичная с/с | | Комментарии |
| 0,000000  0,000000  0,000000  0,000000  3,221223  3,221223  3,322122  0,230222  0,213010  0,021301  3,221223  3,303130  3,330313  0,230222  0,221201  0,022120  3,221223  3,310003  3,331000  0,112111  **0,103111** | 0  0  30  30  030  030  0030  0030  10030  10030  310030  **310030** | 0, 00 00 00 00 00 00  0, 10 10 10 10 10 10  0, 10 10 10 10 10 10  0, 10 10 10 10 10 10  1, 00 00 01 00 00 11  1, 00 00 01 00 00 11  1, 11 00 00 01 00 00  0, 00 11 10 00 00 00  0, 00 01 11 10 01 10  0, 10 00 01 11 10 01  1, 00 00 01 00 00 11  1, 11 10 11 01 11 10  1, 11 11 10 11 01 11  0, 00 11 10 00 00 00  0, 00 00 01 00 10 01  0, 10 00 00 01 00 10  1, 00 00 01 00 00 11  1, 11 01 10 10 10 11  1, 11 11 01 10 10 10  0, 01 01 00 01 01 01  **0, 01 10 11 01 01 01** | 10  10  11 10  11 10  10 11 10  10 11 10  10 10 11 10  10 10 11 10  01 10 10 11 10  01 10 10 11 10  11 01 10 10 11 10  **11 01 10 10 11 10** | 0  П1 = 0  1ч  1ч⋅ 4-1  П2 = b6 ⋅ [-Мн]д  2ч  2ч⋅ 4-1  П3 = b5 ⋅ [2Мн]д  3ч  3ч⋅ 4-1  П4 = b4 ⋅ [-Мн]д  4ч  4ч⋅ 4-1  П5 = b3 ⋅ [2Мн]д  5ч  5ч⋅ 4-1  П6 = b2 ⋅ [-Мн]д  6ч  5ч⋅ 4-1  П7 = b1 ⋅ [Мн]д  7ч |

После окончания умножения необходимо оценить погрешность вычислений. Для этого полученное произведение:

= 0,103111310030 ( 6)

приводится к нулевому порядку, а затем переводится в десятичную систему счисления:

= 103111,310030 ( 0)

= 1237,8154

Результат прямого перемножения операндов дает следующее значение:

= 1238,5296

Δ = 1238,5296 - 1237,8154 = 0,7142

δ; δ =

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**

Структура сумматора-умножителя первого типа строится на базе заданных узлов ОЧС, ОЧУ и аккумулятора (накапливающего сумматора).

Управление режимом работы схемы осуществляется внешним сигналом Mul/sum, который определяет вид текущей арифметической операции (умножение или суммирование).

Если устройство работает как сумматор (на входе Mul/sum – «1»), то оба слагаемых последовательно (за два такта) заносятся в регистр множимого, а на управляющий вход формирователя дополнительного кода (ФДК) F2 поступает «1».

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».

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 |
| *F*1 | *F*2 |
| 0 | 0 | Дополнительный код множимого |
| 0 | 1 | Дополнительный код слагаемого |
| 1 | 0 | Меняется знак множимого |
| 1 | 1 | Меняется знак слагаемого |

Структурная схема сумматора-умножителя первого типа для алгоритма умножения «А» приведена 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 (2 разряда из регистра Мн, 2 разряда из регистра Мт и управляющий вход h) и 4 выхода.

Разряды множителя закодированы : 0 - 00; 1 - 01; 2 - 10; 3 - 11.

Разряды множимого закодированы : 0 - 10; 1 - 01; 2 - 00; 3 - 11.

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.

Принцип работы ОЧУ описывается с помощью таблицы истинности

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

*Таблица 3.1* — *Таблица истинности ОЧУ:*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Мн | | Мт | | Уп. | Ст.разряды | | Мл.разряды | | Пример операции в четверичной с/с |
|  |  |  |  | *h* |  |  |  |  |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 2 ⋅ 0=00 |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | Выход - код «02» |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 2 ⋅ 1=02 |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | Выход - код «02» |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 2 ⋅ 2=10 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | Выход - код «02» |
| 0 | 0 | 1 | 1 | 0 | х | х | х | х | 2 ⋅ 3=12 |
| 0 | 0 | 1 | 1 | 1 | х | х | х | х | Выход - код «02» |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 ⋅ 0=00 |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | Выход - код «01» |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 ⋅ 1=01 |
| 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | Выход - код «01» |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 ⋅ 2=02 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | Выход - код «01» |
| 0 | 1 | 1 | 1 | 0 | х | х | х | х | 1 ⋅ 3=03 |
| 0 | 1 | 1 | 1 | 1 | х | х | х | х | Выход - код «01» |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 ⋅ 0=00 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | Выход - код «00» |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 ⋅ 1=00 |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | Выход - код «00» |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 ⋅ 2=00 |
| 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | Выход - код «00» |
| 1 | 0 | 1 | 1 | 0 | х | х | х | х | 0 ⋅ 3=00 |
| 1 | 0 | 1 | 1 | 1 | х | х | х | х | Выход - код «00» |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 3 ⋅ 0=00 |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | Выход - код «03» |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 3 ⋅ 1=03 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | Выход - код «03» |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 3 ⋅ 2=12 |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | Выход - код «03» |
| 1 | 1 | 1 | 1 | 0 | х | х | х | х | 3 ⋅ 3=21 |
| 1 | 1 | 1 | 1 | 1 | х | х | х | х | Выход - код «03» |

