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

\_\_\_\_\_\_\_\_\_\_\_\_ (Исупов К.С .)

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_2019г.

«СИНТЕЗ МИКРОПРОГРАММНОГО УПРАВЛЯЮЩЕГО АВТОМАТА»

Пояснительная записка курсового проекта по дисциплине

«Теория автоматов»

ТПЖА.09.03.01.887 ПЗ

Разработала студентка группы ИВТ-22 \_\_\_\_\_\_\_\_\_\_\_\_/Верхотина М.А./

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

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

*(оценка) (дата)*

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

(подпись)

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

(подпись)

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

(подпись)

Киров 2019

**Реферат**

Верхотина М.А. Синтез микропрограммного управляющего автомата: ТПЖА.090301887 ПЗ: Курс. проект / ВятГУ, каф. ЭВМ; рук. Исупов К.С. - Киров, 2019. Графическая часть 4 л. – ф. А2, ПЗ 61 с, 18 рис., 23 табл., 5 источников, 7 приложений.

СИНТЕЗ АВТОМАТА, МИКРОПРОГРАММНЫЙ УПРАВЛЯЮЩИЙ АВТОМАТ, СХЕМА АЛГОРИТМА, УМНОЖЕНИЕ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ, ПРЯМОЙ КОД, ХАРАКТЕРИСТИКИ

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

Цель курсового проекта – синтез микропрограммного автомата, управляющего выполнением умножения двоичных чисел в прямом коде с плавающей запятой, с характеристиками первым способом.

Результат работы – функциональная схема микропрограммного управляющего автомата.

**Содержание**

1. [Введение 4](#_Toc9190825)
2. [Постановка задачи 5](#_Toc9190826)
3. [Описание используемого алгоритма умножения 6](#_Toc9190827)
4. [Численные примеры 8](#_Toc9190828)
5. [Разработка функциональной схемы операционной части устройства 15](#_Toc9190829)

[Состав операционного автомата 15](#_Toc9190830)

[Описание работы операционного автомата 17](#_Toc9190831)

[Управляющие и осведомительные сигналы 18](#_Toc9190832)

1. [Разработка содержательной граф-схемы алгоритма 20](#_Toc9190833)
2. [Разработка отмеченной граф-схемы алгоритма 22](#_Toc9190834)
3. [Построение графов автоматов моделей Мили и Мура 25](#_Toc9190835)
4. [Кодирование внутренних состояний для модели Мили 28](#_Toc9190836)

[Кодирование внутренних состояний для модели Мили на D – триггерах 28](#_Toc9190837)

[Кодирование внутренних состояний для модели Мили на RS – триггерах 33](#_Toc9190838)

[Кодирование внутренних состояний для модели Мили на счетчике 42](#_Toc9190839)

1. [Кодирование внутренних состояний для модели Мура 47](#_Toc9190840)

[Кодирование внутренних состояний для модели Мура на D – триггерах 47](#_Toc9190841)

1. [Построение схемы управляющего микропрограммного автомата 52](#_Toc9190842)
2. [Заключение 53](#_Toc9190843)
3. [Приложение А 54](#_Toc9190844)
4. [Приложение Б 55](#_Toc9190845)
5. [Приложение В 56](#_Toc9190846)
6. [Приложение Г 57](#_Toc9190847)
7. [Приложение Д 58](#_Toc9190848)
8. [Приложение Е 59](#_Toc9190849)
9. [Приложение Ж 60](#_Toc9190850)
10. [Библиографический список 61](#_Toc9190851)

# Введение

С каждым годом объемы вычислений возрастают, при этом их все сложнее обрабатывать вручную, поэтому ведутся работы по созданию и применению различных автоматизированных систем для обработки информации. Такие автоматы реализуются в виде самостоятельных устройств или в виде блоков, входящих в системы управления, а также системы обработки информации. Работа ведется с математическими моделями, предназначенными для приближенного отображения физических моделей.

Основной целью данного курсового проекта является получение навыков синтеза управляющего микропрограммного автомата (МПА) 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 прямом коде первым способом.

