**Национальный исследовательский ядерный университет «МИФИ»**

|  |  |
| --- | --- |
|  | **Институт Интеллектуальных Кибернетических Систем**  **Кафедра «Компьютерные системы и технологии»** |

**Пояснительная записка**

**к курсовому проекту**

**на тему «Проектирование процессора ЭВМ»**

Студент гр. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_­\_\_\_\_ / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /

Москва 2019

Оглавление

[*1.1* *Примеры и алгоритмы выполнения операций* 3](#_Toc27327451)

[1.1.1 Алгоритм операции УМНОЖЕНИЕ 4](#_Toc27327452)

[1.1.2 Алгоритм операции ПЕРЕСЫЛКА ОТРИЦАТЕЛЬНАЯ 8](#_Toc27327453)

[1.3.4 Сумматор СМ 15](#_Toc27327454)

1. **Проектирование блока операций**

## *Примеры и алгоритмы выполнения операций*

Формат представления чисел

|  |  |  |  |
| --- | --- | --- | --- |
| 0 разряд | 1 разряд | 2 разряд | 3 разряд |
| Зн **А** | **A2** ст | **А1** | **А0** мл |
| Зн **B** | **B2** ст | **B1** | **B0** мл |
| Зн **РВ** | **РВ2** ст | **РВ1** | **РВ0** мл |
| Зн **РА** | **РА2** ст | **РА1** | **РА0** мл |
| Зн **РР** | **РР2** ст | **РР1** | **РР0** мл |

В 0 разряде хранится знак числа. Это позволяет выполнять операцию сдвига таким образом, чтобы в мнимом разряде b4 оказалось значение b3 и так далее. Таким образом при выполнении операции умножения достаточно брать старшие разряды регистра RB.

## Алгоритм операции УМНОЖЕНИЕ

Умножение в соответствии с заданием выполняется по алгоритму умножения чисел в дополнительном коде с младших разрядов множителя и сдвигом суммы частичных произведений вправо путем последовательного преобразования множителя. Данный алгоритм можно выразить следующей формулой:

Где , k = 0..n

Для четырехразрядных чисел эта формула выглядит следующим образом:

Поясним представленный алгоритм примерами:

Пример 1:

= 1.010 =

= 0.110 =

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | 0 | 0 | 0 | 0 | RR = 0 |
| k = 3 |  | пропуск | | | |  |
|  | + | 0 | 0 | 0 | 0 | RR = RR \* |
| k= 2 | 0 | 1 | 1 | 0 | RA = RA \* |
|  |  | 0 | 1 | 1 | 0 | RR = RR + RA |
|  |  | 0 | 0 | 1 | 1 | RR = RR \* |
| k= 1 |  | пропуск | | | |  |
|  | + | 0 | 0 | 0 | 1 | RR = RR \* |
| k = 0 | 1 | 0 | 1 | 0 | RA = RA \* |
|  |  | 1 | 0 | 1 | 1 | RR = RR + RA |

Пример 2:

= 0.101 =

= 1.011 =

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | + | 0 | 0 | 0 | 0 | RR = 0 |
| k = 3 | 1 | 0 | 1 | 1 | RA = RA \* |
|  |  | 1 | 0 | 1 | 1 | RR = RR + RA |
|  |  | 1 | 1 | 0 | 1 | RR = RR \* |
| k = 2 |  | пропуск | | | |  |
|  | + | 1 | 1 | 1 | 0 | RR = RR \* |
| k = 1 | 0 | 1 | 0 | 1 | RA = RA \* |
|  |  | 0 | 0 | 1 | 1 | RR = RR + RA |
|  | + | 0 | 0 | 0 | 1 | RR = RR \* |
| k = 0 | 1 | 0 | 1 | 1 | RA = RA \* |
|  |  | 1 | 1 | 0 | 0 | RR = RR + RA |

Пример 3:

= 0.111 =

= 1.111 =

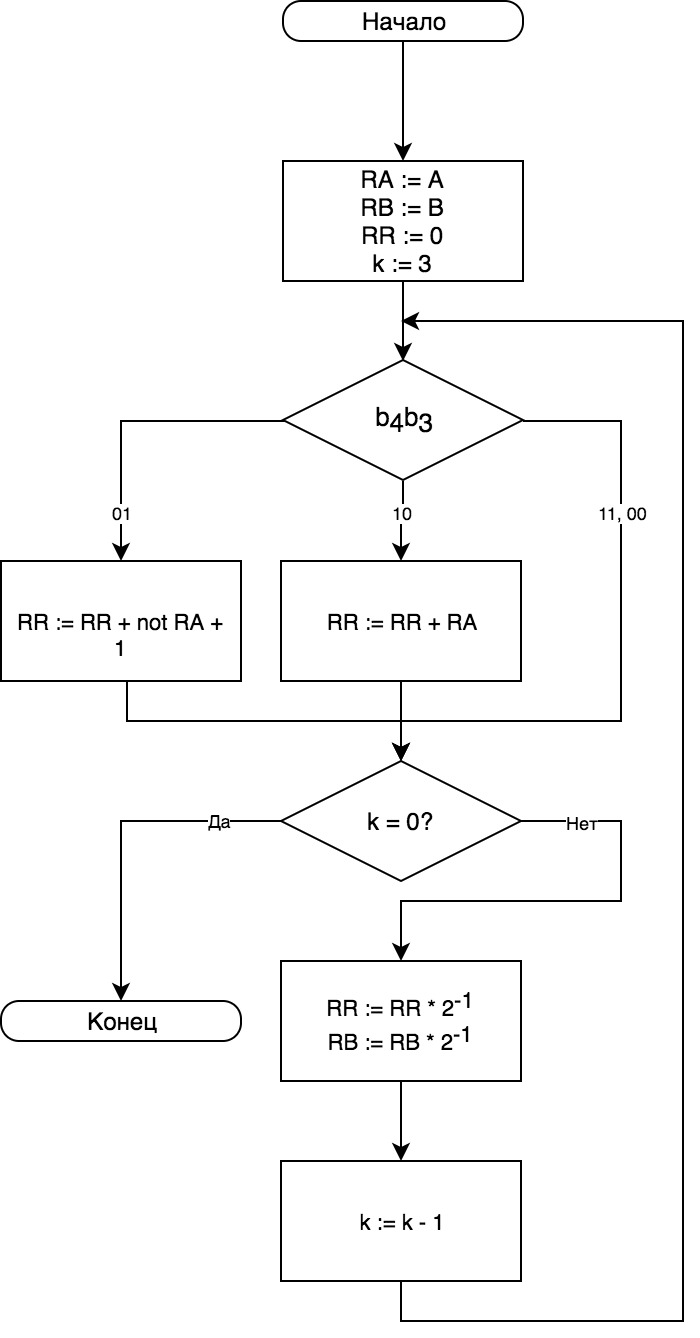
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | + | 0 | 0 | 0 | 0 | RR = 0 |
| k = 3 | 1 | 0 | 0 | 1 | RA = RA \* |
|  |  | 1 | 0 | 0 | 1 | RR = RR + RA |
|  | 1 | 1 | 0 | 0 | RR = RR \* |
| k = 2 | пропуск | | | |  |
|  |  | 1 | 1 | 1 | 0 | RR = RR \* |
| k = 1 |  | пропуск | | | |  |
|  |  | 1 | 1 | 1 | 1 | RR = RR \* |
| k = 0 |  | пропуск | | | |  |
|  |  | 1 | 1 | 1 | 1 | RR |