**Минимизация функции *P*1 картами Вейча:**

x2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** |
|  | **1** | **х** | **х** | **х** | **х** | **1** | **1** |
| **1** | **1** | **х** | **х** | **х** | **х** | **1** |  |
| **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** |

*x*1

*y*1

*y*2

*h*

*h*

Минимизировав функцию получим:

**Минимизация функции картами Вейча:**

x2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| **1** |  | **х** | **х** | **х** | **х** |  |  |
|  |  | **х** | **х** | **х** | **х** |  | **1** |
|  |  |  |  |  |  |  |  |

*x*1

*y*1

y2

h

h

Минимизировав функцию, получим:

**Минимизация функции картами Карно:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***x*1, *x*2**  ***y*1, *y*2, *h*** | | | **0** | **0** | **1** | **1** |
| **0** | **1** | **1** | **0** |
| **0** | **0** | **0** | **1** |  |  | **1** |
| **0** | **0** | **1** |  |  | **1** | **1** |
| **0** | **1** | **1** | **х** | **х** | **х** | **х** |
| **0** | **1** | **0** | **х** | **х** | **х** | **х** |
| **1** | **1** | **0** |  |  | **1** | **1** |
| **1** | **1** | **1** |  |  | **1** | **1** |
| **1** | **0** | **1** |  |  | **1** | **1** |
| **1** | **0** | **0** | **1** | **1** | **1** | **1** |

Минимизировав функцию, получим:

**Минимизация функции картами Карно:**

x2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **1** | **1** | **1** |  |  |  |  |
|  | **1** | **х** | **х** | **х** | **х** |  |  |
|  | **1** | **х** | **х** | **х** | **х** |  |  |
|  | **1** | **1** | **1** |  |  |  |  |

*x*1

*y*1

*h*

*h*

*y*2

Минимизировав функцию, получим:

Эффективность минимизаций:

*KP*1 = (22 · 5 + 22 + 5) / 17 = 8,1;

*KP*2 = (2 · 5 + 2 + 4) / 16 = 1;

*KP*3 = (14 · 5 + 14 + 4) / 22 = 4;

*KP*4 = (8 · 5 + 8 + 5) / 6 = 8,8.

Функциональная схема ОЧУ приведена в приложении Б.

**3.2. Логический синтез одноразрядного четверичного сумматора**

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

* 2 разряда одного слагаемого (множимого);
* 2 разряда второго слагаемого (множителя);
* вход переноса из младшего ОЧС;
* 3 выхода.

Принцип работы ОЧС описывается с помощью таблицы истинности

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

В таблице истинности выделено 16 безразличных наборов, так как на входы ОЧС со старших выходов ОЧУ не могут поступить коды «2» и «3».

*Таблица 3.2* — *Таблица истинности ОЧС:*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | *p* | *П* |  |  | Пример операции в четверичной с/с |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| 0 | 0 | 0 | 0 | 0 | х | х | х | 2+2+0=10 |
| 0 | 0 | 0 | 0 | 1 | х | х | х | 2+2+1=11 |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 2+1+0=03 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 2+1+1=10 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 2+0+0=02 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 2+0+1=03 |
| 0 | 0 | 1 | 1 | 0 | х | х | х | 2+3+0=11 |
| 0 | 0 | 1 | 1 | 1 | х | х | х | 2+3+1=12 |
| 0 | 1 | 0 | 0 | 0 | х | х | х | 1+2+0=03 |
| 0 | 1 | 0 | 0 | 1 | х | х | х | 1+2+1=10 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1+1+0=02 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1+1+1=03 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1+0+0=01 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1+0+1=02 |
| 0 | 1 | 1 | 1 | 0 | х | х | х | 1+3+0=10 |
| 0 | 1 | 1 | 1 | 1 | х | х | х | 1+3+1=11 |
| 1 | 0 | 0 | 0 | 0 | х | х | х | 0+2+0=02 |
| 1 | 0 | 0 | 0 | 1 | х | х | х | 0+2+1=03 |
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0+1+0=01 |
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0+1+1=02 |
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0+0+0=00 |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0+0+1=01 |
| 1 | 0 | 1 | 1 | 0 | х | х | х | 0+3+0=03 |
| 1 | 0 | 1 | 1 | 1 | х | х | х | 0+3+1=10 |
| 1 | 1 | 0 | 0 | 0 | х | х | х | 3+2+0=11 |
| 1 | 1 | 0 | 0 | 1 | х | х | х | 3+2+1=12 |
| 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 3+1+0=10 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 3+1+1=11 |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 3+0+0=03 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 3+0+1=10 |
| 1 | 1 | 1 | 1 | 0 | х | х | х | 3+3+0=12 |
| 1 | 1 | 1 | 1 | 1 | х | х | х | 3+3+1=13 |

**Минимизация функции *П* картами Вейча:**

*b*1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **х** | **х** |  |  | **х** | **х** |
|  | **1** | **х** | **х** | **1** | **1** | **х** | **х** |
|  |  | **х** | **х** |  |  | **х** | **х** |
|  |  | **х** | **х** |  | **1** | **х** | **х** |

*a*1

*a*2

*b*2

*p*

*p*

Минимизировав функцию получим:

