МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

**«Вятский государственный университет»**

**(ФГБОУ ВПО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Допущено к защите

Руководитель проекта

\_\_\_\_\_\_\_\_\_\_\_\_ (Мельцов В.Ю.)

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_2019г.

«СИНТЕЗ МИКРОПРОГРАММНОГО УПАРВЛЯЮЩЕГО АВТОМАТА»

Пояснительная записка курсового проекта по дисциплине

«Теория автоматов»

ТПЖА.09.03.01.014 ПЗ

Разработал студент группы ИВТ-21 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Седов М.Д./

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Мельцов В.Ю./

Работа защищена с оценкой «\_\_\_\_\_\_\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(оценка) (дата)*

Члены комиссии \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Мельцов В.Ю./

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Куваев А.С./

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Исупов К.С./

(подпись)

Киров 2019

**Реферат**

Седов М.Д. Синтез микропрограммного управляющего автомата: ТПЖА.090301.014 ПЗ: Курс. проект / ВятГУ, каф. ЭВМ; рук. Мельцов В.Ю. - Киров, 2019. Графическая часть 4 л. – ф. А2. ПЗ 64 с, 15 рис., 20 табл., 6 источника, 6 приложений.

СИНТЕЗ АВТОМАТА, ОПЕРАЦИОННЫЙ АВТОМАТ, ГРАФ-СХЕМА АЛГОРИТМА, УМНОЖЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ, ДОПОЛНИТЕЛЬНЫЙ КОД, МИКРОПРОГРАММНЫЙ УПРАВЛЯЮЩИЙ АВТОМАТ

Объект исследования и разработки – микропрограммный автомат, выполняющий умножение двоичных чисел.

Цель курсового проекта – синтез микропрограммного автомата, управляющего выполнением умножение двоичных чисел в дополнительном коде с плавающей запятой, с характеристиками первым способом с простой коррекцией.

Результат работы – функциональная схема микропрограммного управляющего автомата.

# Содержание

1. [Введение 5](#_Toc2703802)

2. [Постановка задачи 6](#_Toc2703803)

3. [Описание используемого алгоритма умножения 7](#_Toc2703804)

4. [Численные примеры 9](#_Toc2703805)

5. [Разработка функциональной схемы операционной части устройства 16](#_Toc2703806)

5.1 [Состав операционного автомата 16](#_Toc2703807)

5.2 [Описание работы операционного автомата 18](#_Toc2703808)

5.3 [Управляющие и осведомительные сигналы 19](#_Toc2703809)

6. [Разработка содержательной граф-схемы алгоритма 21](#_Toc2703810)

7. [Разработка отмеченной граф-схемы алгоритма 23](#_Toc2703811)

8. [Построение графов автоматов моделей Мили и Мура и выбор структурной схемы управляющего автомата 27](#_Toc2703812)

9. [Кодирование внутренних состояний для модели Мили 30](#_Toc2703813)

9.1 [Кодирование внутренних состояний для модели Мили на D–триггерах 30](#_Toc2703814)

9.2 [Кодирование внутренних состояний для модели Мили на RS – триггерах 34](#_Toc2703815)

9.3 [Кодирование внутренних состояний для модели Мили на счетчике 42](#_Toc2703816)

9.4 [Кодирование внутренних состояний для модели Мили на сдвиговом регистре 47](#_Toc2703817)

10. [Кодирование внутренних состояний для модели Мура 51](#_Toc2703818)

10.1 [Кодирование внутренних состояний для модели Мура на D – триггерах 51](#_Toc2703819)

11. [Построение схемы управляющего микропрограммного автомата 57](#_Toc2703820)

12. [Заключение 58](#_Toc2703821)

13. [Приложение А 59](#_Toc2703822)

14. [Приложение Б 59](#_Toc2703823)

15. [Приложение В 61](#_Toc2703824)

16. [Приложение Г 61](#_Toc2703825)

17. [Приложение Д 63](#_Toc2703826)

18. [Приложение Е 64](#_Toc2703827)

19. [Библиографический список 65](#_Toc2703828)

# Введение

В последние годы из-за значительного роста объема вычислений и невозможности их обработки вручную, с большой интенсивностью ведутся работы по созданию и применению различных автоматических систем для переработки информации. Такие автоматы реализуются в виде самостоятельных устройств специального назначения или в виде блоков, входящих в другие системы. При этом работа ведется с математическими моделями, предназначенными для приближенного отображения физических моделей.

Основной целью данного курсового проекта является получение навыков синтеза управляющего автомата (УА) с жесткой логикой на основе разработки машинных алгоритмов одной из заданных арифметических операций. Основным требованием является минимизация аппаратурных затрат как управляющего, так и операционного автомата при приемлемом быстродействии.

# Постановка задачи

Синтезировать микропрограммный автомат, управляющий выполнением умножения двоичных чисел с плавающей запятой, с характеристиками в дополнительном коде первым способом с простой коррекцией.

# Описание используемого алгоритма умножения

1. Проверить множимое на равенство нулю: если ноль, то сформировать результат, равный нулю, перейти к пункту 10. Иначе перейти к пункту 2.
2. Проверить множитель на равенство нулю: если ноль, то сформировать результат, равный нулю, перейти к пункту 10. Иначе перейти к пункту 3.
3. Сложить характеристики двух сомножителей.
4. Проверить на наличие ПРС. Если возникло ПРС (признаком ПРС является получение единицы переноса и единицы в старшем разряде результирующей характеристики), то необходимо зафиксировать её появление и прекратить операцию. Временное ПРС может возникнуть, когда старший разряд равен единице, образовалась единица переноса, но все разряды характеристики, за исключением старшего, равны нулю. При этом нужно продолжить алгоритм умножения. Если возникла ситуация ПМР (признаком ПМР является отсутствие единицы переноса и ноль в старшем разряде результирующей характеристики), то необходимо зафиксировать ее появление и сформировать нулевой результат, перейти к пункту 10. В противном случае переходим к пункту 5.
5. Выполнить цикл умножения мантисс со знаком первым способом:

5.1) Произвести анализ младшего разряда множителя.

5.2) Если разряд равен 1, произвести суммирование половины множимого с суммой частичных произведений. Иначе перейти к пункту 6.

1. Произвести сдвиг множителя вправо, а сумму частичных произведений сдвинуть вправо.
2. Если все разряды множителя были рассмотрены, перейти к пункту 8, иначе перейти к пункту 5.
3. Если множитель был отрицательным, выполнить коррекцию половиной множимым.
4. Проверить на нормализацию полученную мантиссу. Если мантисса ненормализованная – сдвинуть мантиссу на 1 разряд влево, вычесть из характеристики 1. Если ранее было зафиксировано временное ПРС, оно устраняется. Если после нормализации мантиссы произошло ПМР, нужно зафиксировать её появление и сформировать результат, равный нулю. Перейти к пункту 10. Если результат нормализован, было зафиксировано временное ПРС, то установить признак ПРС, перейти к пункту 11. Иначе перейти к пункту 10.
5. Выдать результат умножения.
6. Останов операции.

# Численные примеры

* Операция умножения без ПРС и ПМР.

A=-40; B=-23

Представим числа в разрядной сетке

|  |  |  |  |
| --- | --- | --- | --- |
|  | Знак числа | Мантисса | Характеристика |
| A | 1 | 011000 | 100110 |
| B | 1 | 010010 | 100101 |

Сложим характеристики двух операндов

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

ПРС и ПМР не возникли. Множитель и множимое не равны нулю. Выполним произведение мантисс сомножителей.

Таблица 1 -Умножение чисел без ПРС и ПМР

|  |  |  |  |
| --- | --- | --- | --- |
| Множитель | Множимое | СЧП | Пояснение |
| 1.011000 | 1.010010 | 0.0000000 | Исх. данные |
| 1.01100*0* | 1.010010 | 0.0000000 | Сдвиги |
| 1.***1***0110*0* | 1.010010 | 0.0000000 | Сдвиги |
| 1.***11***011*0* | 1.010010 | 0.0000000 | Сдвиги |
| 1.***111***01*1* | 1.010010 | 0.0000000  1.1010010  1.1010010 | +0.5М |
| 1.***111***01*1* | 1.010010 | 1.1010010 | Сдвиги |
| 1.***1111***0*1* | 1.010010 | 1.1101001  1.1010010  1.0111011 | +0.5М |
| 1.***1111***0*1* | 1.010010 | 1.0111011 | Сдвиги |
| 1.***11111****0* | 1.010010 | 1.1011101 | Сдвиги |
| 1.***111111*** | 1.010010 | 1.1101110  0.0101110  0.0011100 | Коррекция множимым |
| 1.***111111*** | 1.010010 | 0.0011100 | Результат |

Результат умножения после нормализации:

|  |  |  |
| --- | --- | --- |
| 0 | 111000 | 101010 |

Полученный результат: 0.11100000002 = 89610

* Истинное ПРС при сложении характеристик.

A=0,5 \* 227; B=0,5\*220

Представим числа в разрядной сетке

|  |  |  |  |
| --- | --- | --- | --- |
|  | Знак числа | Мантисса | Характеристика |
| A | 0 | 100000 | 111011 |
| B | 0 | 100000 | 110100 |

Сложим характеристики двух операндов

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

В результирующей характеристике было получено ПРС. В старшем разряде была получена 1, также возникла единица переноса, остальные разряды характеристики не равны нулю. Значит формируется сигнал о ПРС, происходит останов операции умножения.

* Временное ПРС при сложении характеристик, которое превращается в истинное ПРС после цикла умножения.

A=0,875\*216; B=0,75\*216

Представим числа в разрядной сетке

|  |  |  |  |
| --- | --- | --- | --- |
|  | Знак числа | Мантисса | Характеристика |
| A | 0 | 111000 | 110000 |
| B | 0 | 110000 | 110000 |

Сложим характеристики двух операндов

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

В результирующей характеристике было получено временное ПРС. В старшем разряде была получена 1, также возникла единица переноса, остальные разряды характеристики равны нулю. Выполним цикл умножения мантисс сомножителей.