# Описание используемого алгоритма умножения

|  |
| --- |
| Множитель Множимое  ├────┤ ├────┤  ├──┤  ├────|────┤  Сумма частичных произведений |

Рисунок 1 – Графическое изображение первого способа умножения двоичных чисел

Разрядность регистров:

* Множителя и множимого – n
* Суммы частичных произведений – 2n

Множимое следует прибавлять в старшие n разрядов регистра суммы частичных произведений.

**Особенность 1 способа** - в цикле умножения возможно временное ПРС в регистре СЧП, которое ликвидируется при очередном сдвиге вправо.

**Алгоритм:**

**1.** Определить знак произведения путем сложения по модулю двух знаковых разрядов сомножителей.

**2.** Проверить множимое на равенство нулю: если равно нулю, операцию умножения следует прекратить, т.к. результат будет также равным нулю.

**3.** Проверить множитель на равенство нулю: если равен нулю, операцию умножения следует прекратить, т.к. результат будет также равным нулю.

**4.** Сложить характеристики сомножителей. При этом могут возникнуть следующие ситуации:

**- Переполнение разрядной сетки (ПРС).** Признаком ПРС является получение единицы переноса и единицы в старшем разряде результирующей характеристики. ПРС истинно, если хотя бы один из оставшихся разрядов не нуль. Если возникло, то необходимо зафиксировать его появление и прекратить операцию.

**- Временное ПРС.** Признаком временного ПРС является единица переноса из старшего разряда и единица в старшем разряде, остальная часть характеристики равно нулю. При этом нужно продолжить алгоритм умножения.

**- Потеря младших разрядов (ПМР).** Признаком ПМР является отсутствие единицы переноса и ноль в старшем разряде результирующей характеристики. Если возникла ситуация ПМР, то необходимо зафиксировать её появление и выдать нулевой результат.

В противном случае переходим к пункту 5.

**5.** Выполнить цикл умножения:

5.1 Проанализировать младшую цифру очередного разряда множителя.

5.2 Если цифра множителя «1», то суммировать множимое с накопленной суммой частичных произведений (СЧП). В результате суммирования может возникнуть ситуация временного ПРС в мантиссе, которая устраняется последующим сдвигом вправо.

5.3Выполнить сдвиги множителя и суммы частичных произведений на один разряд вправо.

**6.** Количество тактов равно количеству разрядов мантиссы множителя.

**7.** После цикла умножения проверить необходимость нормализации результата. Если произведение денормализовано, провести нормализацию результата: сдвинуть произведение на 1 разряд влево, вычесть “1” из характеристики. При этом, если ранее было зафиксировано временное ПРС, оно устраняется. Если после нормализации мантиссы произошло ПМР, нужно зафиксировать её появление и вывести результат равный «0»; в противном случае переходим к пункту 8.

Денормализация возможна лишь на один разряд, т.к. операнды поступают на входную шину уже нормализованными. Если результат нормализован, необходимо проверить, было ли зафиксировано временное ПРС. Если да, то установить признак ПРС и операцию необходимо прекратить.

**8.** Присвоить знак модулю произведения из п. 1 алгоритма. Если после нормализации результата зафиксирован признак ПМР, то в качестве знака результата необходимо выдать ноль.

# Численные примеры

1. Один из операндов равен нулю. В данном случае результат также равен нулю.

* Исходные данные:

С=5210

D=010

* Переведём числа в 2СС:

С=5210=32+16+4=25+24+22=1101002

D=010==00000002

Сложение знаков по модулю два : 0 0 = 0;

Так как множитель равен нулю, то произведение тоже равно нулю.

1. Штатная ситуация:

Исходные данные:

С=5210 = 0,8125\*26=0,1101\*26

D=6910=0,5390625\*27=0,1000101\*27

Изобразим числа С и D в разрядной сетке условной машины. Под мантиссы со знаком отведем 8 разрядов, под характеристики отведем 5 разрядов.

Изобразим числа С и D в разрядной сетке условной машины. Под мантиссы со знаком отведем 8 разрядов, под характеристики отведем 5 разрядов. Результат изображен в таблице 1.