**Минимизация функции *S*1 картами Карно:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***а*1, *a*2**  ***b*1, *b*2, *p*** | | | **0** | **0** | **1** | **1** |
| **0** | **1** | **1** | **0** |
| **1** | **0** | **0** |  |  | **1** | **1** |
| **1** | **0** | **1** | **1** |  | **1** |  |
| **1** | **1** | **1** | **х** | **х** | **х** | **х** |
| **1** | **1** | **0** | **х** | **х** | **х** | **х** |
| **0** | **1** | **0** | **1** |  | **1** |  |
| **0** | **1** | **1** | **1** | **1** |  |  |
| **0** | **0** | **1** | **х** | **х** | **х** | **х** |
| **0** | **0** | **0** | **х** | **х** | **х** | **х** |

Минимизировав функцию получим:

**Минимизация функции *S*2 картами Карно:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***a*1, *a*2**  ***b*1, *b*2, *p*** | | | **0** | **0** | **1** | **1** |
| **0** | **1** | **1** | **0** |
| **0** | **0** | **0** |  | **1** | **1** |  |
| **0** | **0** | **1** | **1** |  |  | **1** |
| **0** | **1** | **1** | **х** | **х** | **х** | **х** |
| **0** | **1** | **0** | **х** | **х** | **х** | **х** |
| **1** | **1** | **0** | **1** |  |  | **1** |
| **1** | **1** | **1** |  | **1** | **1** |  |
| **1** | **0** | **1** | **х** | **х** | **х** | **х** |
| **1** | **0** | **0** | **х** | **х** | **х** | **х** |

Минимизировав функцию, получим:

Эффективность минимизаций:

*KS*1 = (8 · 5 + 8 + 5) / 44 = 1,2;

*KS*2 = (8 · 5 + 8 + 5) / 28 = 1,9;

*KП* = (4 · 5 + 4 + 5) / 23 = 1,3.

**3.3. Минимизация функции *S1* алгоритмом Рота**

Определим множество единичных кубов:

00010, 10100

*L* = 00011, 11010

00101, 11100 .

01011, 11101

Определим множество безразличных наборов:

00000, 01000, 10000, 11000

*N* = 00001, 01001, 10001, 11001

00110, 01110, 10110, 11110 .

00111, 01111, 10111, 11111

Сформируем множество *C*0 *= L* U *N*

00010, 10100, 00000, 01000, 10000, 11000

*C*0 = 00011, 11010, 00001, 01001, 10001, 11001

00101, 11100, 00110, 01110, 10110, 11110 .

01011, 11101, 00111, 01111, 10111, 11111

Первым этапом алгоритма Рота является нахождение множества простых импликант.

Для реализации этого этапа будем использовать операцию умножения (\*) над множествами *С*0, *С*1 и т. д., пока в результате операции будут образовываться новые кубы большей размерности.

Первый шаг умножения (*С*0 \* *С*0) приведен в таблице 3.3.

В результате этой операции сформируется новое множество кубов:

0001*x*, 000*x*0, 00*x*10, 0*x*011, 000*x*1, 00*x*11, 00*x*01, 001*x*1

010*x*1, 01*x*11, 1*x*100, 10*x*00, 101*x*0, 110*x*0, 11*x*10, 1110*x*

*C*1 = 11*x*00, 111*x*0, 11*x*01, 111*x*1, 0000*x*, 0*x*000, *x*0000, 0*x*001

*x*0001, 0011*x*, 0*x*110, *x*0110, 0*x*111, *x*0111, 0100*x*, *x*1000 .

*x*1001, 0111*x*, *x*1110, *x*1111, 1000*x*, 1*x*000, 1*x*001, 1011*x*

1*x*110, 1*x*111, 1100*x*, 1111*x*

Множество *Z*0 кубов, не участвовавших в образовании новых кубов, пустое.

В таблице 3.4 приведён следующий шаг поиска простых импликант с помощью операции *С*1 \* *С*1.

В результате образовалось множество *С*2 кубов второй размерности:

000*xx*, 00*x*1*x*, 0*x*0*x*1, 00*xx*1, 0*xx*11, 1*xx*00

*C*2 = 1*x*1*x*0, 11*xx*0, 11*x*0*x*, 111*xx*, *x*000*x*, 0*x*00*x*

*xx*000, *xx*001, 0*x*11*x*, *x*011*x*, *xx*110, *xx*111 .