Таблица 2 -Умножение чисел с возникновением неустранимого временного ПРС

|  |  |  |  |
| --- | --- | --- | --- |
| Множитель | Множимое | СЧП | Пояснение |
| 0.111000 | 0.110000 | 0.0000000 | Исх. данные |
| 0.11100*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.01110*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.00111*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.00011*1* | 0.110000 | 0.0000000  0.0110000  0.0110000 | +0.5M |
| 0.00011*1* | 0.110000 | 0.0110000 | Сдвиги |
| 0.00001*1* | 0.110000 | 0.0011000  0.0110000  0.1001000 | +0.5М |
| 0.00001*1* | 0.110000 | 0.1001000 | Сдвиги |
| 0.00000*1* | 0.110000 | 0.0100100  0.0110000  0.1010100 | +0.5М |
| 0.00000*0* | 0.110000 | 0.1010100 | Сдвиги |
| 0.000000 | 0.110000 | 0.0101010 | Результат |

Результат умножения нормализован. Временное ПРС становится истинным ПРС. Формируется сигнал о ПРС.

* Временное ПРС, которое устраняется при нормализации.

A=0,5\*216; B=0,75\*216

Представим числа в разрядной сетке

|  |  |  |  |
| --- | --- | --- | --- |
|  | Знак числа | Мантисса | Характеристика |
| A | 0 | 100000 | 110000 |
| B | 0 | 100000 | 110000 |

Сложим характеристики двух операндов

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

В результирующей характеристике было получено временное ПРС. В старшем разряде была получена 1, также возникла единица переноса, остальные разряды характеристики равны нулю. Выполним цикл умножения мантисс сомножителей.

Таблица 3 – Умножение чисел с устранением временного ПРС при нормализации

|  |  |  |  |
| --- | --- | --- | --- |
| Множитель | Множимое | СЧП | Пояснение |
| 0.100000 | 0.110000 | 0.0000000 | Исх. данные |
| 0.10000*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.01000*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.00100*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.00010*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.00001*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.00000*1* | 0.110000 | 0.0000000  0.0110000  0.0110000 | +0.5М |
| 0.00000*1* | 0.110000 | 0.0110000 | Сдвиги |
| 0.000000 | 0.110000 | 0.0011000 | Результат |

Мантисса не нормализована, сдвинем мантиссу влево на один разряд, из результирующей характеристики вычтем один. Временное ПРС устраняется при нормализации.

|  |  |  |
| --- | --- | --- |
| 0 | 110000 | 111111 |

Полученный результат: 0.75 \* 23110

Искомый результат: 0.75 \* 23110.

* ПМР при сложении характеристик.

A=0,5\*2-18; B=0,5\*2-17

Представим числа в разрядной сетке

|  |  |  |  |
| --- | --- | --- | --- |
|  | Знак числа | Мантисса | Характеристика |
| A | 0 | 100000 | 001110 |
| B | 0 | 100000 | 001111 |

Сложим характеристики двух операндов

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

В результирующей характеристике было получено ПМР. В старшем разряде была получена 0, единица переноса не возникла. Формируется сигнал о ПМР, выдается результат, равный нулю.

|  |  |  |
| --- | --- | --- |
| 0 | 000000 | 100000 |

* Возникновение ПМР после нормализации результата умножения

A=0,5\*2-16; B=0,75\*2-16

Представим числа в разрядной сетке

|  |  |  |  |
| --- | --- | --- | --- |
|  | Знак числа | Мантисса | Характеристика |
| A | 0 | 100000 | 010000 |
| B | 0 | 110000 | 010000 |

Сложим характеристики двух операндов

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

ПРС и ПМР не возникли. Множитель и множимое не равны нулю. Выполним произведение мантисс сомножителей.

Таблица 4 – Умножение чисел с возникновением ПМР после нормализации

|  |  |  |  |
| --- | --- | --- | --- |
| Множитель | Множимое | СЧП | Пояснение |
| 0.100000 | 0.110000 | 0.0000000 | Исх. данные |
| 0.10000*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.01000*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.00100*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.00010*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.00001*0* | 0.110000 | 0.0000000 | Сдвиги |
| 0.00000*1* | 0.110000 | 0.0000000  0.0110000  0.0110000 | +0.5М |
| 0.00000*1* | 0.110000 | 0.0110000 | Сдвиги |
| 0.000000 | 0.110000 | 0.0011000 | Результат |

Мантисса не нормализована, сдвинем мантиссу влево на один разряд, из результирующей характеристики вычтем один. В характеристиках возникает ПМР.

Уменьшение характеристики на 1

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

Формируется сигнал о ПМР, выдается результат, равный нулю.

|  |  |  |
| --- | --- | --- |
| 0 | 000000 | 100000 |

# Разработка функциональной схемы операционной части устройства

## Описание работы операционного автомата

Операнды разрядностью 32 бита поступают в операционный автомат (ОА) в ДК по входной шине. Первый операнд - множимое. Запись мантиссы множимого (со знаком) производится в RG1 и RG2. В RG4 происходит запись характеристики (со знаком) множимого. Выполняется проверка операнда на ноль. Если операнд равен нулю, то на выходную шину подаем ноль. В счетчик СТ1 записывается значение регистра RG4. В счетчик циклов СТ2 записывается значение «001000». Вторым операндом приходит множитель. Запись мантиссы множителя (со знаком) осуществляется в RG1, запись его характеристики в RG4. Выполняется проверка операнда на ноль, если операнд равен «0», то на выходную шину подается «0». В СТ1 записывается значение суммы характеристик. Если возникло ПРС характеристик, триггер Т2 устанавливаем в единицу. При этом может возникнуть временная ПРС, которая впоследствии может быть исправлена при нормализации мантиссы. Если же ПРС не возникло, то в RG3 заносим результат на выходе сумматора SM1. После этого производится сдвиг RG1 вправо и RG3 вправо, а значение СТ2 увеличивается на 1. Цикл умножения заканчивается, когда в старшем разряде СТ2 появится «1». Если два старших разряда RG3 равны 0 или 1, производится нормализация. В противном случае необходимо проверить: а было ли зафиксировано временное ПРС. Если да, то устанавливаем триггер Т2 в единицу. Если после нормализации возникло ПМР, то обнуляем RG3 и счетчик CT1 и выдаем на выходную шину ноль.

Для выдачи результата на выходную шину содержимое RG3 без старшего разряда, СТ1 подается на усилитель формирователь. Причем старший разряд СТ1 инвертируется.

## Управляющие и осведомительные сигналы

Для организации работы операционной части из управляющей части автомата (УА) подаются следующие управляющие сигналы:

* y0 - запись в RG1, RG4;
* y1 - запись в RG2, установка T1, Т2 в положение «0», обнуление RG3 и CT1, запись СТ2;
* y2 - запись в СТ1 значения выхода SM2, запись в T1 значения выхода переноса SM2;
* y3 - сдвиг RG3 влево, СТ1 := СТ1-1;
* y4 - запись в RG3 значения выхода SM1
* y5 - сдвиг RG1 вправо и RG3 вправо, СТ2 := СТ2+1;
* y6 - установка Т2 в положение «1»;
* y7 - инвертирование значения регистра RG2
* y8 - выдача результата на выходную шину
* y9 - сдвиг RG2 вправо.

Из ОА в УА необходимо передавать осведомительные сигналы о состоянии ОА, которые определяются следующим списком логических условий:

* Х - проверка наличия операндов на входной шине
* p1 - проверка на ноль;
* p2 - проверка на ПРС;
* p3 - проверка на временное ПРС
* p4 - старшие разряды RG3 (проверка нормализации результата);
* p5 - проверка на ПМР;
* p6 - проверка на окончание операции умножения;
* p7 - проверка очередного разряда множителя
* Z - проверка возможности выдачи результата на выходную шину

Таким образом, УА должен вырабатывать 10 управляющих сигналов и посылать их в ОА в нужные такты машинного времени в соответствии с алгоритмом выполнения операции умножения, учитывая 9 осведомительных сигналов, поступающих из ОА. Функциональная схема (ФС) ОА изображена на рисунке 1 и представлена в приложении А.