Таблица 1 – Знаки, мантиссы и характеристики для штатной ситуации

|  |  |  |
| --- | --- | --- |
| Знак мантиссы | Мантисса | Характеристика |
| 0 | 1101000 | 10110 |
| 0 | 1000101 | 10111 |

1. Знак результата
2. Определим характеристику произведения сложением характеристик сомножителей в таблице 2:

Таблица 2 – Нахождение характеристики произведения

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| СА | 0 | 1 | 0 | 1 | 1 | 0 |
| СВ | 0 | 1 | 0 | 1 | 1 | 1 |
| СА+СВ=Cc | 1 | 0 | 1 | 1 | 0 | 1 |

ПРС характеристик не произошло!

Перемножим мантиссы 1 способом. Результат умножения представлен в таблице 3.

Таблица 3 – Умножение операндов без возникновения исключительных ситуаций

|  |  |  |  |
| --- | --- | --- | --- |
| **Множитель** | **Множимое** | **СЧП** | **Комментарий** |
| 0,1101000 | 0,1000101 | 0,0000000 0000000 | Исх.данные |
| 0,1101000 | 0,1000101 | 0,0000000 0000000 | - |
| 0,**0**110100 | 0,1000101 | 0,0000000 0000000 | Сдвиги |
| 0,**0**110100 | 0,1000101 | 0,0000000 0000000 | - |
| 0,**00**11010 | 0,1000101 | 0,0000000 0000000 | Сдвиги |
| 0,**00**11010 | 0,1000101 | 0,0000000 0000000 | - |
| 0,**000**1101 | 0,1000101 | 0,0000000 0000000 | Сдвиги |
| 0,**000**1101 | 0,1000101 | 0,0000000 0000000  0,0100010 1000000  0,0100010 1000000 | Сложение |
| 0,**0000**110 | 0,1000101 | 0,0010001 0100000 | Сдвиги |
| 0,**0000**110 | 0,1000101 | 0,0010001 0100000 | - |
| 0,**00000**11 | 0,1000101 | 0,0001000 1010000 | Сдвиги |
| 0,**00000**11 | 0,1000101 | 0,0001000 1010000  0,0100010 1000000  0,0101011 0010000 | Сложение |
| 0,**000000**1 | 0,1000101 | 0,0010101 1001000 | Сдвиги |
| 0,**000000**1 | 0,1000101 | 0,0010101 1001000 0,0100010 1000000 0,0111000 0001000 | Сложение |
| 0,**0000000** | 0,1000101 | 0,0111000 0001000 | Сдвиги |
| 0,**0000000** | 0,1000101 | 0,0111000 0001000 | Результат! |

Нормализуем мантиссу путем сдвига влево на 1 разряд, характеристику уменьшим на 1:

0,0111000 0001000 - > 0,1110000 0010000

Cc=Cc-1 Cc=101100

|  |  |  |
| --- | --- | --- |
| Знак мантиссы | Мантисса | Характеристика |
| 0 | 1110000001 | 101100 |

C\*D = 0,1110000001\*212 = =1110000001002=211+210+29+22=2048+1024+512+4=3588

Проверка результата:

C\*D=52\*69=3588

**3.** Возникновение ПРС:

ca = 1100 = +4

cb = 1101 = +5

Определим характеристику произведения сложением характеристик сомножителей в таблице 4:

Таблица 4 – Определение характеристики произведения для ситуации с ПРС

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| РА |  | 1 | 1 | 0 | 0 |
| РВ |  | 1 | 1 | 0 | 1 |
| РА+РВ | 1 | 1 | 0 | 0 | 1 |

Произошло ПРС характеристик!

Процесс умножения останавливается.

**4.** Возникновение ПМР при сложении характеристик:

РА=0010= -6; РВ=0101= -3;

Найдем характеристику произведения путем сложения характеристик сомножителей в таблице 5.