*x*100*x*, *x*111*x*, 1*x*00*x*, 1*x*11*x*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***C*0 *\* C*0** | **00010** | **00011** | **00101** | **01011** | **10100** | **11010** | **11100** | **11101** | **00000** | **00001** | **00110** | **00111** | **01000** | **01001** | **01110** | **01111** | **10000** | **10001** | **10110** | **10111** | **11000** | **11001** | **11110** | **11111** |
| **00010** | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00011** | 0001*x* | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00101** |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **01011** |  | 0*x*011 |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **10100** |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **11010** |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **11100** |  |  |  |  | 1*x*100 |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **11101** |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00000** | 000*x*0 |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00001** |  | 000*x*1 | 00*x*01 |  |  |  |  |  | 0000*x* | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00110** | 00*x*10 |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00111** |  | 00*x*11 | 001*x*1 |  |  |  |  |  |  |  | 0011*x* | ----- |  |  |  |  |  |  |  |  |  |  |  |  |
| **01000** |  |  |  |  |  |  |  |  | 0*x*000 |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |
| **01001** |  |  |  | 010*x*1 |  |  |  |  |  | 0*x*001 |  |  | 0100*x* | ----- |  |  |  |  |  |  |  |  |  |  |
| **01110** |  |  |  |  |  |  |  |  |  |  | 0*x*110 |  |  |  | ----- |  |  |  |  |  |  |  |  |  |
| **01111** |  |  |  | 01*x*11 |  |  |  |  |  |  |  | 0*x*111 |  |  | 0111*x* | ----- |  |  |  |  |  |  |  |  |
| **10000** |  |  |  |  | 10*x*00 |  |  |  | *x*0000 |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |
| **10001** |  |  |  |  |  |  |  |  |  | *x*0001 |  |  |  |  |  |  | 1000*x* | ----- |  |  |  |  |  |  |
| **10110** |  |  |  |  | 101*x*0 |  |  |  |  |  | *x*0110 |  |  |  |  |  |  |  | ----- |  |  |  |  |  |
| **10111** |  |  |  |  |  |  |  |  |  |  |  | *x*0111 |  |  |  |  |  |  | 1011*x* | ----- |  |  |  |  |
| **11000** |  |  |  |  |  | 110*x*0 | 11*x*00 |  |  |  |  |  | *x*1000 |  |  |  | 1*x*000 |  |  |  | ----- |  |  |  |
| **11001** |  |  |  |  |  |  |  | 11*x*01 |  |  |  |  |  | *x*1001 |  |  |  | 1*x*001 |  |  | 1100*x* | ----- |  |  |
| **11110** |  |  |  |  |  | 11*x*10 | 111*x*0 |  |  |  |  |  |  |  | *x*1110 |  |  |  | 1*x*110 |  |  |  | ----- |  |
| **11111** |  |  |  |  |  |  |  | 111*x*1 |  |  |  |  |  |  |  | *x*1111 |  |  |  | 1*x*111 |  |  | 1111*x* | ----- |

*Таблица 3.3* — *Поиск простых импликант (C0 \* С0).*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***C*1 \* *C*1** | **0001*x*** | **000*x*0** | **00*x*10** | **0*x*011** | **000*x*1** | **00*x*11** | **00*x*01** | **001*x*1** | **010*x*1** | **01*x*11** | **1*x*100** | **10*x*00** | **101*x*0** | **110*x*0** | **11*x*10** | **1110*x*** | **11*x*00** | **111*x*0** | **11*x*01** | **111*x*1** | **0000*x*** | **0*x*000** |
| **0001*x*** | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **000*x*0** |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00*x*10** |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0*x*011** |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **000*x*1** |  | 000*xx* |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00*x*11** |  |  | 00*x*1*x* |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00*x*01** |  |  |  |  |  | 00*xx*1 | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **001*x*1** |  |  |  |  | 00*x*11 |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **010*x*1** |  |  |  |  | 0*x*0*x*1 |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **01*x*11** |  |  |  |  |  | 0*xx*11 |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |
| **1*x*100** |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |
| **10*x*00** |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |
| **101*x*0** |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |
| **110*x*0** |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |
| **11*x*10** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |
| **1110*x*** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |
| **11*x*00** |  |  |  |  |  |  |  |  |  |  |  | 1*xx*00 |  |  | 11*x*0*x* |  | ----- |  |  |  |  |  |
| **111*x*0** |  |  |  |  |  |  |  |  |  |  |  |  | 1*x*1*x*0 | 11*xx*0 |  |  |  | ----- |  |  |  |  |
| **11*x*01** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 11*x*0*x* |  | ----- |  |  |  |
| **111*x*1** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 111*xx* |  | ----- |  |  |
| **0000*x*** | 000*xx* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |
| **0*x*000** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **x0000** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0x001** |  |  |  | 0*x*0*x*1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 0*x*00*x* |
| **x0001** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0011x** | 00*x*1*x* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0x110** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **x0110** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0x111** |  |  |  | 0*xx*11 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **x0111** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0100x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 0*x*00*x* |  |
| **x1000** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **x1101** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0111x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **x1110** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **x1111** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **1000x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | *x*000*x* |  |
| **1x000** |  |  |  |  |  |  |  |  |  |  | 1*xx*00 |  |  |  |  |  |  |  |  |  |  |  |
| **1x001** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **1011x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **1x110** |  |  |  |  |  |  |  |  |  |  | 1*x*1*x*0 |  |  |  |  |  |  |  |  |  |  |  |
| **1x111** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **1100x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 11*x*0*x* |  |  |  |  |  |  |
| **1111x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 111*xx* |  |  |  |  |  |  |

*Таблица 3.4 — Поиск*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***x*0000** | **0*x*001** | ***x*0001** | **0011*x*** | **0*x*110** | ***x*0110** | **0*x*111** | ***x*0111** | **0100*x*** | ***x*1000** | ***x*1001** | **0111*x*** | ***x*1110** | ***x*1111** | **1000*x*** | **1*x*000** | **1*x*001** | **1011*x*** | **1*x*110** | **1*x*111** | **1100*x*** | **1111*x*** |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| *x*000*x* |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | 0*x*11*x* |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  | *x*011*x* |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |
| *xx*000 |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | *xx*001 |  |  |  |  |  |  | *x*100*x* | ----- |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | 0*x*11*x* |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  | *xx*110 |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | *xx*111 |  |  |  |  | *x*111*x* | ----- |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |
|  | *xx*001 |  |  |  |  |  |  |  |  |  |  |  |  |  | 1*x*00*x* | ----- |  |  |  |  |  |
|  |  |  | *x*011*x* |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |
|  |  |  |  | *xx*110 |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |
|  |  |  |  |  |  | *xx*111 |  |  |  |  |  |  |  |  |  |  |  | 1*x*11*x* | ----- |  |  |
|  |  |  |  |  |  |  |  | *x*100*x* |  |  |  |  |  | 1*x*00*x* |  |  |  |  |  | ----- |  |
|  |  |  |  |  |  |  |  |  |  |  | *x*111*x* |  |  |  |  |  | 1*x*11*x* |  |  |  | ----- |