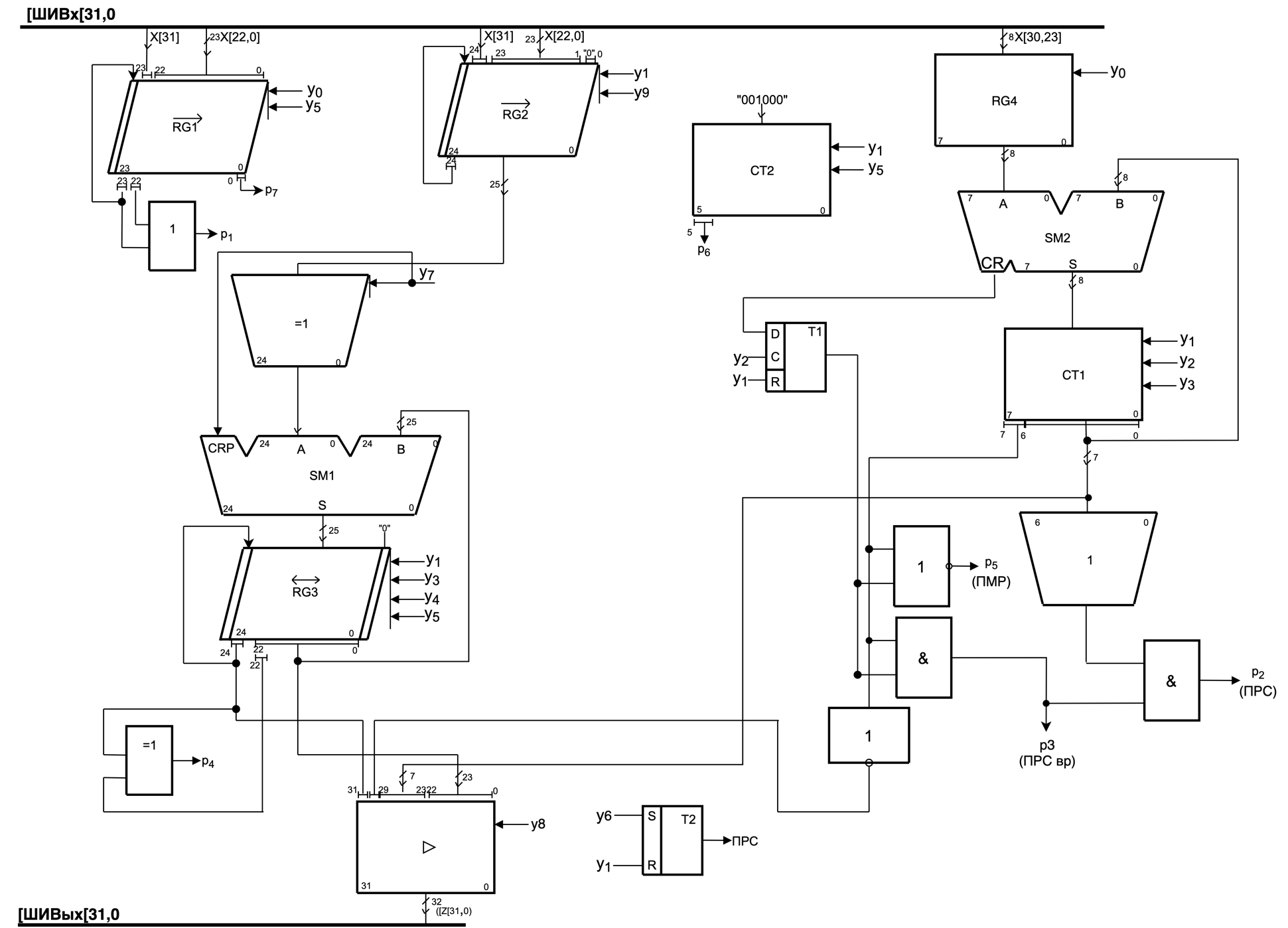


Рисунок 1 - Функциональная схема операционного автомата

# Разработка содержательной граф-схемы алгоритма

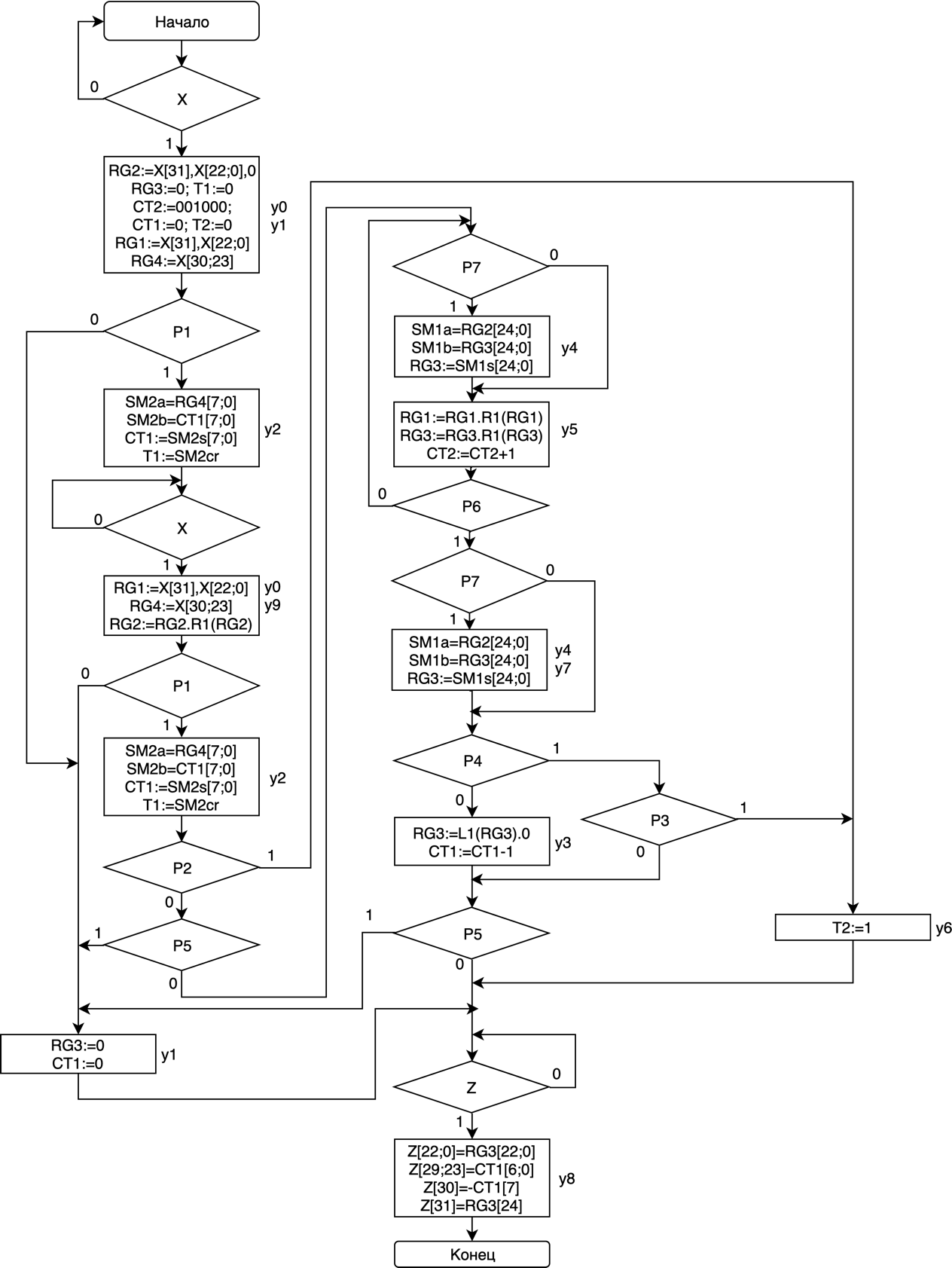


Рисунок 2 – ГСА алгоритма умножения

# Разработка отмеченной граф-схемы алгоритма

Для разметки граф-схемы алгоритма каждой совокупности микроопераций, находящихся в операторных вершинах, ставятся в соответствие управляющие микрокоманды (МК) Y1…Yn. Эти МК являются выходными сигналами УА и обеспечивают выполнение требуемых действий в соответствии со списком микроопераций (МО) ОА. Совокупность МО для каждой операторной вершины образуют микрокоманды, список которых представлен в таблице 5. Каждой условной вершине содержательной ГСА ставится в соответствие один из входных сигналов управляющего автомата X1…Xm.

Таблица 5 – Список микрокоманд

|  |  |
| --- | --- |
| МК | Совокупность МО |
| Y0 | y0, y1 |
| Y1 | y2 |
| Y2 | y0 |
| Y3 | y4 |
| Y4 | y5 |
| Y5 | y3 |
| Y6 | y1 |
| Y7 | y6 |
| Y8 | y7 |
| Y9 | y0, y9 |
| Y10 | y4, y8 |

Далее в полном соответствии с правилами разметки содержательной ГСА (см. ниже) строится отмеченная ГСА.

Предварительно в каждой условной вершине проставляются символы из множества входных сигналов УА – Х1, Х2, …, ХМ (таблица 6). Во всех операторных вершинах ГСА проставляют символы из множества выходных сигналов УА – У1, У2, …, УN (таблица 5). Удобно в каждой операторной вершине ГСА вслед за символом МК указать в скобках набор МО, образующих каждую МК.

Таблица 6 – Список входных сигналов для УА

|  |  |
| --- | --- |
| Входной сигнал УА | Логическое условие ОА (осведомительные сигналы) |
| X1 | Х |
| X2 | P1 |
| X3 | P2 |
| X4 | P5 |
| X5 | P6 |
| X6 | P7 |
| X7 | P4 |
| X8 | P3 |
| X9 | Z |

Разметка ГСА в соответствии с моделью Мили, выполняется по следующим правилам:

1. Вход вершины, следующей за начальной, и вход конечной вершины отмечаются символом начального состояния автомата а0.

2. Входы всех вершин, следующих за операторными, отмечаются символами а1, …, аК.

3. Если вход вершины отмечается, то только одним символом.

4. Входы различных вершин за исключением конечной отмечаются различными символами.

Разметка ГСА в соответствии с моделью Мура, выполняется по следующим правилам:

1. Символом начального состояния автомата а0 отмечаются начальная и конечная вершины.

2. Различные операторные вершины отмечаются различными символами а1, …, аК.

Все операторные вершины должны быть отмечены, то есть каждой МК, отдельно представленной в ГСА ставится в соответствие отдельное состояние автомата Мура.

В логических вершинах ГСА, реализующих режим ожидания, существует возвратная дуга, когда один из выходов вершины подан на ее вход. На этой дуге необходимо вводить дополнительное фиктивное состояние автомата Мура.

Получается ГСА, размеченная для модели Мили символами a0..a9 , для модели Мура символами b0..b13.

Отмеченная граф–схема алгоритма представлена на рисунке 3 и в приложении В.