Пример 3:

= 0.111 =

= 0.101 =

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | + | 0 | 0 | 0 | 0 | RR = 0 |
| k = 3 | 1 | 0 | 0 | 1 | RA = RA \* |
|  |  | 1 | 0 | 0 | 1 | RR = RR + RA |
|  |  | 1 | 1 | 0 | 0 | RR = RR \* |
| k = 2 | + | 0 | 1 | 1 | 1 | RA = RA \* |
|  |  | 0 | 0 | 1 | 1 | RR = RR + RA |
|  |  | 0 | 0 | 0 | 1 | RR = RR \* |
| k = 1 | + | 1 | 0 | 0 | 1 | RA = RA \* |
|  |  | 1 | 0 | 1 | 0 | RR = RR + RA |
|  |  | 1 | 1 | 0 | 1 | RR = RR \* |
| k = 0 | + | 0 | 1 | 1 | 1 | RA = RA \* |
|  |  | 0 | 1 | 0 | 0 | RR = RR + RA |



A

B

C

D

E

F

G

1 2

*Рис. 1.1 Алгоритм операции УМНОЖЕНИЕ*

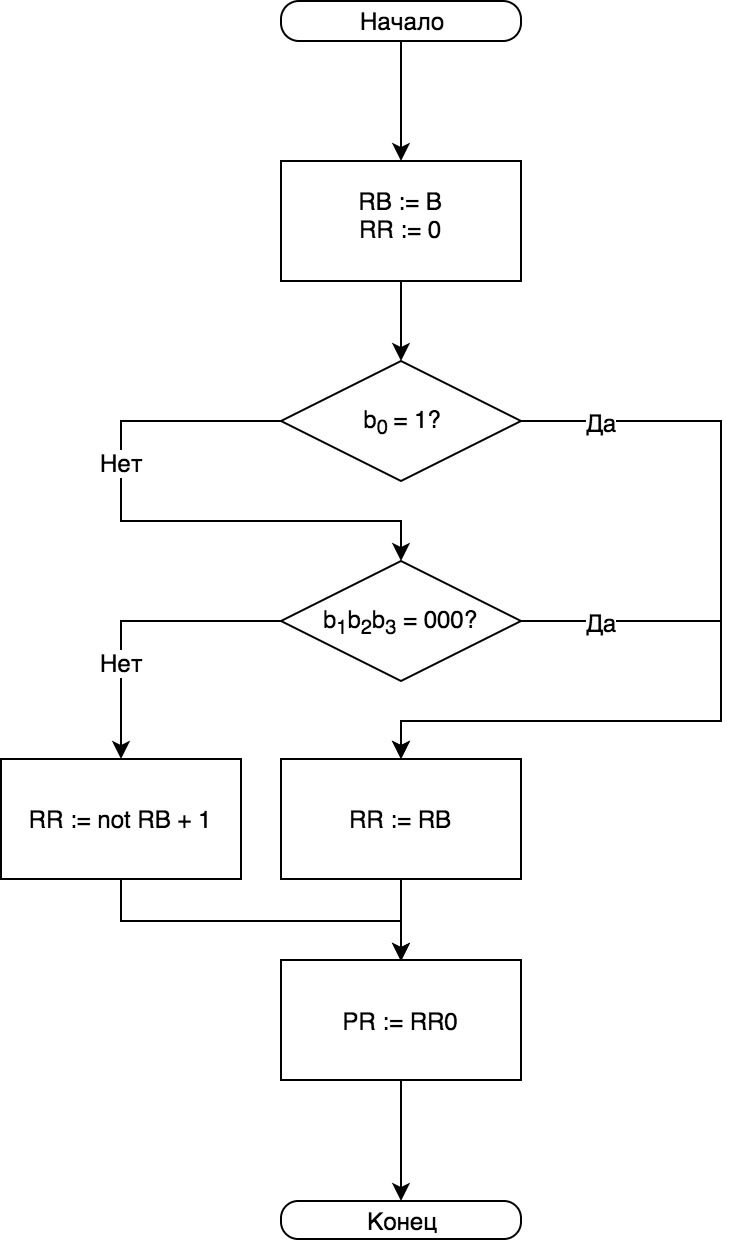
## Алгоритм операции ПЕРЕСЫЛКА ОТРИЦАТЕЛЬНАЯ

Дополнительный код абсолютного значения второго операнда помещается по адресу первого операнда. Устанавливается признак результата: 0 – результат равен нулю, 1 – результат меньше нуля. Поясним алгоритм примерами:

Пример 1

Пример 2

Пример 3



A

B

C

D

E

F

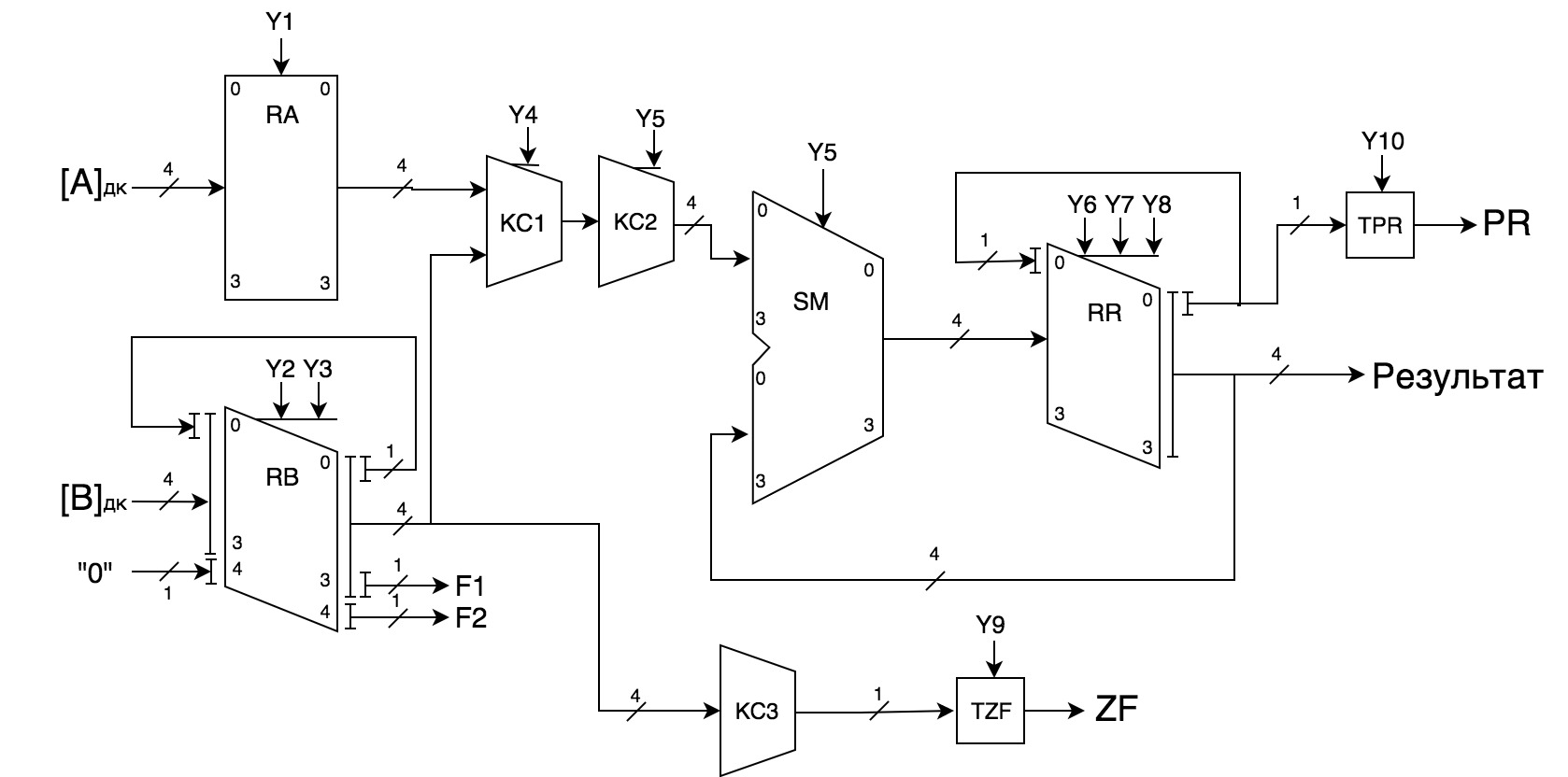
G

H

1 2

*Рис. 1.2 Алгоритм операции ПЕРЕСЫЛКА ОТРИЦАТЕЛЬНАЯ*

* 1. ***Функциональная схема блока операций***

******

*Рис. 1.3. Функциональная схема блока операций*

На рис. 1.3 представлена функциональная схема БО, в следующей таблице 1.1 приведены все элементы данной схемы блока операций с их назначением и функциями.

Таблица 1.1

Функциональное назначение блоков

|  |  |
| --- | --- |
| Обозначение блока | Назначение |
| RA | Регистр первого операнда; 4 разряда  Параллельная загрузка [А] |
| RB | Регистр второго операнда; 4 разряда  Параллельная загрузка [В], арифметический сдвиг вправо на 1 разряд |
| RR | Регистр результата; 4 разрядов;  Установка в 0, параллельная загрузка, арифметический сдвиг вправо на 1 разряд |
| TPR | Триггер признака результата; 1 разряд  Установка в 0, загрузка |
| КС1 | Выбор операнда (RA или RB) |
| КС2 | Инверсия операнда |
| КС3 | Определение положительного числа |
| SM | Сумматор; 4 разряда |

* 1. Проектирование логических элементов блока операций
     1. Регистр первого операнда RA

Данный регистр является четырехразрядным регистром хранения. Наиболее подходящим для реализации функций регистра RA является регистр FD4CE (рис. 1.4).

*Таблица 1.2*

*Таблица управляющих сигналов RA*

|  |  |
| --- | --- |
| Обозначение | Назначение |
| У1 |
| 0 | Хранение |
| 1 | Разрешение загрузки |

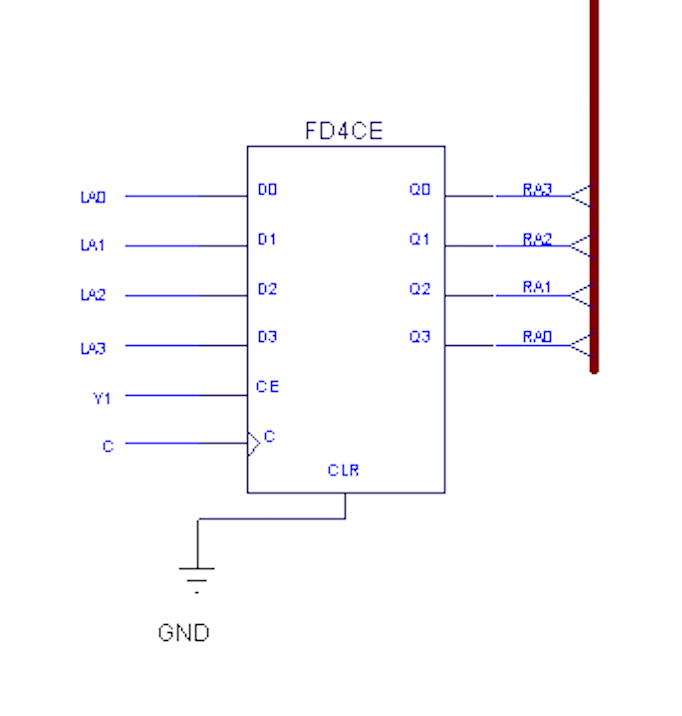
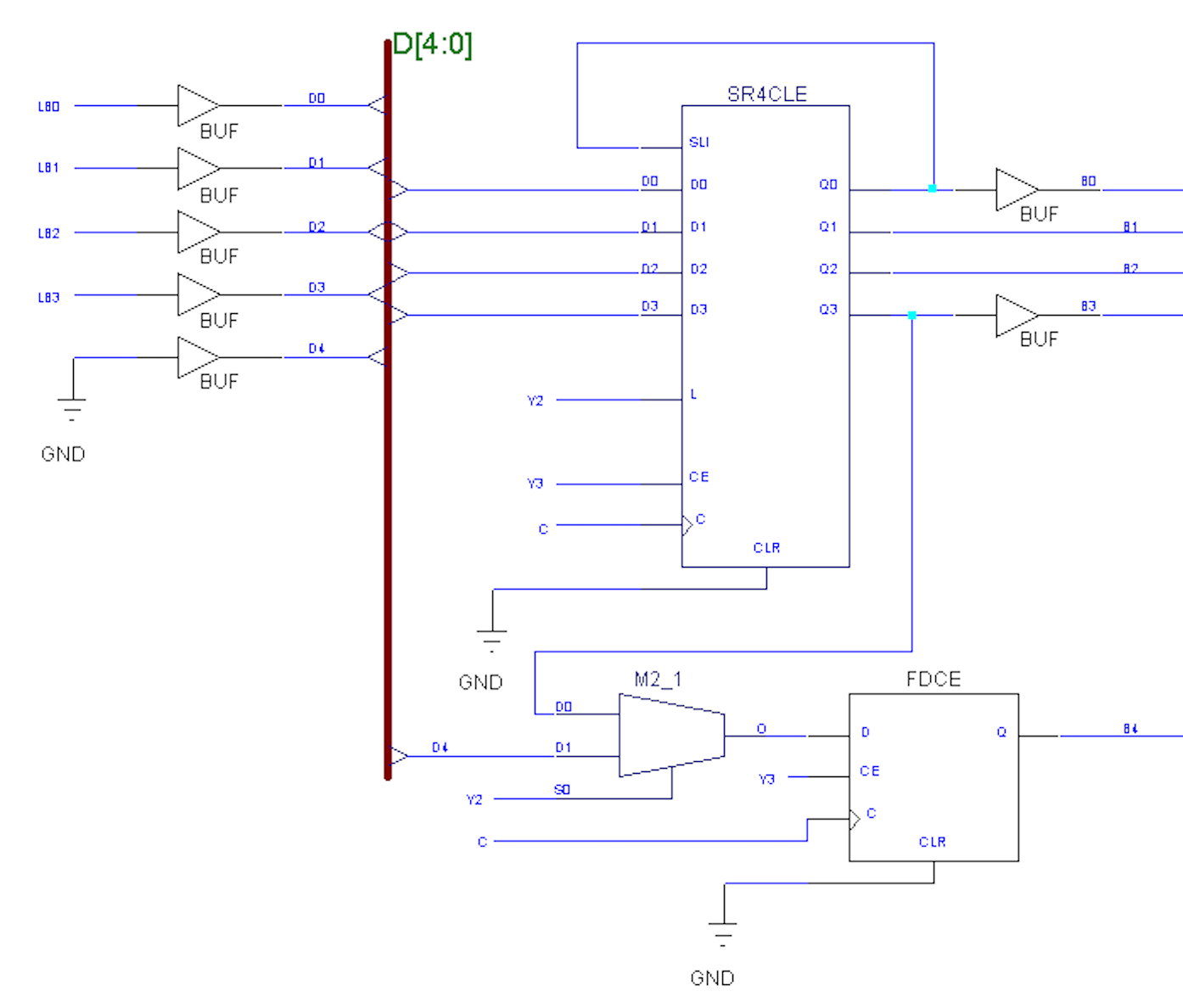