*простых импликант (С1 \* С1).*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***C*2 \* *C*2** | **000*xx*** | **00*x*1*x*** | **0*x*0*x*1** | **00*xx*1** | **0*xx*11** | **1*xx*00** | **1*x*1*x*0** | **11*xx*0** | **11*x*0*x*** | **111*xx*** | ***x*000*x*** | **0*x*00*x*** | ***xx*000** | ***xx*001** | **0*x*11*x*** | ***x*011*x*** | ***xx*110** | ***xx*111** | ***x*100*x*** | ***x*111*x*** | **1*x*00*x*** | **1*x*11*x*** |
| **000*xx*** | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00*x*1*x*** |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0*x*0*x*1** |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **00*xx*1** |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0*xx*11** |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **1*xx*00** |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **1*x*1*x*0** |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **11*xx*0** |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **11*x*0*x*** |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **111*xx*** |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |  |
| ***x*000*x*** |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |  |
| **0*x*00*x*** |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |  |
| ***xx*000** |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |  |  |
| ***xx*001** |  |  |  |  |  |  |  |  |  |  |  |  | *xx*00*x* | ----- |  |  |  |  |  |  |  |  |
| **0*x*11*x*** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |  |
| ***x*011*x*** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |  |
| ***xx*110** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ----- |  |  |  |  |  |
| ***xx*111** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | *xx*11*x* | ----- |  |  |  |  |
| ***x*100*x*** |  |  |  |  |  |  |  |  |  |  | *xx*00*x* |  |  |  |  |  |  |  | ----- |  |  |  |
| ***x*111*x*** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | *xx*11*x* |  |  |  | ----- |  |  |
| **1*x*00*x*** |  |  |  |  |  |  |  |  |  |  |  | *xx*00*x* |  |  |  |  |  |  |  |  | ----- |  |
| **1*x*11*x*** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | *xx*11*x* |  |  |  |  |  |  | ----- |

*Таблица 3.5* — *Поиск простых импликант (C2 \* C2).*

Множество *Z*0 кубов, не участвовавших в образовании новых кубов, пустое.

В таблице 3.5 приведён следующий шаг поиска простых импликант с помощью операции *С*2 \* *С*2.

В результате образовалось множество *С*3 кубов третьей размерности:

*C*3 = {*xx*00*x*, *xx*11*x*}.

Множество *Z*2 кубов, не участвовавших в образовании новых кубов имеет вид:

000*xx*, 00*x*1*x*, 0*x*0*x*1, 00*xx*1

*Z*2 = 0*xx*11, 1*xx*00, 1*x*1*x*0, 11*xx*0 .

11*x*0*x*, 111*xx*

Результат *С*3\**С*3 приведён в таблице 3.6.

|  |  |  |
| --- | --- | --- |
| ***С*3 \* *С*3** | ***xx*00*x*** | ***xx*11*x*** |
| ***xx*00*x*** | ----- |  |
| ***xx*11*x*** |  | ----- |

*Таблица 3.6* — *Поиск простых импликант (C3 \* C3).*

Новых кубов (четвёртой размерности) не образовалось.

Получено множество *Z*3:

*Z*3 = {*xx*00*x*, *xx*11*x*}.

На этом заканчивается этап поиска простых импликант, т. к. |*С*4| < 1. Множество простых импликант:

000*xx*, 00*x*1*x*, 0*x*0*x*1, 00*xx*1

*Z* = *Z*0 *U Z*1 U *Z*2 U *Z*3 = 0*xx*11, 1*xx*00, 1*x*1*x*0, 11*xx*0 .

11*x*0*x*, 111*xx*, *xx*00*x*, *xx*11*x*

Следующий этап – поиск *L*-экстремалей на множестве простых импликант (таблица 3.7). Для этого используется операция # (решётчатое вычитание).

Из каждой простой импликанты поочередно вычитаются все остальные простые импликанты *Z*#(*Z*\*z*), результат операции (последняя строка таблицы) указывает на то, что *L*-экстремалями стали следующие простые импликанты:

*E* = {00*xx*1, 11*xx*0}.

Необходимо проверить, нет ли среди полученных *L*-экстремалей таких,

которые стали *L*-экстремалями за счёт безразличных кубов. Для этого в таблице 6 из кубов множества *L* вычитаются остатки простых импликант, полученные в таблице 5 (результат выполнения операции *Z*#(*Z*\*z*)).