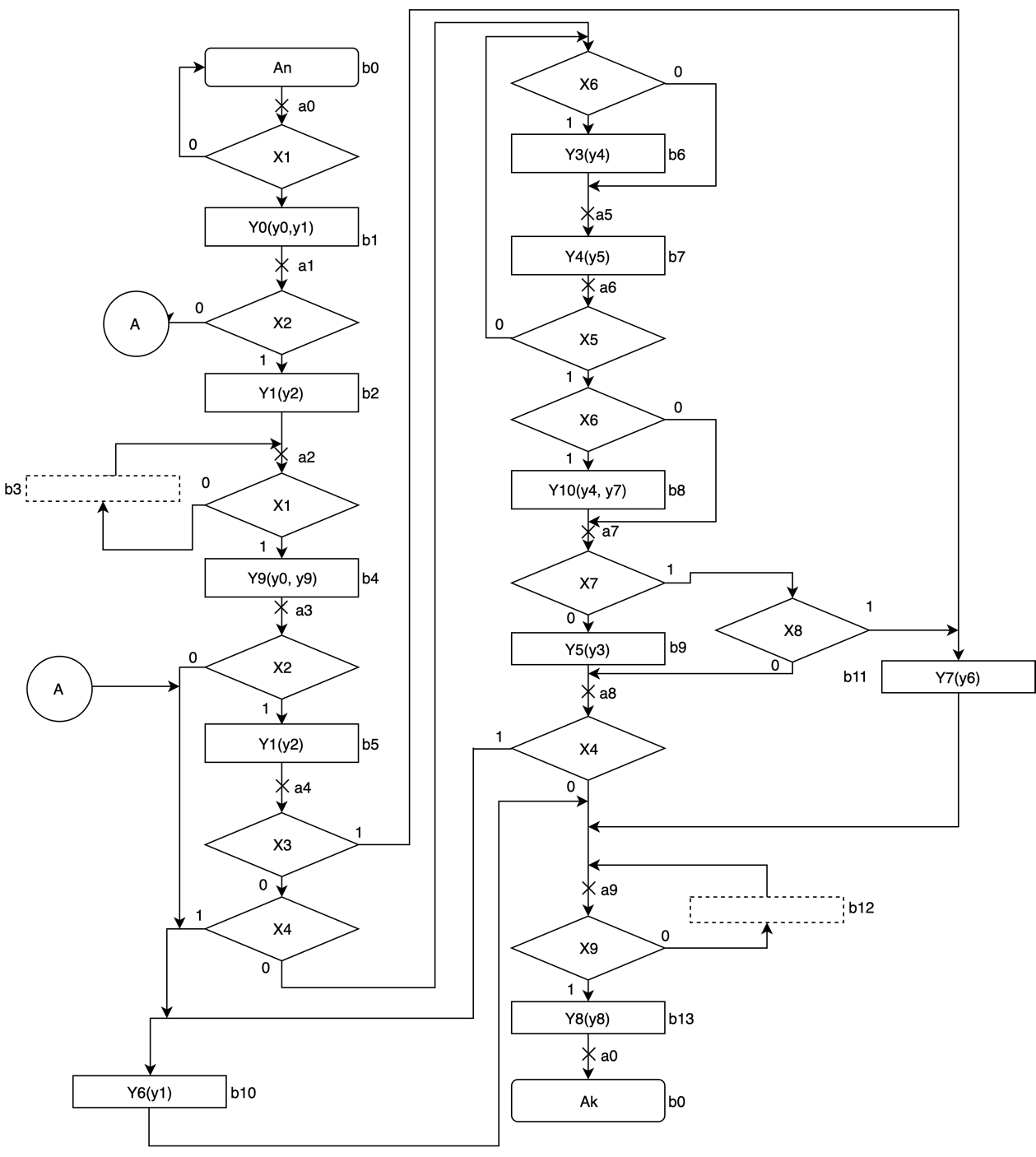


Рисунок 3 - Отмеченxная ГСА для алгоритма умножения

Рисунок 3 - Отмеченная граф-схема алгоритма

# Построение графов автоматов моделей Мили и Мура и выбор структурной схемы управляющего автомата

Граф автомата модели Мили имеет 10 вершин, соответствующих состояниям автомата а0…a9. Дуги его отмечены входными сигналами X1…X9, действующими на каждом переходе, и набором выходных сигналов y0…y9, вырабатываемых управляющим автоматом на данном переходе. Граф автомата модели Мили представлен на рисунке 4 и в приложении Г.

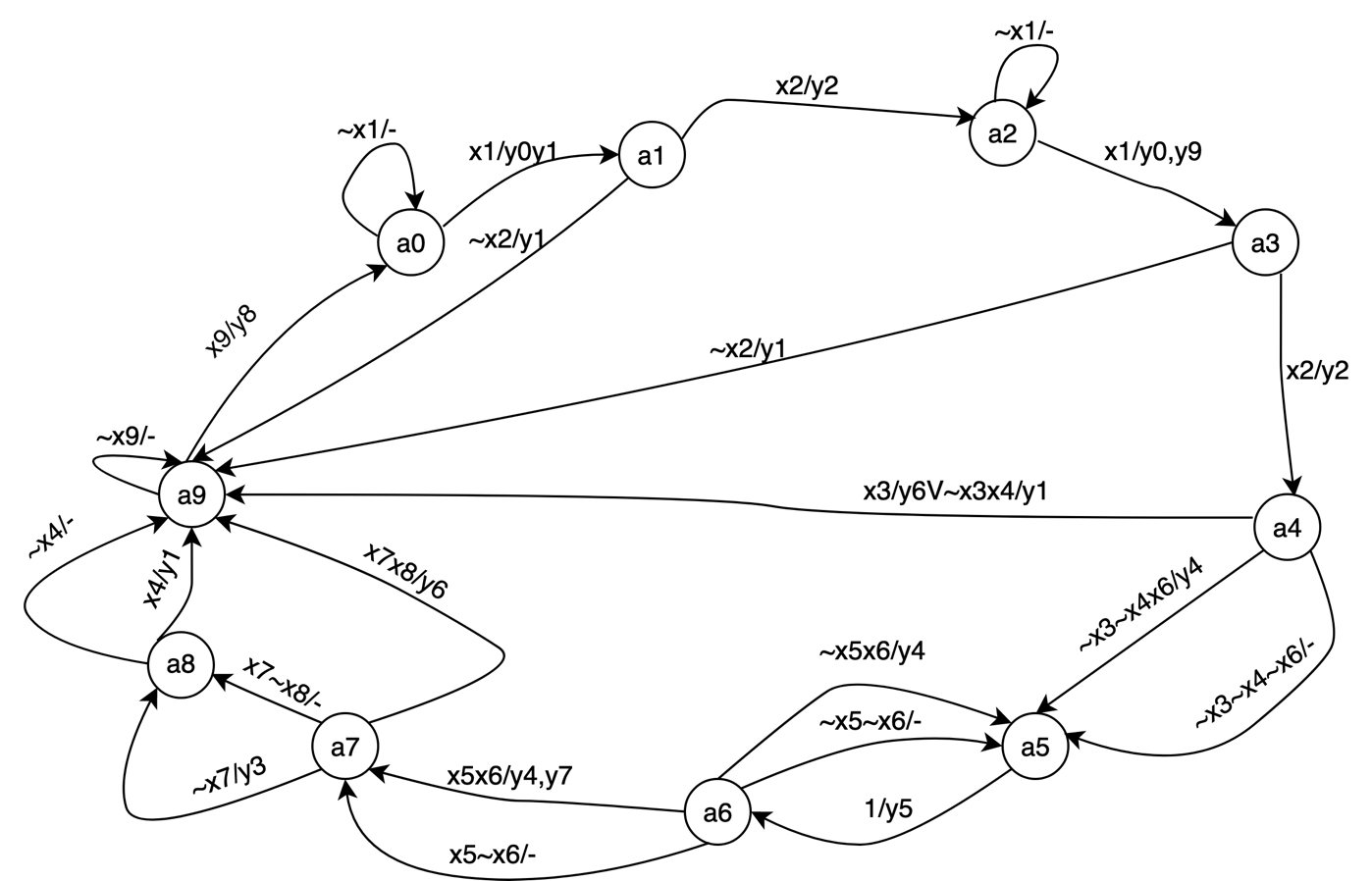


Рисунок 4 - Граф автомата модели Мили

Рисунок 5 - Граф автомата модели Мили с раскрытыми пустыми переходами

Граф автомата Мура имеет 14 вершин, соответствующих состояниям автомата b0…b13. Каждое состояние определяет наборы выходных сигналов y0…y9 управляющего автомата, а дуги графа отмечены входными сигналами X1...X9, действующими на данном переходе. Граф автомата модели Мура представлен на рисунке 5 и в приложении Д.

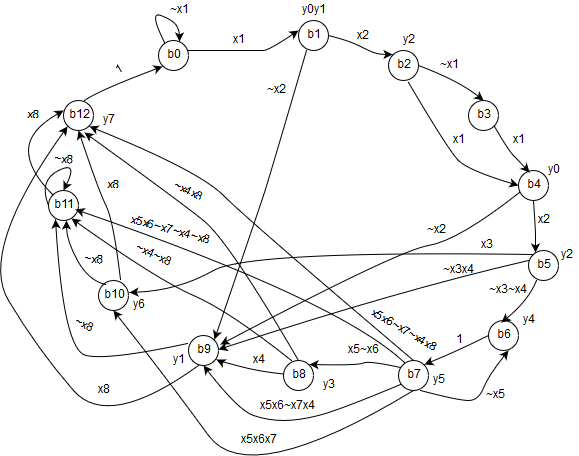


Рисунок 5 - Граф автомата модели Мура

Основываясь на том виде, который принимает граф Мили (большое количество последовательных переходов и незначительное число нестандартных), можно сделать вывод, что при использовании последовательного кодирования, счетчика и дешифратора есть вероятность построения варианта УУ, близкого к оптимальному (минимальная сумма по Квайну). Но для более точной оценки необходим более детальный анализ. А именно, сравнение схем УУ, построенных на D-триггерах с дешифратором, на RS-триггерах с дешифратором с использованием эвристического кодирования, на счетчике с дешифратором, а также на сдвиговом регистре с использованием унитарного кодирования.

При использовании D–триггеров в качестве ЭП при переходе из одного состояния в другое сигналы возбуждения должны быть поданы только на те триггеры, которые в коде состояния содержат единицу. Отсюда следует, что для получения комбинационной схемы меньшей сложности следует использовать следующий метод кодирования: чем больше переходов в какое–либо состояние, тем меньше единиц должен содержать код этого состояния.

Для RS–триггеров лучше использовать соседнее кодирование, так как именно этот способ минимизирует число переключений ЭП. Если соседнее кодирование невозможно, то применяется эвристический метод кодирования состояний.

При использовании счетчика в качестве элемента памяти необходимо придерживаться последовательного кодирования.

Для кодирования состояний автомата на сдвиговом регистре применяется унитарное кодирование.

Основываясь на том виде, который принимает граф автомата модели Мили можно предположить, что кодирование с использованием счетчика или D – триггера будет наиболее эффективным.

# Кодирование внутренних состояний для модели Мили

## Кодирование внутренних состояний для модели Мили на D – триггерах

Для кодирования 10 состояний a0...a9 графа автомата по модели Мили, представленного в приложении Г, минимально необходимо четыре элемента памяти. С учетом особенностей работы D–триггера для кодирования состояний применяется эвристический метод. Он состоит в следующем:

* Каждому состоянию ai ставится в соответствие целое число Ni, равное числу переходов в данное состояние;
* Числа Ni сортируются в порядке убывания;
* Состоянию, соответствующему первому Ni после сортировки, то есть наибольшему из Ni, присваивается код, состоящий только из нулей;
* Следующему состоянию в порядке убывания Ni присваивается незанятый код, содержащий наименьшее количество единиц. Данный пункт повторяется до тех пор, пока все состояния не будут закодированы.