Рис. 1.4. Логическая схема RA

* + 1. Регистр второго операнда RB

Данный четырехразрядный регистр RB, в отличие от регистра RA, должен также осуществлять микрооперацию арифметического сдвига вправо на 1 разряд. Для этой цели был выбран универсальный регистр сдвига SR4CLE (рис. 1.5).

|  |  |  |
| --- | --- | --- |
| Обозначение | | Назначение |
| У2 | У3 |
| 0 | 0 | Хранение |
| 1 | 0 | Разрешение загрузки |
| 0 | 1 | Сдвиг вправо 1 разряд |



*Таблица 1.3*

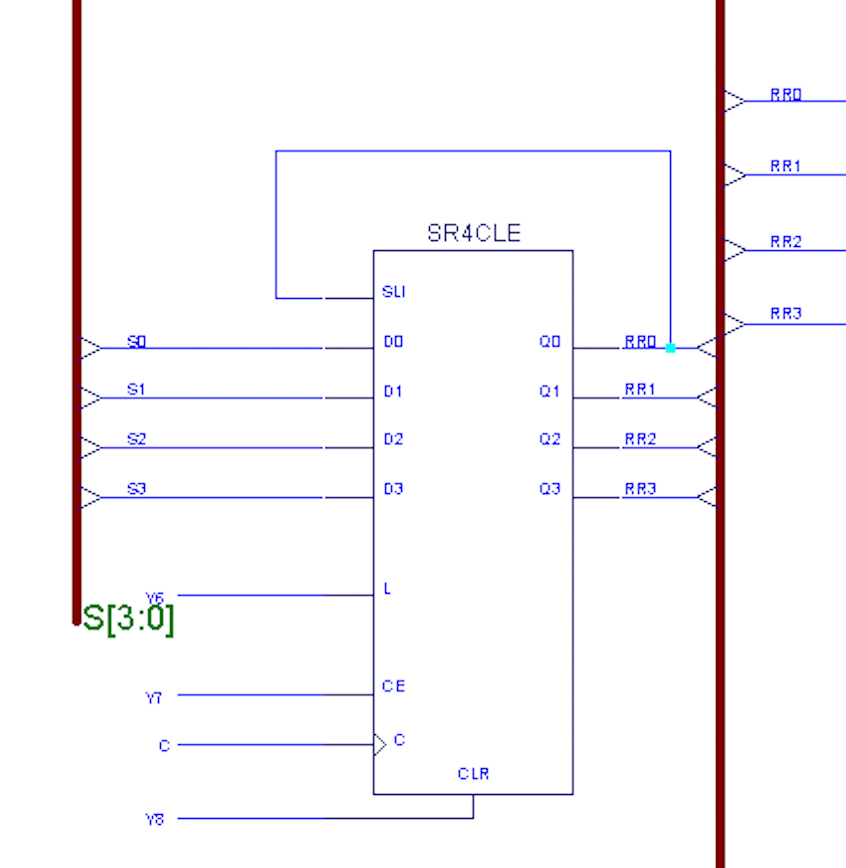
*Таблица управляющих сигналов РВ*

Рис. 1.5. Логическая схема RB

* + 1. Регистр результата RR

Регистр RR должен выполнять микрооперации параллельной загрузки, хранения, арифметического сдвига вправо на 1 разряд и установки в нуль. С учетом данных требований был выбран универсальный регистр сдвига SR8CLE (рис. 1.6). В таблице 1.4 представлены обозначения управляющих сигналов регистра и их назначение.

Причем результат умножения хранится в четырех старших разрядах регистра. Таким образом происходит обрезание результата микрооперации без округления.