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Z*#(*Z\z*)** | **000*xx*** | **00*x*1*x*** | **0*x*0*x*1** | **00*xx*1** | **0*xx*11** | **1*xx*00** | **1*x*1*x*0** | **11*xx*0** | **11*x*0*x*** | **111*xx*** | ***xx*00*x*** | ***xx*11*x*** |
| **000*xx*** | ----- | *zz*1*zz*  0011*x* | *z*1*zzz*  010*x*1 | *zz*1*zz*  001*x*1 | *z*11*zz*  0*x*111 01*x*11 | *y*11*zz*  1*x1*00 11*x*00 | *y*11*zz*  1*x*1*x*0 | *yy*1*zz*  11*xx*0 | *yy*1*zz*  11*x*0*x* | *yyyzz*  111*xx* | 11*zzz*  *x*100*x*  1*x*00*x* | 11*yzz*  *xx*11*x* |
| **00*x*1*x*** | *zzz*0*z*  0000*x* | ----- | *zyz*0*z*  01001 | *zzz*0*z*  00101 | *z*1*zzz* *zyzzz*  01111 01*x*11 | *y*1*zyz* *yyzyz*  1*x*100 11*x*00 | *y*1*z*0*z*  1*x*100 111*x*0 | *yyzyz*  11*xx*0 | *yyzyz*  11*x*0*x* | *yyz*0*z*  111*xx* | 1*yzyz*  *y*1*zyz*  *x*100*x* 1*x*00*x* | 11*zzz*  *x*111*x* 1*x*11*x* |
| **0*x*0*x*1** | *zzzz*0  00000 | *zzyz*0  00110 | ----- | *zzyzz*  00101 | *zzyzz* *zz*1*zz*  01111 01111 | *yzyzy* *yz*1*zy*  1*x*100 11*x*00 | *yzyzy yzyzy*  1*x*100 111*x*0 | *yz1zy*  11*xx*0 | *yz*1*z*0  11*x*00 1110*x* | *yzyz*0  111xx | 1*zzz*0 *yzzz*0  1100*x* *x*1000 1*x*000 | 1*zyz*0 *yzyz*0  *x*1110 1111*x* 1*x*11*x* |
| **00*xx*1** | *zzzzy*  00000 | *zzzzy*  00110 | *zyzzz*  01001 | ----- | *zyzzz* *zyzzz*  01111 01111 | *y*1*zzy yyzzy*  1*x*10011*x*00 | *y1zzy yyzzy*  1*x*100 111*x*0 | *yyzzy*  11*xx*0 | *yzzyy*  *yyzz*0  11*x*00 11100 | *yyzz*0  111*xx* | *yyzz*0 1*yzzy* *y*1*zzy*  1100*x* *x*1000 1*x*000 | 1*yzzy* *yyzz*0 *y*1*zz*0  *x*1110 1111*x* 1*x*110 1111*x* |
| **0*xx*11** | *zzzyy*  00000 | *zzzzy*  00110 | *zzzyz*  01001 | *zzzyz*  00101 | ----- ----- | *yzzyy yzzyy*  1*x*100 11*x*00 | *yzzyy yzz*0*y*  1*x*100 111*x*0 | *yzz*0*y*  11*xx*0 | *yzzyy yzzyy*  11*x*00 11100 | *yzz*00  111*x*0 1110*x* | *yzzy*0 1*zzyy* *yzzyy*  1100*x* *x*1000 1*x*000 | 1*zzzy* *yzzz*0 *yzzzy* *yzzz*0  11110 11110 1*x*110 11110 |
| **1*xx*00** | *yzzzz*  00000 | *yzzyz*  00110 | *yzzzy*  01001 | *yzzzy*  00101 | *yzzyy yzzyy*  01111 01111 | ----- | *zzzzz*   *zzz*1*z*  ∅ 11110 | *zzz*1*z*  11*x*10 | *zzzzz zzzzz*  ∅ ∅ | *zzz*1*z* *zzzz*1  11110 11101 | *zzzz*1 0*zzzz* *zzzzz*  11001 01000 ∅ | *zzzyz zzzyz zzzyz zzzyz*  11110 11110 1*x*110 11110 |
| **1*x*1*x*0** | *yzyzz*  00000 | *yzzzz*  00110 | *yzyzy*  01001 | *yzzzy*  00101 | *yzzzy yzzzy*  01111 01111 | *zzzzz* *zz*0*zz*  ∅ 11000 | ----- | *zz0zz*  11010 |  | *zzzzz zzzzy*  ∅ 11101 | *zzyzy yzyzz*  11001 01000 | *zzzzz zzzzz zzzzz zzzzz*  ∅ ∅ ∅ ∅ |
| **11*xx*0** | *yyzzz*  00000 | *yyzzz*  00110 | *yzzzy*  01001 | *yyzzy*  00101 | *yzzzy yzzzy*  01111 01111 | *zzzzz*  ∅ | *zzzzz*  ∅ | ----- |  | *zzzzy*  11101 | *zzzzy yzzzz*  11001 01000 |  |
| **11*x*0*x*** | *yyzzz*  00000 | *yyzyz*  00110 | *yzzzz*  01001 | *yyzzz*  00101 | *yzzyz yzzyz*  01111 01111 |  |  | *zzzyz*  11010 | ----- | *zzzzz*  ∅ | *zzzzz yzzzz*  ∅ 01000 |  |
| **111*xx*** | *yyyzz*  00000 | *yyzzz*  00110 | *yzyzz*  01001 | *yyzzz*  00101 | *yzzzz yzzzz*  01111 01111 |  |  | *zzyzz*  11010 |  | ----- | *yzyzz*  01000 |  |
| ***xx*00*x*** | *zzzzz*  ∅ | *zzyyz*  00110 | *zzzzz*  ∅ | *zzyzz*  00101 | *zzyyz zzyyz*  01111 01111 |  |  | *zzzyz*  11010 |  |  | ----- ----- ----- |  |
| ***xx*11*x*** |  | *zzzzz*  ∅ |  | *zzzyz*  **00101** | *zzzzz zzzzz*  ∅ ∅ |  |  | *zzyzz*  **11010** |  |  | *zzyyz*  01000 | ----- ----- ----- ----- |