Кодирование состояний для модели Мили на D–триггерах представлено в таблице 7.

Таблица 7 - Коды состояний для модели Мили на D–триггерах

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Состояние | a0 | a1 | a2 | a3 | a4 | a5 | a6 | a7 | a8 | a9 |
| Переходы | a0,  a9 | a0 | a1,  a2 | a2 | a3 | a4,  a6 | a5 | a6 | a7 | a1,  a3,  a4,  a7,  a8,  a9 |
| Число  переходов | 2 | 1 | 2 | 1 | 1 | 4 | 1 | 2 | 2 | 8 |
| Код | 0010 | 0101 | 0100 | 1001 | 0110 | 0001 | 1010 | 1000 | 0011 | 0000 |

Далее составляется прямая структурная таблица переходов и выходов автомата по модели Мили, результаты которой представлены в таблице 8, и формируются логические выражения для функций возбуждения.

Таблица 8 - Прямая структурная таблица переходов и выходов автомата модели Мили на D–триггерах

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Исходное  состояние  am | Код  am | Состояние  перехода  as | Код  as | Входной  сигнал  X(am,as) | Выходные  сигналы  Y(am,as) | Функции  возбуждения  D  триггера |
| a0 | 0010 | a0  a1 | 0010  0101 | ~x1  x1 | -  y0y1 | D1  D2D0 |
| a1 | 0101 | a2  a9 | 0100  0000 | x2  ~x2 | y2  y1 | D2  - |
| a2 | 0100 | a2  a3 | 0100  1001 | ~x1  x1 | -  y0y9 | D2  D3D1 |
| a3 | 1001 | a4  a9 | 0110  0000 | x2  ~x2 | y2  y1 | D2D1  - |
| a4 | 0110 | a5  a5  a9  a9 | 0001  0001  0000  0000 | ~x3~x4x6  ~x3~x4~x6  ~x3x4  x3 | y4  -  y1  y6 | D0  D0  -  - |
| a5 | 0001 | a6 | 1010 | 1 | y5 | D3D1 |
| a6 | 1010 | a7  a7  a5  a5 | 1000  1000  0001  0001 | x5x6  x5~x6  ~x5x6  ~x5~x6 | y4y7  -  y4  - | D3  D3  D0  D0 |
| a7 | 1000 | a8  a8  a9 | 0011  0011  0000 | ~x7  x7~x8  x7x8 | y3  -  y6 | D1D0  D1D0  - |
| a8 | 0011 | a9  a9 | 0000  0000 | x4  ~x4 | y1  - | -  - |
| a9 | 0000 | a0  a9 | 0010  0000 | x9  ~x9 | y8  - | D1  - |

Логические выражения для каждой функции возбуждения D–триггера получают по таблице, как конъюнкции соответствующих исходных состояний am и входных сигналов, которые объединены знаками дизъюнкции для всех строк, содержащих данную функцию возбуждения:

D0 = a0x1 V a4~x3~x4x6 V ~x3~x4~x6 V a6~x5x6 V a6~x5~x6 V a7~x7 V a7x7~x8

D1 = a0~x1 V a2x1 V a3x2 V a5 V a7~x7 V a7x7~x8 V a9x9

D2 = a0x1 V a1x2 V a2~x1 V a3x2

D3 = a2x1 V a5 V a6x5x6 V a6x5~x6

Аналогично составляются логические выражения для функций выходов:

y0 = a0x1 V a2x1

y1 = a0x1 V a1~x2 V a3~x2 V a4~x3x4 V a8x4

y2 = a1x2 V a3x2

y3 = a7~x7

y4 = a4~x3~x4x6 V a6x5x6 V a6~x5x6

y5 = a5

y6 = a4x3 V a7x7x8

y7 = a6x5x6

y8 = a9x9

y9 = a2x1

После выделения общих частей в логических выражениях и некоторого их упрощения получаем логические уравнения для построения функциональной схемы управляющего автомата:

D0 = a V d V a6~x5 V f (6)

D1 = a0~x1 V y9 V g V a5 V f V y7 (8)

D2 = a V y2 V a2~x1 (5)

D3 = y9 V a5 V a6x5 (5)

y0 = a V y9 (2)

y1 = a V a1~x2 V a3~x2 V cx4 V a8x4 (13)

y2 = a1x2 V g (4)

y3 = a7~x7 (2)

y4 = dx6 V a6x6 (6)

y5 = a5 (0)

y6 = a4x3 V ex8 (6)

y7 = a6x5x6 (3)

y8 = a9x9 (2)

y9 = a2x1 (2)

a = a0x1 (2)

c = a4~x3 (2)

d = c~x4 (2)

e = a7x7 (2)

f = y3 V e~x8 (4)

g = a3x2 (2)

Инверторы (ИНВ): X̅1, X̅2, X̅3, X̅4, X̅5, X̅7, X̅8 (7)

Цена комбинационной схемы по Квайну автомата по модели Мили при использовании графа, построенного на основе ГСА, который представлен в приложении Г, с использованием в качестве элементов памяти 4 D–триггеров:

∑ = КС + ИНВ + ЭП + НУ + 𝐷𝐶 = 78+7+16+0+4=105

Схема начальной установки (НУ) для D–триггеров в качестве ЭП приведена на рисунке 6, где D0, D1, D2, D3 – функции возбуждения соответствующих ЭП.

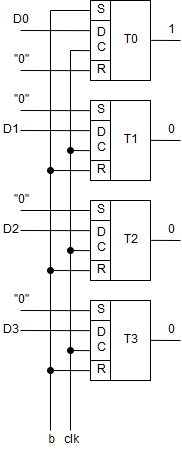


Рисунок 6 - Схема формирования начальной установки на D–триггерах

## Кодирование внутренних состояний для модели Мили на RS – триггерах

Для кодирования 10 состояний автомата Мили, представленного в приложении Г, на RS–триггерах так же потребуется 4 триггера. Наиболее оптимальным способом кодирования для RS–триггеров является соседнее кодирование. Данный граф не получится полностью закодировать по принципу соседнего кодирования, так как в нем присутствуют циклы с нечетным числом вершин. Следовательно, для минимизации числа переключений триггеров при переходе из одного состояния в другое необходимо применить эвристический метод кодирования. Данный метод минимизирует суммарное число переключений элементов памяти на всех переходах автомата. Уменьшение числа переключений триггеров приводит к уменьшению количества единиц соответствующих функций возбуждения, что однозначно приводит к упрощению комбинационной схемы автомата.

Произведем кодирование состояний автомата эвристическим методом кодирования:

1) Составим матрицу *|T|* пар переходов.

|  |  |  |
| --- | --- | --- |
| 0 | 1 |  |
| 1 | 2 |  |
| 1 | 9 |  |
| 2 | 3 |  |
| 3 | 4 |  |
| 3 | 9 |  |
| 4 | 5 |  |
| 4 | 9 |  |
| 5 | 6 |  |
| 6 | 5 |  |
| 6 | 7 |  |
| 7 | 8 |  |
| 7 | 9 |  |
| 8 | 9 |  |
| 9 | 0 |  |

p(0) + p(1) = 5

p(1) + p(2) = 5

p(1) + p(9) = 9

p(2) + p(3) = 5

p(3) + p(4) = 6

p(3) + p(9) = 9

p(4) + p(5) = 6

p(4) + p(9) = 9

p(5) + p(6) = 6

p(6) + p(5) = 6

p(6) + p(7) = 6

p(7) + p(8) = 5

p(7) + p(9) = 9

p(8) + p(9) = 8

p(9) + p(0) = 8

p(0) = 2 p(1) = 3

p(1) = 3 p(2) = 2

p(1) = 3 p(9) = 6

p(2) = 2 p(3) = 3

p(3) = 3 p(4) = 3

p(3) = 3 p(9) = 6

p(4) = 3 p(5) = 3

p(4) = 3 p(9) = 6

p(5) = 3 p(6) = 3

p(6) = 3 p(5) = 3

p(6) = 3 p(7) = 3

p(7) = 3 p(8) = 2

p(7) = 3 p(9) = 6

p(8) = 2 p(9) = 6

p(9) = 6 p(0) = 2

|T|=

2) Упорядочим строки матрицы |𝑇|, для чего строим матрицу |𝑀| следующим образом: отсортируем матрицу так чтобы вверху были наиболее встречающиеся состояния, при этом в каждой следующей строке, кроме первой, содержался хотя бы один уже закодированный элемент. Формирование матрицы |𝑀| заканчивается, когда все элементы матрицы |𝑇| размещены в матрице |𝑀|:

|  |  |
| --- | --- |
| 1 | 9 |
| 3 | 9 |
| 4 | 9 |
| 7 | 9 |
| 8 | 9 |
| 9 | 0 |
| 3 | 4 |
| 4 | 5 |
| 5 | 6 |
| 6 | 5 |
| 6 | 7 |
| 0 | 1 |
| 1 | 2 |
| 2 | 3 |
| 7 | 8 |

*|M|=*

3) Закодируем первые 2 состояния:

**𝑎1=0000; 𝑎9=0001;**

Вычеркнем из матрицы |𝑀| первую строку, соответствующую закодированным состояниям 𝑎9 и 𝑎1. Далее в матрице находиться первый не закодированное состояние as, для него составляется собственная матрица M\_s, состоящая из пар переходов, содержащих это состояние. Найдем множество 𝐷, где элементами множества являются коды, соседние для уже закодированных состояний, которые присутствуют в матрице M\_s. Для каждого кода из множества 𝐷 определяем суммарное количество переключений триггера при кодировании состояния as данным кодом. Код, который потребует минимальное число переключений назначается состоянию as.

Для определения эффективности кодирования применяется коэффициент 𝑘, который является отношением общего количества переключений триггеров к общему количеству переходов, где состояния закодированы с помощью эвристического метода кодирования:

Эффективность кодирования: 𝑘 = 20/15 = 1,33;

Получившиеся коды состояний представлены в таблице 9.