*Рис 1.6 Логическая схема RR*

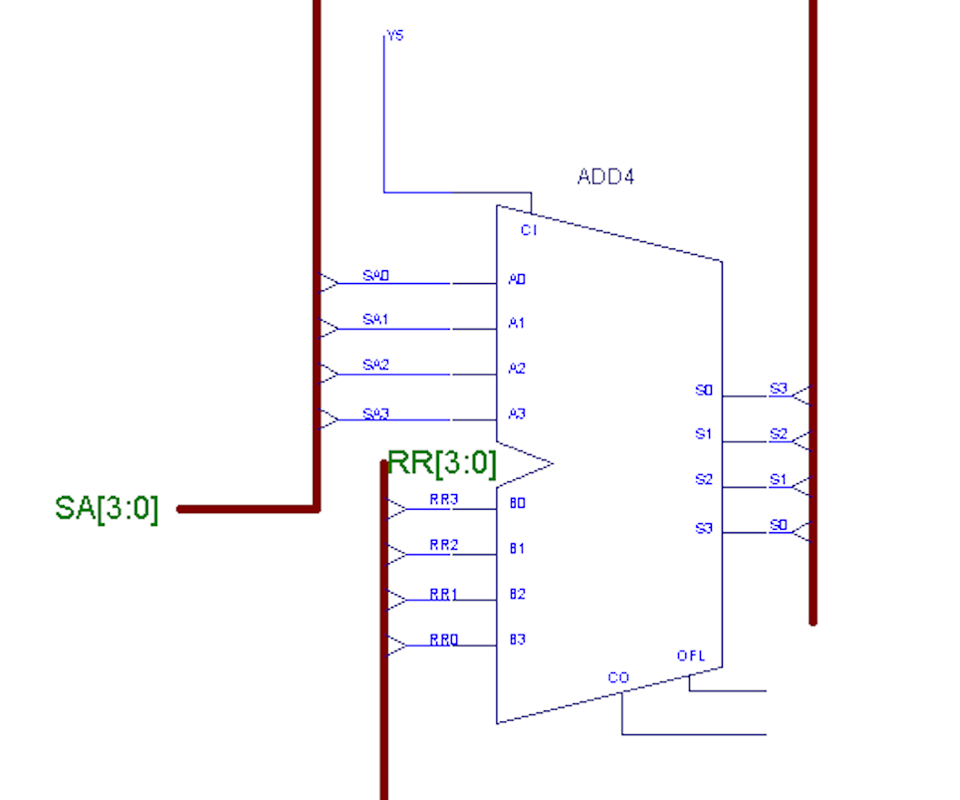
*Таблица 1.4*

*Таблица управляющих сигналов RR*

|  |  |  |  |
| --- | --- | --- | --- |
| Обозначение | | | Назначение |
| У6 | У7 | У8 |
| 0 | 0 | 1 | Асинхронный сброс в 0 |
| 0 | 0 | 0 | Хранение |
| 0 | 1 | 0 | Сдвиг вправо 1 разряд |
| 1 | 0 | 0 | Разрешение загрузки |

## Сумматор СМ

Для построения сумматора возьмем восьмиразрядный сумматор ADD8, подавая сигнал Y5 во время корректирующего шала для получения инверсного значения А или B в дополнительном коде (рис. 1.7).



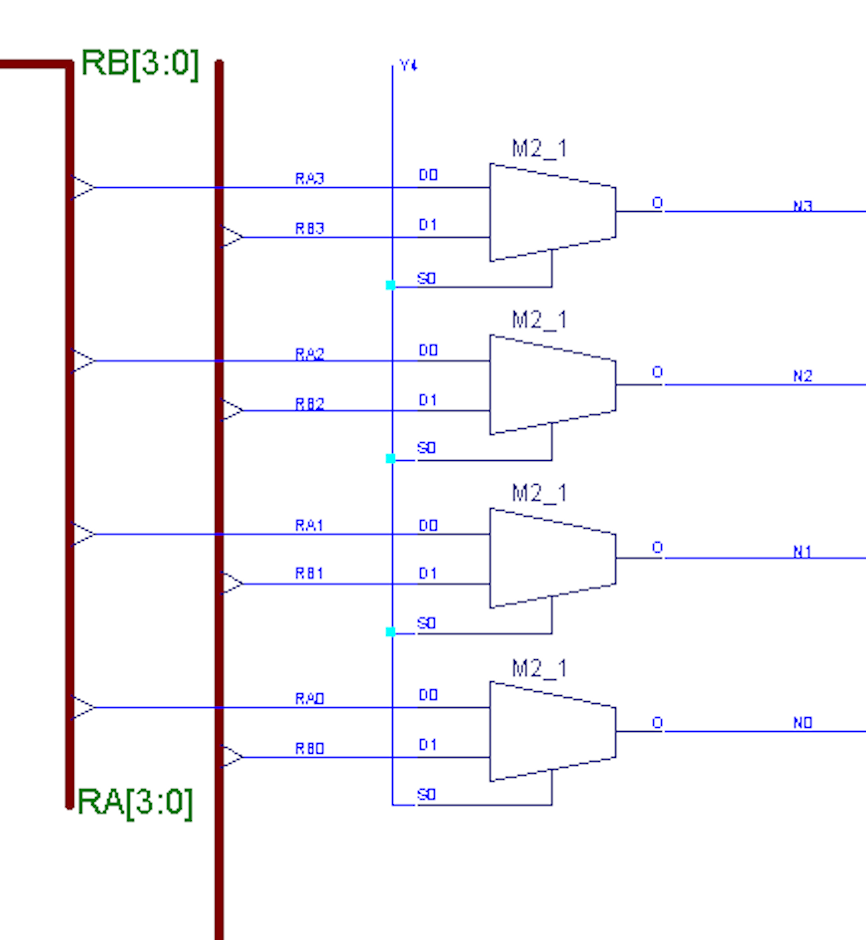
*Рис 1.7 Сумматор с корректирующим входом*

* + 1. Комбинационная схема КС1

С помощью данной комбинационной схемы происходит выбор операнда A или В. За выбор операнда отвечает У4 (рис. 1.8).

В таблице 1.5 представлены назначения управляющего сигнала комбинационной схемы.

|  |  |
| --- | --- |
| **У4** | **N0** |
| 0 | RA0 |
| 1 | RB0 |
| **У4** | **N1** |
| 0 | RA1 |
| 1 | RB1 |
| **У4** | **N2** |
| 0 | RA2 |
| 1 | RB2 |
| **У4** | **N3** |
| 0 | RA3 |
| 1 | RB3 |



*Рис. 1.8 Комбинационная схема КС1*

*Таблица 1.5*

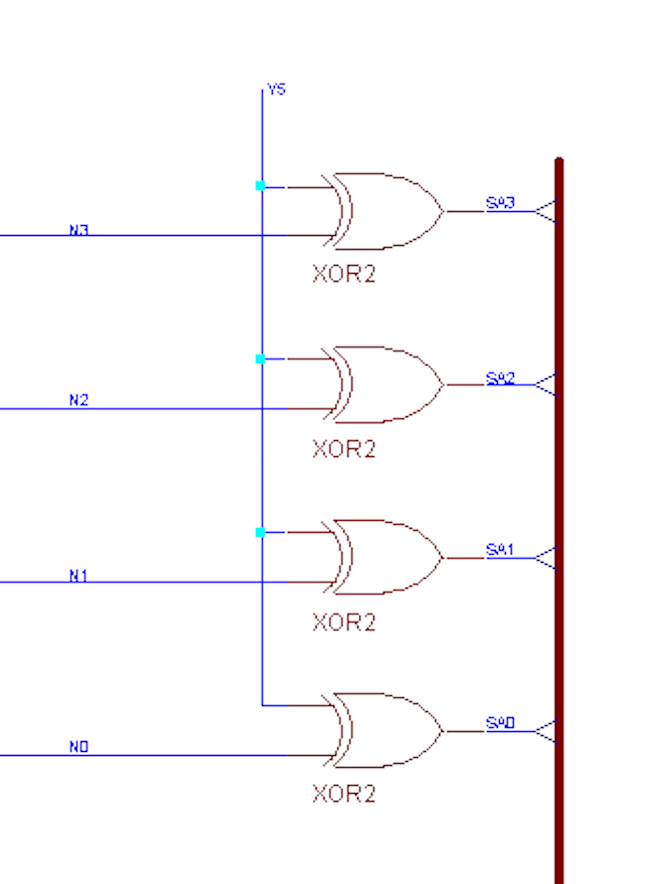
*Таблица управляющих сигналов КС1*

|  |  |
| --- | --- |
| Обозначение | Назначение |
| У4 |
| 0 | Подача на сумматор |
| 1 | Подача на сумматор |