*Таблица 3.7*  — *Поиск L-экстремалей*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***L* ⋂ *E*** | **00010** | **00011** | **00101** | **01011** | **10100** | **11010** | **11100** | **11101** |
| **00101** | ∅ | ∅ | 00101 | ∅ | ∅ | ∅ | ∅ | ∅ |
| **11010** | ∅ | ∅ | ∅ | ∅ | ∅ | 11010 | ∅ | ∅ |

*Таблица 3.8 — Проверка L-экстремалей*

По результатам таблицы 3.8 *L*-экстремалями, не связанными с безразличными наборами, стали кубы 00*xx*1 и 11*хх*0 (остатки от вычитания из них всех остальных простых импликант – 00101 и 11010 – относятся к множеству единичных наборов *L* исходного задания функции). Эти кубы обязательно должны войти в минимальное покрытие.

Далее необходимо проанализировать, какие из исходных единичных кубов (множество *L*) не покрыты найденными *L*-экстремалями. Этот анализ осуществляется с помощью таблицы 3.9.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***L* # *E*** | **00010** | **00011** | **00101** | **01011** | **10100** | **11010** | **11100** | **11101** |
| **00*xx*1** | *zzzzy*  00010 | *zzzzz*  ∅ | *zzzzz*  ∅ | *zyzzz*  01011 | *yzzzy*  01011 | *yyzzy*  11010 | *yyzzy*  11100 | *yyzzz*  11101 |
| **11*xx*0** | *yyzzz*  00010 | *yyzzy*  00011 | *yyzzy*  00101 | *yzzzy*  01011 | *zyzzz*  10100 | *zzzzz*  ∅ | *zzzzz*  ∅ | *zzzzy*  11101 |

*Таблица 3.9*  — *Поиск непокрытых исходных наборов*

Из таблицы 3.9 видно, что *L*-экстремалями не покрыты четыре единичных куба (00010, 01011, 10100, 11101). Чтобы их покрыть, воспользуемся множеством простых импликант, не являющихся *L*-экстремалями (таблица 3.10).

Из таблицы 3.10 видно, что каждый из непокрытых единичных кубов может быть покрыт двумя равнозначными способами. Следовательно, существуют шестнадцать тупиковых (минимальных) форм:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***L* ⋂ *Z*** | **00010** | **01011** | **10100** | **11101** |
| **000*xx*** | 00010 | ∅ | ∅ | ∅ |
| **00*x*1*x*** | 00010 | ∅ | ∅ | ∅ |
| **0*x*0*x*1** | ∅ | 01011 | ∅ | ∅ |
| **0*xx*11** | ∅ | 01011 | ∅ | ∅ |
| **1*xx*00** | ∅ | ∅ | 10100 | ∅ |
| **1*x*1*x*0** | ∅ | ∅ | 10100 | ∅ |
| **11*x*0*x*** | ∅ | ∅ | ∅ | 11101 |
| **111*xx*** | ∅ | ∅ | ∅ | 11101 |
| ***xx*00*x*** | ∅ | ∅ | ∅ | ∅ |
| ***xx*11*x*** | ∅ | ∅ | ∅ | ∅ |

*Таблица 3.10*  — *Покрытие оставшихся кубов*

Функциональная схема ОЧС приведена в приложении В.

**3.4. Логический синтез преобразователя множителя**

Преобразователь множителя (ПМ) – это устройство, которое преобразовывает диады множителя в соответствии с методом умножения.

При умножении в дополнительных кодах ПМ заменяет диады 11 и 10 на триады 101 и 110 соответственно.

В случае образования единицы переноса в старшую диаду множителя она должна быть учтена при преобразовании следующей старшей диады (выход 1 ПМ), т.е. сохраняться до следующего такта на триггер.

Выход 2 ПМ переходит в единичное состояние, если текущая диада содержит отрицание (01).

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

Для случая умножения множимого на два разряда двоичного множителя одновременно в прямых кодах с использованием алгоритма умножения «А» будет происходит замена только диады 11

Принцип работы ПМ представлен с помощью таблицы истинности (таблица 3.11).

*Таблица 3.11* — *Таблица истинности ПМ:*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Входная  диада | | Младший  разряд | Перенос | Выходная  диада | | Знак |
| *a1* | *a2* | *p* | *h* | *S1* | *S2* | *Q* |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 |

Минимизацию переключательных функций произведём с помощью карт Вейча и реализуем их в базисе И, ИЛИ, НЕ.

**Минимизация функции *Q* картами Вейча:**

*a1*

|  |  |  |  |
| --- | --- | --- | --- |
| **1** |  |  |  |
|  | **1** |  |  |

*p*

*a2*

**Минимизация функции *S1* картами Вейча:**

*a*1

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | **1** |  |
| **1** |  |  |  |

*p*

*a2*

**Минимизация функции *S2* картами Вейча:**

*a1*

|  |  |  |  |
| --- | --- | --- | --- |
| **1** |  |  | **1** |
|  | **1** | **1** |  |

*p*

*a2*

Эффективность минимизаций:

*KQ* = (2 · 3 + 2 + 3) / 11 = 1;

*KS1* = (2 · 3 + 2 + 3) / 11 = 1;

*KS2* = (4 · 3 + 4 + 3) / 8 = 2,4;

Функциональная схема ПМ приведена в приложении Г.