Таблица 9 - Коды состояний для модели Мили на RS–триггерах

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a0 | a1 | a2 | a3 | a4 | a5 | a6 | a7 | a8 | a9 |
| 0010 | 0000 | 1000 | 0011 | 0111 | 1111 | 1101 | 0101 | 1101 | 0001 |

Далее составляется прямая структурная таблица переходов и выходов автомата модели Мили, представленная в таблице 10, и по известному правилу формируются логические выражения для функций возбуждения. Прямая структурная таблица переходов и выходов автомата модели Мили на RS–триггерах.

Таблица 10 - Прямая структурная таблица переходов и выходов автомата модели Мили на RS– триггерах

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Исходное  состояние  am | Код  am | Состояние  перехода  as | Код  as | Входной  сигнал  X(am,as) | Выходные  сигналы  Y(am,as) | Функции  возбуждения  D  триггера |
| a0 | 0010 | a0  a1 | 0010  0000 | ~x1  x1 | -  y0y1 | -  R1 |
| a1 | 0000 | a2  a9 | 1000  0001 | x2  ~x2 | y2  y1 | S3  S0 |
| a2 | 1000 | a2  a3 | 1000  0011 | ~x1  x1 | -  y0y9 | -  R3S1S0 |
| a3 | 0011 | a4  a9 | 0111  0001 | x2  ~x2 | y2  y1 | S2  S1 |
| a4 | 0111 | a5  a5  a9  a9 | 1111  1111  0001  0001 | ~x3~x4x6  ~x3~x4~x6  ~x3x4  x3 | y4  -  y1  y6 | S3  S3  R2R1  R2R1 |
| a5 | 1111 | a6 | 1101 | 1 | y5 | R1 |
| a6 | 1101 | a7  a7  a5  a5 | 0101  0101  1111  1111 | x5x6  x5~x6  ~x5x6  ~x5~x6 | y4y7  -  y4  - | R3  R3  S1  S1 |
| a7 | 0101 | a8  a8  a9 | 1101  1101  0001 | ~x7  x7~x8  x7x8 | y3  -  y6 | S3  S3  R2 |
| a8 | 1101 | a9  a9 | 0001  0001 | x4  ~x4 | y1  - | R3R2  R3R2 |
| a9 | 0001 | a0  a9 | 0010  0001 | x9  ~x9 | y8  - | S1R0  - |

Логические выражения для каждой функции возбуждения RS–триггера получают по таблице, как конъюнкции соответствующих исходных состояний am и входных сигналов, которые объединены знаками дизъюнкции для всех строк, содержащих данную функцию возбуждения:

S0 = a1~x2 V a2x1

S1 = a2x1 V a3~x2 V a6~x5x6 V a6~x5~x6 V a9x9

S2 = a3x2

S3 = a1x2 V a4~x3~x4x6 V a4~x3~x4~x6 V a7~x7 V a7x7~x8

R0 = a9x9

R1 = a0x1 V a4~x3x4 V a4x3 V a5

R2 = a4~x3x4 V a4x3 V a7x7x8 V a8x4 V a8~x4

R3 = a2x1 V a6x5x6 V a6x5~x6 V a8x4 V a8~x4

Аналогично составляются логические выражения для функций выходов:

y0 = a0x1 V a2x1

y1 = a0x1 V a1~x2 V a3~x2 V a4~x3x4 V a8x4

y2 = a1x2 V a3x2

y3 = a7~x7

y4 = a4~x3~x4x6 V a6x5x6 V a6~x5x6

y5 = a5

y6 = a4x3 V a7x7x8

y7 = a6x5x6

y8 = a9x9

y9 = a2x1

После выделения общих частей в логических выражениях и некоторого их упрощения, получаем логические уравнения для построения функциональной схемы управляющего автомата.

S0 = a V y9 (2)

S1 = y9 V c V a6~x5 V y8 (6)

S2 = a3x2 (2)

S3 = d V g V y3 V i~x8 (6)

R0 = y8 (0)

R1 = h V f V a4x3 V a5 (6)

R2 = f V y6 V a8 (3)

R3 = y9 V k V a8 (3)

y0 = h V y9 (2)

y1 = h V a V c V f V a8x4 (7)

y2 = d V S2 (2)

y3 = a7~x7 (2)

y4 = gx6 V a6x6 (6)

y5 = a5 (0)

y6 = a4x3 V ix8 (6)

y7 = kx6 (2)

y8 = a9x9 (2)

y9 = a2x1 (2)

a = a1~x2 (2)

c = a3~x2 (2)

d = a1x2 (2)

e = a4~x3 (2)

f = ex4 (2)

g = e~x4 (2)

h = a0x1 (2)

i = a7x7 (2)

k = a6x5 (2)

Инверторы (ИНВ): X̅2, X̅3, X̅4, X̅5, X̅7, X̅8 (6)

Цена комбинационной схемы по Квайну автомата по модели Мили при использовании графа, построенного на основе ГСА, который представлен в приложении Г, с использованием в качестве элементов памяти 4 RS –триггеров:

∑ = КС + ИНВ + ЭП + НУ + 𝐷𝐶 =77+6+12+17+4 = 116;

Схема начальной установки для RS–триггеров в качестве ЭП приведена на рисунке 7, где S0, R0, S1, R1, S2, R2, S3, R3 – функции возбуждения соответствующих ЭП, а b – сигнал начальной установки.

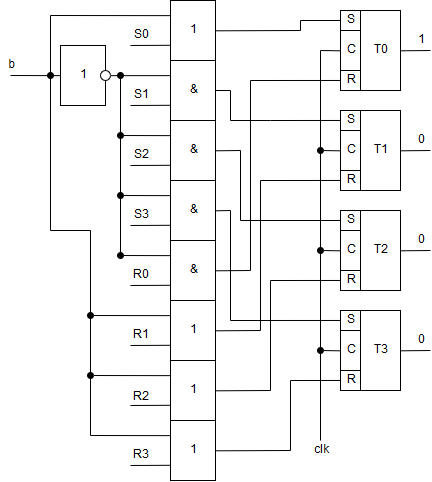


Рисунок 7 - Схема формирования начальной установки на RS–триггерах

## Кодирование внутренних состояний для модели Мили на счетчике

При кодировании состояний на счетчике необходимо стараться закодировать состояния таким образом, чтобы код состояния, отличающегося от предыдущего порядковым номером на единицу, был больше или меньше на единицу, так как счетчик имеет входные управляющие сигналы «+1», «–1», «Reset». Если при кодировании возникает ситуация, когда происходит переход в состояние отличное от предыдущего состояния более чем на единицу, необходимо обнулять счётчик сигналом “Reset”. Для кодирования 10 состояний автомата по модели Мили потребуется 4–х разрядный счетчик. Получившиеся коды состояний представлены в таблице 11.

Таблица 11 - Коды состояний для модели Мили на счетчике

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a0 | a1 | a2 | a3 | a4 | a5 | a6 | a7 | a8 | a9 |
| 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 | 0000 |

Далее составляется прямая структурная таблица переходов и выходов автомата Мили на счетчике, результаты которой представлены в таблице 12, и по известному правилу формируются логические выражения для функций возбуждения.

Таблица 12 - Прямая структурная таблица переходов и выходов автомата модели Мили на счетчике

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Исходное  состояние  am | Код  am | Состояние  перехода  as | Код  as | Входной  сигнал  X(am,as) | Выходные  сигналы  Y(am,as) | Функции  возбуждения  D  триггера |
| a0 | 0001 | a0  a1 | 0001  0010 | ~x1  x1 | -  y0y1 | -  +1 |
| a1 | 0010 | a2  a9 | 0011  0000 | x2  ~x2 | y2  y1 | +1  R |
| a2 | 0011 | a2  a3 | 0011  0100 | ~x1  x1 | -  y0y9 | -  +1 |
| a3 | 0100 | a4  a9 | 0101  0000 | x2  ~x2 | y2  y1 | +1  R |
| a4 | 0101 | a5  a5  a9  a9 | 0110  0110  0000  0000 | ~x3~x4x6  ~x3~x4~x6  ~x3x4  x3 | y4  -  y1  y6 | +1  +1  R  R |
| a5 | 0110 | a6 | 0111 | 1 | y5 | +1 |
| a6 | 0111 | a7  a7  a5  a5 | 1000  1000  0110  0110 | x5x6  x5~x6  ~x5x6  ~x5~x6 | y4y7  -  y4  - | +1  +1  -1  -1 |
| a7 | 1000 | a8  a8  a9 | 1001  1001  0000 | ~x7  x7~x8  x7x8 | y3  -  y6 | +1  +1  R |
| a8 | 1001 | a9  a9 | 0000  0000 | x4  ~x4 | y1  - | R  R |
| a9 | 0000 | a0  a9 | 0001  0000 | x9  ~x9 | y8  - | +1  - |

Логические выражения для каждой функции возбуждения получаются по таблице, как конъюнкции соответствующих исходных состояний 𝑎𝑚 и входных сигналов, которые объединены знаками дизъюнкции для всех строк, содержащих данную функцию возбуждения.

+1 = a0x1 V a1x2 V a2x1 V a3x2 V a4~x3~x4 V a5 V a6x5 V a7~x7 V a7x7~x8 V a9x9

-1 = a6~x5

R = a1~x2 V a3~x2 V a4x3 V a4~x3x4 V a7x7x8 V a8 V a9~x9

Аналогично составляются логические выражения для функций выходов:

y0 = a0x1 V a2x1

y1 = a0x1 V a1~x2 V a3~x2 V a4~x3x4 V a8x4

y2 = a1x2 V a3x2

y3 = a7~x7

y4 = a4~x3~x4x6 V a6x5x6 V a6~x5x6

y5 = a5

y6 = a4x3 V a7x7x8

y7 = a6x5x6

y8 = a9x9

y9 = a2x1

После выделения общих частей в логических выражениях и некоторого их упрощения, получаем логические уравнения для построения функциональной схемы управляющего автомата.