* + 1. Комбинационная схема КС2

Комбинационная схема инвертирует входное значение (рис. 1.9). В качестве управляющего сигнала поступает У5. Таблица назначений представлена ниже (Таблица 1.6)

|  |  |  |
| --- | --- | --- |
| **У5** | **Ni** | **XOR** |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |



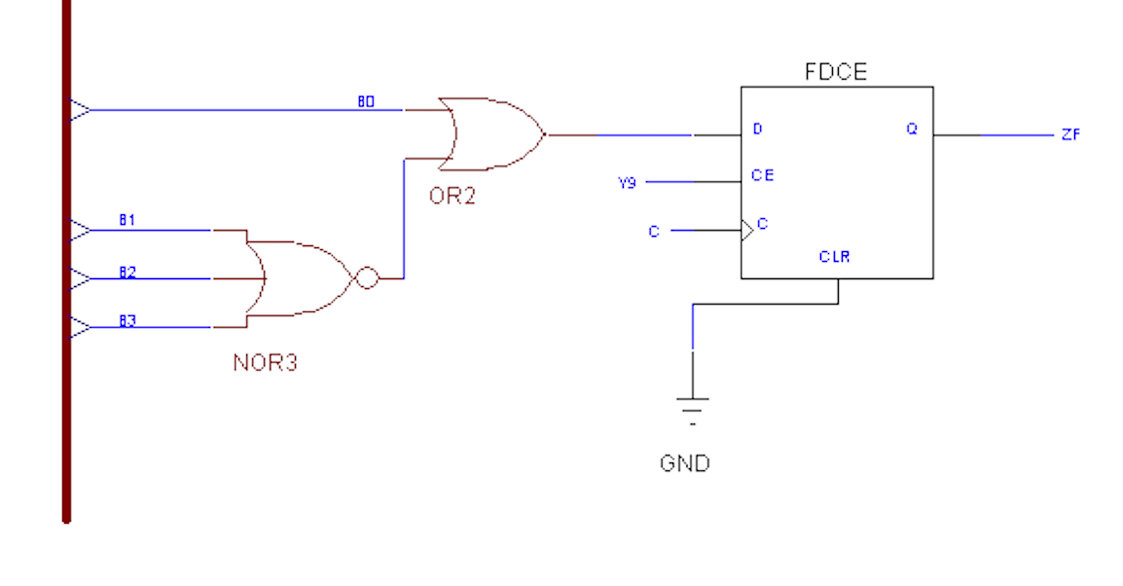
*Рис. 1.9 Комбинационная схема КС2*

*Таблица 1.6*

*Таблица управляющих сигналов КС2*

|  |  |
| --- | --- |
| Обозначение | Назначение |
| У5 |
| 0 | Подача на сумматор |
| 1 | Подача на сумматор |

* + 1. Комбинационная схема КС3 и триггер TZF

Комбинационная схема КС3 осуществляет определение числа на положительность анализируя 4 разряда регистра RB по следующей таблице истинности. (Таблица 1.7) Результат записывается в триггер TZF (рис. 1.10)

*Рис. 1.10 Комбинационная схема КС3 и триггер TZF*

*Таблица 1.7*

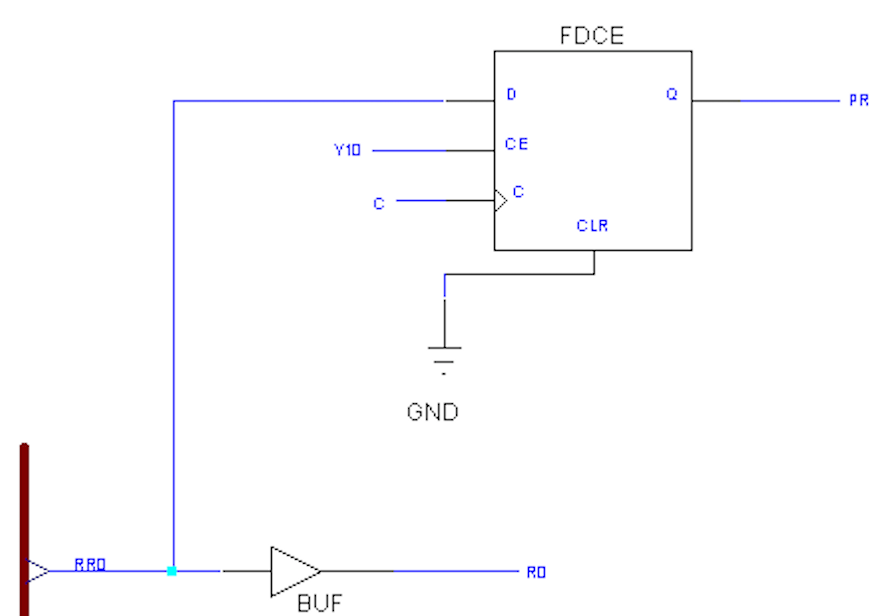
*Таблица истинности для RB*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RB0 | RB1 | RB2 | RB3 | ZF |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

1.3.8. Триггер TPR

Триггер TPR хранит знак результата – признак результата (рис. 1.11). Может принимать значения 0 – результат равен 0, 1 – результат меньше 0.

Управляется сигналом У10. Его назначение представлено в таблице 1.8



*Рис. 1.11 Триггер TPR*

|  |  |
| --- | --- |
| Обозначение | Назначение |
| У9 |
| 0 | Хранение |
| 1 | Разрешение загрузки |