**4. СИНТЕЗ КОМБИНАЦИОННЫХ СХЕМ УСТРОЙСТВ**

**НА ОСНОВЕ МУЛЬТИПЛЕКСОРОВ**

Мультиплексор – это логическая схема, которая имеет *n* информационных входов, *m* управляющих входов и один выход. При этом должно выполнятся условие *m* = 2*n*.

На выход мультиплексора может быть пропущен без изменений один любой 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, подаваемым на управляющие входы.

Переключательные функции (ПФ) от пяти переменных (как, например, ОЧС) можно реализовать на мультиплексоре «один из восьми». Управляющее поле такого мультиплексора будет определяться тремя переменными, следовательно, число групп с одинаковыми значениями этих переменных будет равно восьми. Также, реализация нескольких ПФ требует для каждой ПФ отдельного мультиплексора.

Для определения управляющего поля мультиплексора возьмём переменные *a1*, *a2*.

Таблица истинности для синтеза ПФ ОЧС приведена в таблице 4.1.

*Таблица 4.1 – Таблица истинности для синтеза ПФ ОЧС*

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***a*1** | ***a*2** | ***b*1** | ***b*2** | ***p*** | ***П*** | **Функция** | ***S*1** | **Функция** | ***S*2** | **Функция** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** |
| 0 | 0 | 0 | 0 | 0 | х | - | х | - | x | - |
| 0 | 0 | 0 | 0 | 1 | х | х | x |
| 0 | 0 | 0 | 1 | 0 | 0 | *b*2 ⋅ *p* | 1 | *b*2 + *p* | 1 | *b*2 ⊕ *p* |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | х | - | х | - | x | - |
| 0 | 0 | 1 | 1 | 1 | х | х | x |
| 0 | 1 | 0 | 0 | 0 | х | х | x |
| 0 | 1 | 0 | 0 | 1 | х | х | x |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | *b*2 ⋅ *p* | 0 | *b*2 ⊕ *p* |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | х | - | x | - | x | - |
| 0 | 1 | 1 | 1 | 1 | х | x | x |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** |
| 1 | 0 | 0 | 0 | 0 | х | - | x | - | x | - |
| 1 | 0 | 0 | 0 | 1 | х | x | x |
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | *b*2 + *p* | 1 | *b*2 ⊕ *p* |
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | х | - | x | - | x | - |
| 1 | 0 | 1 | 1 | 1 | х | x | x |
| 1 | 1 | 0 | 0 | 0 | х | x | x |
| 1 | 1 | 0 | 0 | 1 | х | x | x |
| 1 | 1 | 0 | 1 | 0 | 1 | *b*2 + *p* | 1 | *b*2 ⋅ *p* | 0 | *b*2 ⊕ *p* |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 | х | - | x | - | x | - |
| 1 | 1 | 1 | 1 | 1 | х | x | x |

Функциональная схема ОЧС на основе мультиплексоров представлена в приложении Д.

**5. ОЦЕНКА РЕЗУЛЬТАТОВ РАЗРАБОТКИ**

Формула расчёта временных затрат на умножение выглядит следующим образом:

T = n · (TПМ + TФДК + (n + 1) · TОЧУ + TОЧС + Tсдвига), где

TПМ – время преобразования множителя;

TФДК – время формирования дополнительного кода;

TОЧУ – время умножение на ОЧУ;

TОЧС – время формирования единицы переноса в ОЧС;

Tсдвига – время сдвига частичной суммы;

n – количество разрядов множителя.

**ЗАКЛЮЧЕНИЕ**

В ходе выполнения данной курсовой работы были разработаны структурная схема сумматора-умножителя и функциональные схемы основных узлов данного устройства. Для упрощения и уменьшения стоимости логических схем были выполнены минимизации переключательных функций различными способами. Это позволило выявить достоинства и недостатки каждого из алгоритмов.

В качестве главных достоинств карт Карно–Вейча можно выделить простоту минимизации и минимальные затраты времени. Однако использование данного метода для функций многих переменных будет весьма затруднительно.

Для минимизации функций многих переменных удобно использовать алгоритм Рота, который полностью формализует алгоритмы минимизации и делает минимизацию доступной для выполнения компьютерной программой.

Функциональные схемы были построены в различных логических базисах. Это позволило закрепить теоретические знания об основных законах булевой алгебры как, например, правил де Моргана.

Реализация переключательных функций на основе мультиплексоров позволяет облегчить минимизацию этих функций и значительно упростить функциональную схему устройства.

**СПИСОК ЛИТЕРАТУРЫ**

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

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

3 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.

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

5 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 первого типа. Схема электрическая структурная

**ПРИЛОЖЕНИЕ Б**

*(обязательное)*

Одноразрядный четверичный умножитель. Схема электрическая функциональная

**ПРИЛОЖЕНИЕ В**

*(обязательное)*

Одноразрядный четверичный сумматор. Схема электрическая функциональная

**ПРИЛОЖЕНИЕ Г**

*(обязательное)*

Преобразователь множителя. Схема электрическая функциональная

**ПРИЛОЖЕНИЕ Д**

*(обязательное)*

Одноразрядный четверичный сумматор. Реализация на мультиплексорах. Схема электрическая функциональная