+1 = a0x1 V a1x2 V a2x1 V a3x2 V a4~x3~x4 V a5 V a6x5 V a7~x7 V a7x7~x8 V a9x9

-1 = a6~x5

R = c V y6 V a8 V a9~x9

y0 = d V y9 (2)

y1 = d V c V a8x4 (5)

y2 = a1x2 V a3x2 (6)

y3 = a7~x7 (2)

y4 = a~x4x6 V e (5)

y5 = a5 (0)

y6 = a4x3 V a7x7x8 (7)

y7 = ex5 (2)

y8 = a9x9 (2)

y9 = a2x1 (2)

a = a4~x3 (2)

c = a1~x2 V a3~x2 V ax4 (9)

d = a0x1 (2)

e = a6x6 (2)

f = a0~x1 (2)

g = a2~x1 (2)

h = a9~x9 (2)

i = a6~x5 (2)

По данным логическим функциям видно, что запись в счетчик во время работы УУ не производится (она нужна только в момент сброса УУ до начала его работы). Следовательно, в процессе работы УУ используются только счетные входы и вход сброса. Логическую функцию для счетного входа «+1» можно еще упростить. Если в определенный момент времени формирования следующего внутреннего состояния УУ поступает сигнал на R, то в этот же момент времени информация не должна повлиять на срабатывание счетных входов. Если используется счетчик с дополнительным входом разрешения счета и счетный вход работает по принципу «1» - счетчик работает на сложение, «0» - счетчик работает на вычитание, то логическая функция для входа разрешения счета E = ~R. В тот момент времени, когда информация влияет на срабатывание счетных входов, необходимо разграничить две ситуации: +1 и -1. Поскольку «+1» не используется в момент, когда срабатывает «-1», то для «+1» логическая функция будет равна ~m.

В результате получим:

+1 = ~(i V R V f V g V h) (6)

-1 = i (0)

R = c V y6 V a8 V a9~x9 (6)

Инверторы (ИНВ): X̅1,X̅2, X̅3, X̅4, X̅5, X̅7, X̅9 (7)

Цена комбинационной схемы по Квайну автомата по модели Мили при использовании графа, построенного на основе ГСА, который представлен в приложении Г, с использованием в качестве элемента памяти 4–х разрядного счетчика:

∑ = КС + ИНВ + ЭП + НУ + 𝐷𝐶 =68+7+9+0+4=86;

Схема начальной установки для счетчика в качестве ЭП приведена на рисунке 8, где b – сигнал начальной установки.

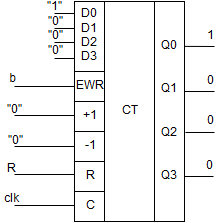


Рисунок 8 - Схема формирования начальной установки на счетчике

# Кодирование внутренних состояний для модели Мура

## Кодирование внутренних состояний для модели Мура на D – триггерах

Для кодирования 13 состояний (b0…b12) графа автомата Мура, представленного в приложении Д, минимально необходимо четыре элемента памяти. С учетом особенностей работы D–триггера для кодирования состояний применяется эвристический метод. Он состоит в следующем:

* Каждому состоянию bi ставится в соответствие целое число Ni, равное числу переходов в данное состояние;
* Числа Ni сортируются в порядке убывания;
* Состоянию, соответствующему первому Ni после сортировки, то есть наибольшему из Ni, присваивается код, состоящий только из нулей;
* Следующему состоянию в порядке убывания Ni присваивается незанятый код, содержащий наименьшее количество единиц. Данный пункт повторяется до тех пор, пока все состояния не будут закодированы.

Кодирование состояний для модели Мура на D–триггерах представлено в таблице 15.

Таблица 15 - Коды состояний для модели Мура на D–триггерах

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Исходное  состояние | b0 | b1 | | b2 | b3 | b4 | b5 | b6 | b7 | b8 | b9 | b10 |
| Состояния перехода | b0  b12 | b0 | | b1 | b2  b3 | b2  b3 | b4 | b5 | b6 | b7 | b1  b4  b5  b7  b8 | b5  b7 |
| Число переходов | 2 | 1 | | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 5 | 2 |
| Код состояния | 0100 | 0101 | | 1001 | 0110 | 1000 | 1010 | 1100 | 0111 | 1011 | 0000 | 0011 |
| Исходное  состояние | b11 | b12 |
| Состояния перехода | b7  b8  b9  b10  b11 | b7  b8  b9  b10  b11 |
| Число переходов | 5 | 5 |
| Код состояния | 0001 | 0010 |

Далее составляется прямая структурная таблица переходов и выходов автомата модели Мура на D–триггерах, результаты которой представлены в таблице 16 и формируются логические выражения для функций возбуждения.

Таблица 16 - Прямая структурная таблица переходов и выходов модели Мура на D–триггерах

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Исходное Состояние bm | Код  bm | Выходной сигнал  Y(bm) | Состояние перехода  bs | Код  bs | Входной сигнал  X(bm, bs) | Функции возбуждения D–триггера |
| b0 | 0100 | - | b0  b1 | 0100  0101 | ~x1  x1 | D2  D2D0 |
| b1 | 0101 | y0y1 | b2  b9 | 1001  0000 | x2  ~x2 | D3D0  - |
| b2 | 1001 | y2 | b3  b4 | 0110  1000 | ~x1  x1 | D2D1  D3 |
| b3 | 0110 | - | b3  b4 | 0110  1000 | ~x1  x1 | D2D1  D3 |
| b4 | 1000 | y0 | b5  b9 | 1010  0000 | x2  ~x2 | D3D1  - |
| b5 | 1010 | y2 | b6  b9  b10 | 1100  0000  0011 | ~x3~x4  ~x3x4  x3 | D3D2  -  D1D0 |
| b6 | 1100 | y4 | b7 | 0111 | 1 | D2D1D0 |
| b7 | 0111 | y5 | b6  b8  b9  b10  b11  b12 | 1100  1011  0000  0011  0001  0010 | ~x5  x5~x6  x4  x5x6x7  x5x6~x7~x4~x8  x5x6~x7~x4x8 | D3D2  D3D1D0  -  D1D0  D0  D1 |
| b8 | 1011 | y3 | b9  b11  b12 | 0000  0001  0010 | x4  ~x4~x8  ~x4x8 | -  D0  D1 |
| b9 | 0000 | y1 | b11  b12 | 0001  0010 | ~x8  x8 | D0  D1 |
| b10 | 0011 | y6 | b11  b12 | 0001  0010 | ~x8  x8 | D0  D1 |
| b11 | 0001 | - | b11  b12 | 0001  0010 | ~x8  x8 | D0  D1 |
| b12 | 0010 | y7 | b0 | 0100 | 1 | D2 |

Логические выражения для каждой функции возбуждения D–триггера получают по таблице как конъюнкции соответствующих исходных состояний bm и входных сигналов, которые объединены знаками дизъюнкции для всех строк, содержащих данную функцию возбуждения:

D0 = b0x1 V b1x2 V b5x3 V b6 V b7x5~x6 V b7x5x6x7 V b7x5x6~x7~x4~x8 V b8~x4~x8 V b9~x8 V b10~x8 V b11~x8

D1 = b2~x1 V b3~x1 V b4x2 V b6 V b7x5~x6 V b7x5x6x7 V b7x5x6~x7~x4x8 V b8~x4x8 V b9x8 V b10x8 V b11x8

D2 = b0~x1 V b0x1 V b2~x1 V b3~x1 V b5~x3~x4 V b6 V b7~x5 V b12

D3 = b1x2 V b2x1 V b3x1 V b4x2 V b5~x3~x4 V b7~x5 V b7x5~x6

Аналогично составляются логические выражения для функций выходов:

y0 = b1 V b4

y1 = b1 V b9

y2 = b2 V b5

y3 = b8

y4 = b6

y5 = b7

y6 = b10

y7 = b12

После выделения общих частей в логических выражениях и некоторого их упрощения получаем логические уравнения для построения функциональной схемы управляющего автомата:

D0 = b0x1 V c V i V k~x8 V l~x8 V b9~x8 V b10~x8 V b11~x8 (20)

D1 = g V b4x2 V i V kx8 V lx8 V b9x8 V b10x8 V b11x8 (20)

D2 = b0 V g V h V b6 V b12 (5)

D3 = c V b2x1 V b3x1 V b4x2 V h V f (12)

y0 = b1 V b4 (2)

y1 = b1 V b9 (2)

y2 = b2 V b5 (2)

y3 = b8 (0)

y4 = b6 (0)

y5 = b7 (0)

y6 = b10 (0)

y7 = b12 (0)

c = b1x2 (2)

d = b5x3 V b6 (4)

e = b7x5 (2)

f = e~x6 (2)

g = b2~x1 V b3~x1 (6)

h = b5~x3~x4 V b7~x5 (7)

i = d V f V b7x5x6x7 (7)

k = ex6~x7~x4 (4)

l = b8~x4 (2)

Инверторы: X̅1, X̅2, X̅3, X̅4, X̅5, X̅6, X̅7, X̅8 (8)

Цена комбинационной схемы по Квайну автомата по модели Мура при использовании графа, построенного на основе ГСА, представленного в приложении Е, с использованием в качестве элементов памяти 5 D–триггеров:

∑ = КС + ИНВ + ЭП + НУ + 𝐷𝐶 =107+8+12+0+4=131;

Схема формирования начальной установки на D–триггерах представлена на рисунке 9.

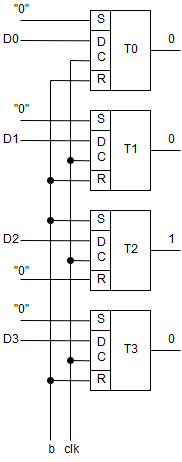


Рисунок 9 - Схема формирования начальной установки на D–триггерах автомата Мура

Цена по Квайну автомата модели Мура на D–триггерах получилась больше (на 31%), чем цена по Квайну автомата модели Мили на D–триггерах. Отсюда можно сделать вывод, что цена по Квайну автомата модели Мура на RS–триггерах не будет минимальной. Кодирование на счетчике для автомата модели Мура также является нецелесообразным, так как граф имеет множество нестандартных переходов, которые составляют более 50% от всех переходов.

# Построение схемы управляющего микропрограммного автомата

Наиболее оптимальной ценой по Квайну, равной 69, и быстродействием обладает модель автомата модели Мили на счетчике поэтому микропрограммный автомат будет строиться для этой модели. В приложении Е приведена функциональная схема проектируемого управляющего автомата, выполняющего операцию умножения чисел в двоичной системе счисления с плавающей запятой в дополнительном коде вторым способом с автоматической коррекцией. Функциональная схема построена в основном логическом базисе «И, ИЛИ, НЕ» в полном соответствии с приведенной для модели Мили системой логических уравнений для функций возбуждения элемента памяти и функций получения выходных сигналов.