*Таблица 1.8*

*Таблица управляющих сигналов TPR*

* 1. Логическая схема блока операций

На рис. 1.12 приведена полная логическая схема блока операций

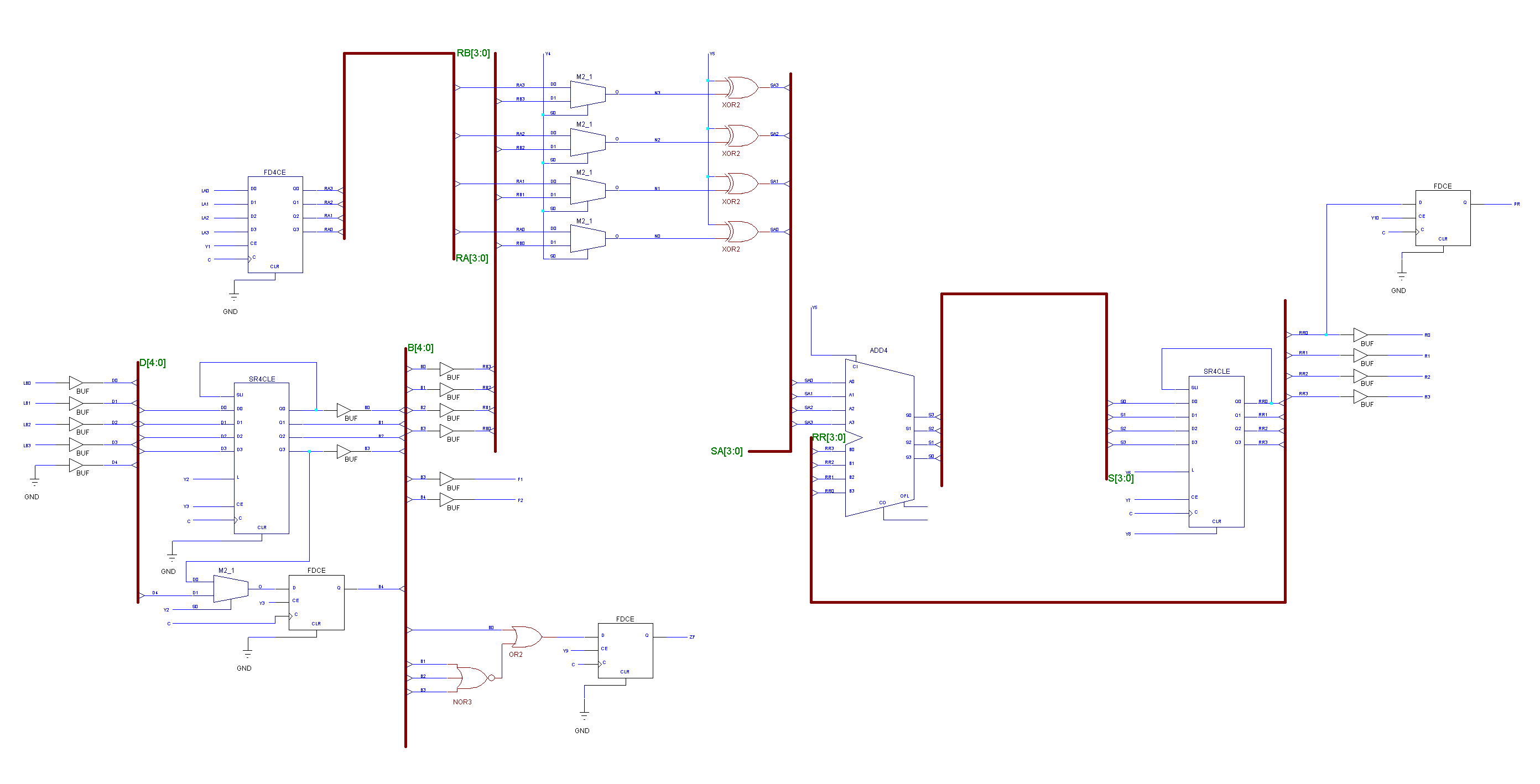


Рис. 1.12. Логическая схема блока операций

В таблицах 1.9 и 1.10 сведены описания управляющих сигналов и сигналов-признаков блока операций.

**Определение управляющих сигналов БО**

*Таблица 1.9*

*Таблица управляющих сигналов*

|  |  |
| --- | --- |
| Сигнал | Операция |
| У1 | Загрузка А в RA |
| У2 | Загрузка B в RB |
| У3 | Арифметический сдвиг RB вправо на 1 разряд |
| У4 | Прием значения из RA или из RB |
| У5 | Подача инверсного входа на сумматор |
| У6 | Загрузка RR |
| У7 | Арифметический сдвиг RR вправо на 1 разряд |
| У8 | Сброс RR |
| У9 | Загрузка TZF |
| У10 | Загрузка TPR |

**Определение сигналов-признаков БО**

*Таблица 1.10*

*Таблица сигналов признаков*

|  |  |
| --- | --- |
| Сигнал | Признак |
| F1 | 3 разряд В |
| F2 | 4 разряд В |
| F3 | Признак положительного числа |

* 1. Микропрограммы выполнения операций
     1. Алгоритмы выполнения микропрограмм

На рис. 1.13 и 1.14 представлены блок-схемы алгоритмов микропрограмм выполнения операций Умножение и Пересылка отрицательная.

1 2

2

A

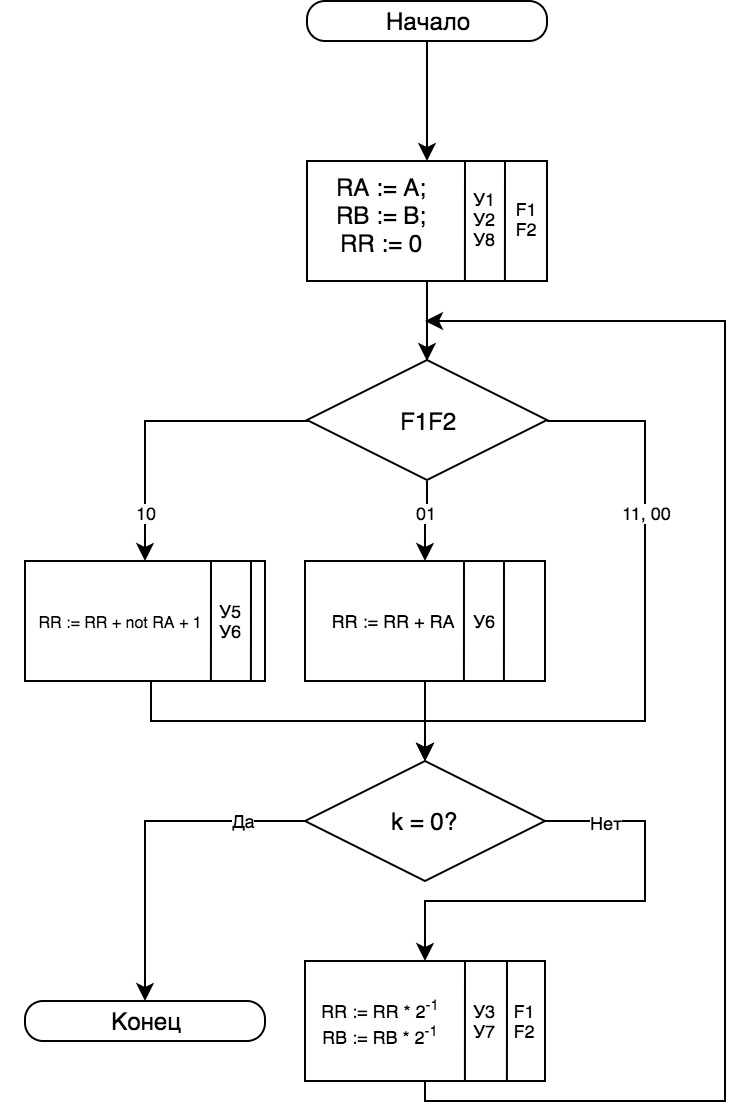
B

C

D

E

F



*Рис. 1.13 Микропрограмма операции УМНОЖЕНИЕ*

1 2

2

A

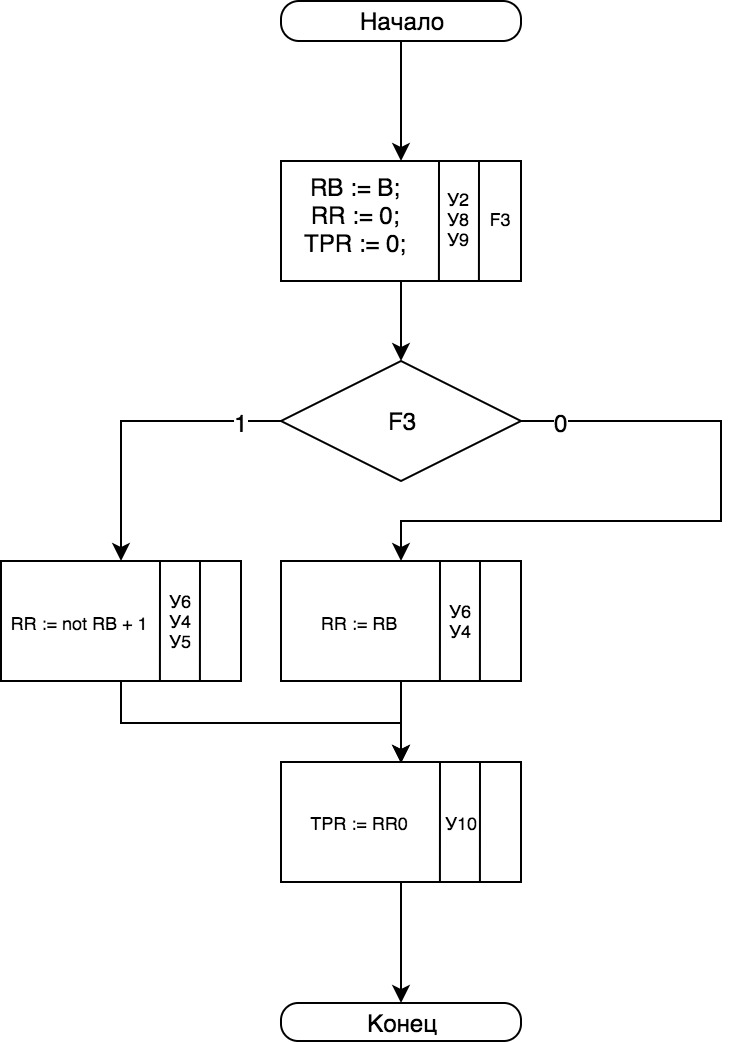
B

C

D

E

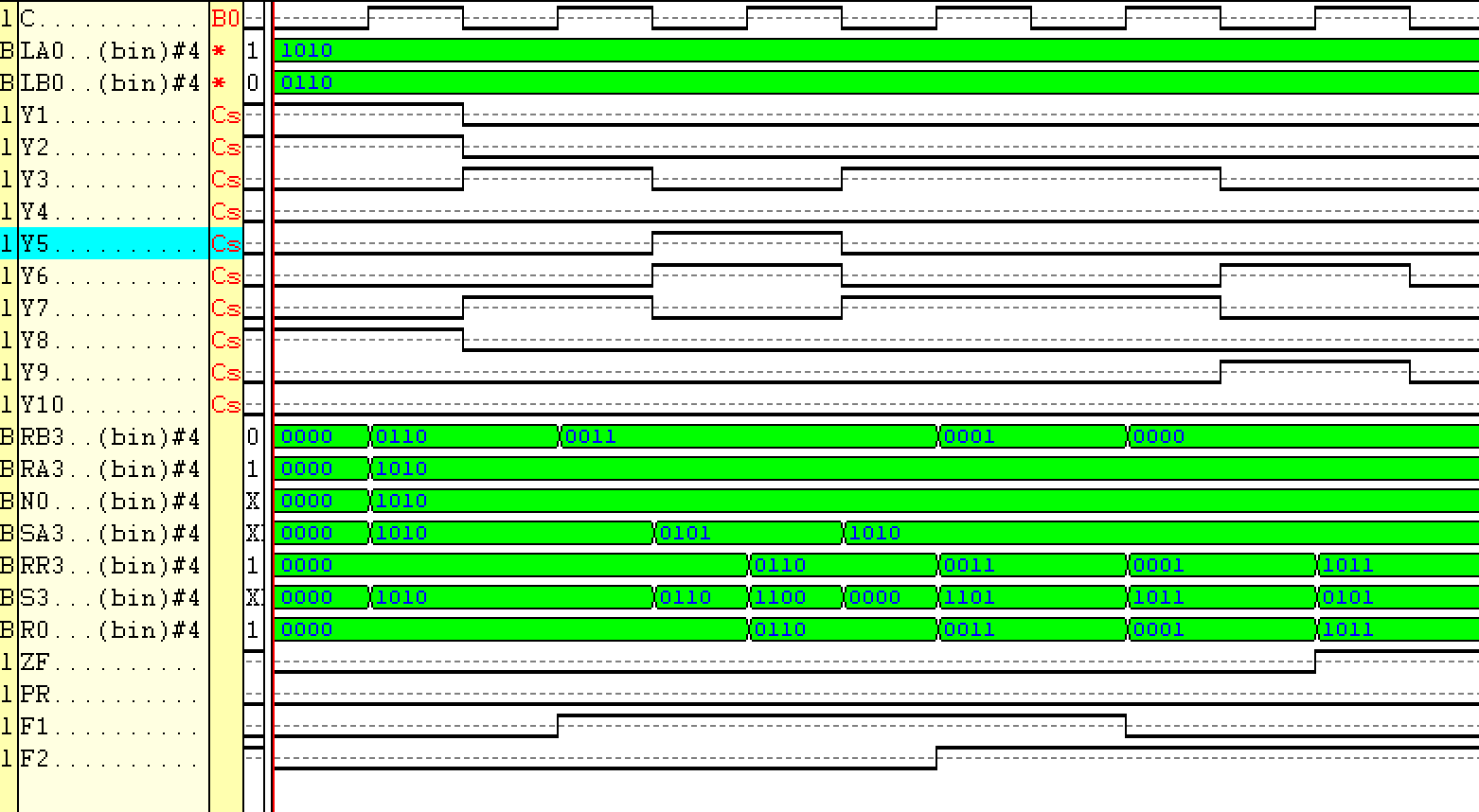
F



*Рис. 1.14 Микропрограмма операции ПЕРЕСЫЛКА ОТРИЦАТЕЛЬНАЯ*

* + 1. Примеры выполнения микропрограмм

На рис. 1.15-1.17 приведены результаты моделирования схемы блока операций при выполнении операции умножения и пересылки отрицательной для примеров, приведенных в пункте 1.1.1.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | 0 | 0 | 0 | 0 | RR = 0 |
| k = 3 |  | пропуск | | | |  |
|  | + | 0 | 0 | 0 | 0 | RR = RR \* |
| k= 2 | 0 | 1 | 1 | 0 | RA = RA \* |
|  |  | 0 | 1 | 1 | 0 | RR = RR + RA |
|  |  | 0 | 0 | 1 | 1 | RR = RR \* |
| k= 1 |  | пропуск | | | |  |
|  | + | 0 | 0 | 0 | 1 | RR = RR \* |
| k = 0 | 1 | 0 | 1 | 0 | RA = RA \* |
|  |  | 1 | 0 | 1 | 1 | RR = RR + RA |