# Заключение

В ходе курсового проекта был синтезирован автомат, выполняющий умножение вторым способом в двоичной системе счисления с плавающей запятой с использованием дополнительного кода при умножении мантисс. Управляющий автомат был синтезирован по модели Мили с использованием 4–х разрядного счетчика в качестве элемента памяти, так как цена по Квайну, равная 69, и быстродействие данного автомата получились наиболее оптимальными. Автомат, полученный в ходе выполнения курсового проекта, задан множеством внутренних состояний a0–a8, множеством входных сигналов X1–X8, множеством выходных сигналов y0–y7, функциями переходов и выходов, заданными в таблице 12, начальным состоянием a0.

# Приложение А

(обязательное)

**Функциональная схема операционного автомата**

# Приложение Б

Рисунок 10 - Функциональная схема операционного автомата

(обязательное)

**Содержательная граф – схема алгоритма**

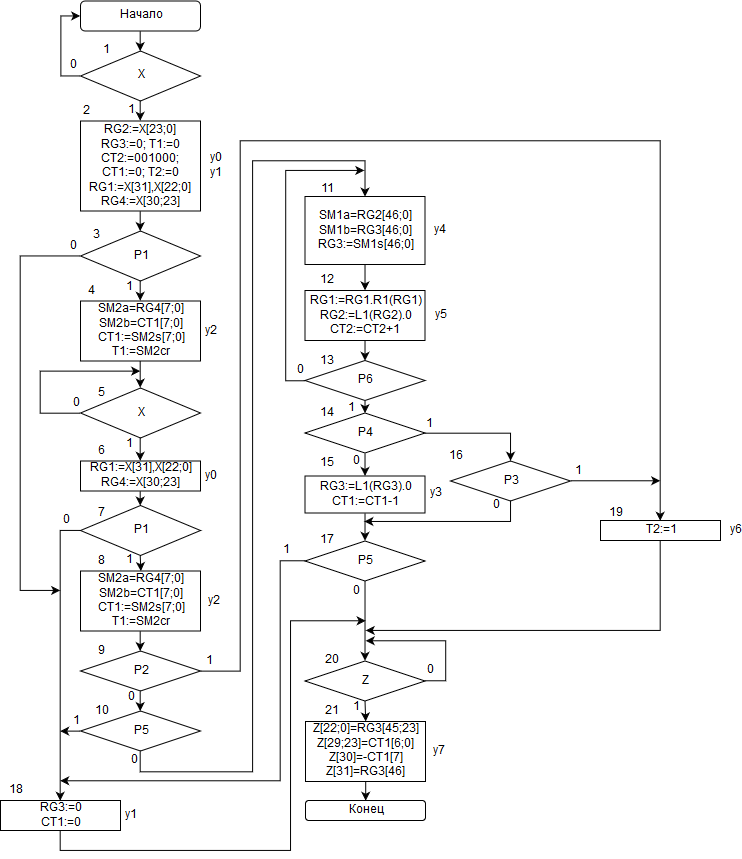


Рисунок 11 – ГСА алгоритма умножения

# Приложение В

(обязательное)

**Отмеченная граф – схема алгоритма**

# Приложение Г

Рисунок 12 - Отмеченная ГСА для алгоритма умножения

(обязательное)

**Граф автомата модели Мили**

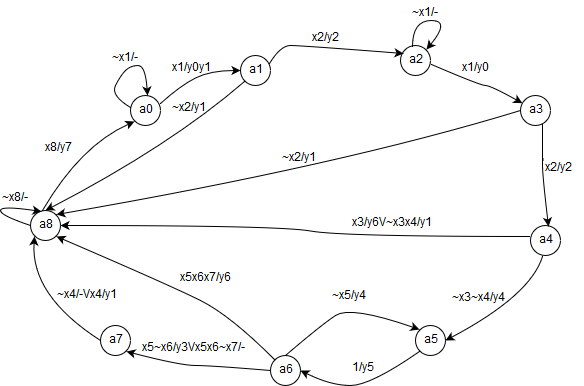


Рисунок 13– Граф автомата модели Мили

Рисунок 13 - Граф автомата модели Мили

Рисунок 14 - Граф автомата модели Мили с 1 раскрытым пустым переходом

# Приложение Д

(обязательное)

**Граф автомата модели Мура**

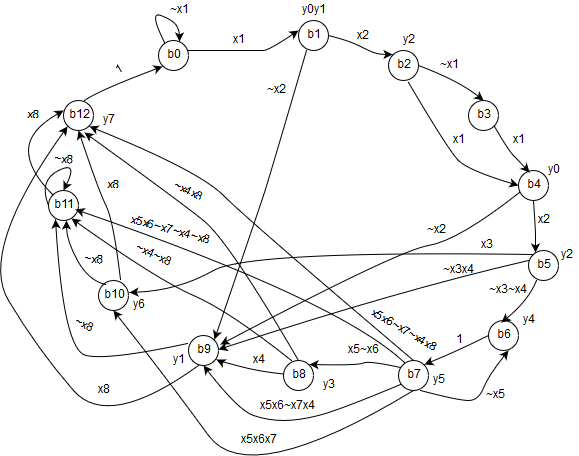


Рисунок 14 – Граф автомата модели Мура

Рисунок 15 - Граф автомата модели Мура

# Приложение Е

(обязательное)

**Функциональная схема управляющего автомата**

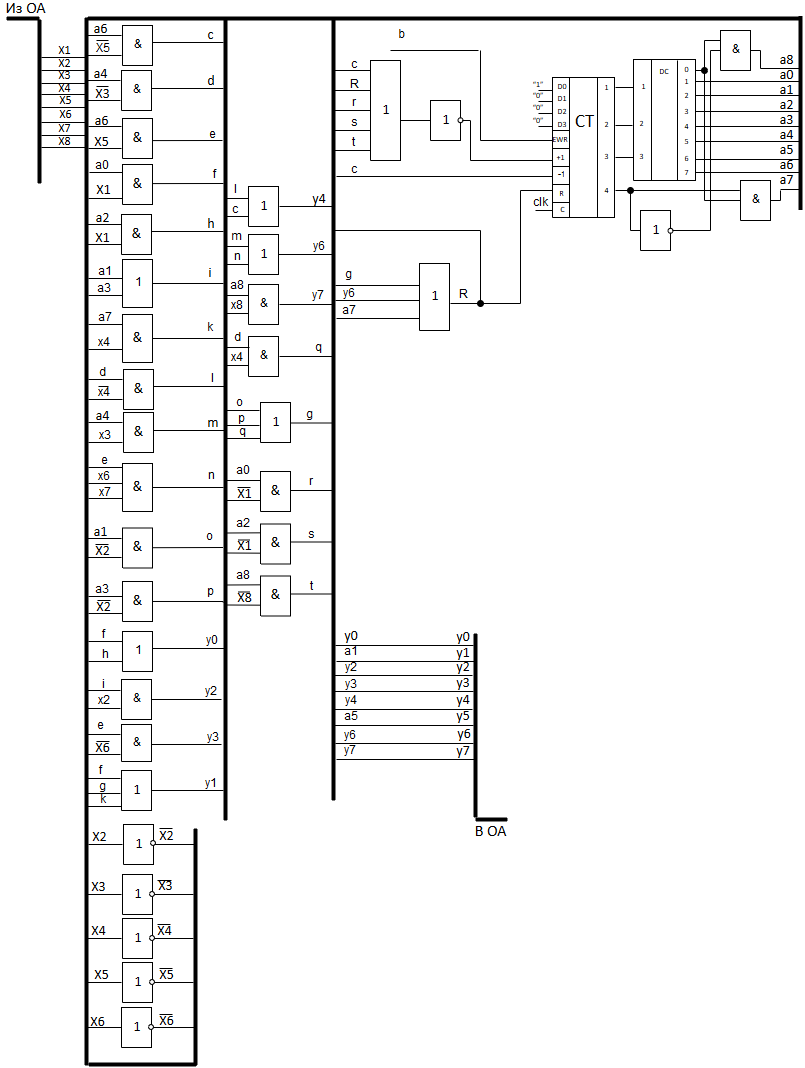


Рисунок 15 - Функциональная схема управляющего автомата

Рисунок 16 - Функциональная схема управляющего автомата

# Библиографический список

*1.* *Мельцов, В.Ю. Синтез микропрограммных управляющих автоматов. Учебное пособие [Текст] / Мельцов, В. Ю., Фадеева, Т.Р. – Киров: Вятский государственный технический университет, 2000. – 69с.*

*2. Мельцов, В.Ю. Применение САПР Quartus для синтеза абстрактных и структурных автоматов. Учебное пособие [Текст] – Киров: ГОУ ВПО ВятГУ, 2011. – 86с.*

*3. Фадеева, Т.Р. Арифметические основы ЭВМ. Методические указания к курсовой работе [Текст] / Фадеева, Т.Р., Матвеева, Л.И., Долженкова, М.Л. – Киров, 2007. – 80с*

*4. Шихов М.М. Курс лекций по дисциплине «Информатика» [Электронный ресурс]–Режим доступа:*

[*https://vyatsu-my.sharepoint.com/:f:/g/personal/stud097115\_vyatsu\_ru/ErEsGrUtvUJPgc0hmqlHe1IB-2UQbJOiYHttHJjshrXmmA?e=GIi6Nf*](https://vyatsu-my.sharepoint.com/:f:/g/personal/stud097115_vyatsu_ru/ErEsGrUtvUJPgc0hmqlHe1IB-2UQbJOiYHttHJjshrXmmA?e=GIi6Nf)

*5. Фадеева Т. Р., Матвеева Л. И., Шихов М. М. Арифметические основы*

*вычислительной техники и элементы микропрограммного управления /*

*Фадеева Т. Р., Матвеева Л. И., Шихов М. М. - 2017. - 166 с.*

*6. ГОСТ 2.743 – 91 ЕСКД. «Обозначения условные графические в схемах. Элементы цифровой техники» [Электронный ресурс] – Режим доступа:*

*http://docs.cntd.ru/document/gost-2-743-91-eskd*