Таблица 5 – Определение характеристики произведения для ситуации с ПМР

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| РА | Перенос | 0 | 0 | 1 | 0 |
| РВ | 0 | 0 | 1 | 0 | 1 |
| РА+РВ | 0 | 0 | 1 | 1 | 1 |

Признак неустранимого ПМР, выводим нулевой результат.

**5.** Возникновение временного ПРС, переходящего в устранимое:

Исходные данные:

ma = 0,5 = 0,10000000 ca = 11100 = +12

mb = 0,5 = 0,10000000 cb = 10100 = +4

Определим характеристику произведения сложением характеристик сомножителей:

Таблица 6 – Определение характеристики произведения для ситуации с временным ПРС

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| РА |  | 1 | 1 | 1 | 0 | 0 |
| РВ |  | 1 | 0 | 1 | 0 | 0 |
| РА+РВ | 1 | 1 | 0 | 0 | 0 | 0 |

Произошло временное ПРС характеристик!

Алгоритм умножения можно продолжить.

Перемножим мантиссы 1 способом. Произведение представлено в таблице 7.

Таблица 7 – Умножение операндов с устранимым временным ПРС

|  |  |  |  |
| --- | --- | --- | --- |
| **Множитель** | **Множимое** | **СЧП** | **Комментарий** |
| 0,10000000 | 0,10000000 | 0,00000000 00000000 | Исх. данные |
| 0,10000000 | 0,10000000 | 0,00000000 00000000 | - |
| 0,**0**1000000 | 0,10000000 | 0,00000000 00000000 | Сдвиги |
| 0,**0**1000000 | 0,10000000 | 0,00000000 00000000 | - |
| 0,**00**100000 | 0,10000000 | 0,00000000 00000000 | Сдвиги |
| 0,**00**100000 | 0,10000000 | 0,00000000 00000000 | - |
| 0,**000**10000 | 0,10000000 | 0,00000000 00000000 | Сдвиги |
| 0,**000**10000 | 0,10000000 | 0,00000000 00000000 | - |
| 0,**0000**1000 | 0,10000000 | 0,00000000 00000000 | Сдвиги |
| 0,**0000**1000 | 0,10000000 | 0,00000000 00000000 | - |
| 0,**00000**100 | 0,10000000 | 0,00000000 00000000 | Сдвиги |
| 0,**00000**100 | 0,10000000 | 0,00000000 00000000 | - |
| 0,**000000**10 | 0,10000000 | 0,00000000 00000000 | Сдвиги |
| 0,**000000**10 | 0,10000000 | 0,00000000 00000000 | - |
| 0,**0000000**1 | 0,10000000 | 0,00000000 00000000 | Сдвиги |
| 0,**0000000**1 | 0,10000000 | 0,00000000 00000000  0,01000000 00000000  0,01000000 00000000 | Сложение |
| 0,**00000000** | 0,10000000 | 0,01000000 00000000 | Сдвиги |
| 0,**00000000** | 0,10000000 | 0,01000000 00000000 | Результат! |

Для нормализации мантиссы сдвинем ее влево, характеристику уменьшим на 1:

0,01000000 00000000 0,10000000 00000000

Cc = Cc – 1

Нормализуем характеристику произведения в таблице 8.

Таблица 8 – Определение характеристики произведения при нормализации

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| РА | 1 | 1 | 0 | 0 | 0 | 0 |
| РВ | 1 | 1 | 1 | 1 | 1 | 1 |
| РА+РВ | 1 | 0 | 1 | 1 | 1 | 1 |

Временное ПРС устранено.

**6.** Возникновение временного ПРС, переходящего в неустранимое:

Исходные данные:

ma = 0,75 = 0,11000000 ca = 11100 = +12

mb = 0,5 = 0,10000000 cb = 10100 = +4

Определим характеристику произведения сложением характеристик сомножителей в таблице 9:

Таблица 9 – Определение характеристики произведения при неустранимом временном ПРС

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| РА |  | 1 | 1 | 1 | 0 | 0 |
| РВ |  | 1 | 0 | 1 | 0 | 0 |
| РА+РВ | 1 | 1 | 0 | 0 | 0 | 0 |

Произошло временное ПРС характеристик! Алгоритм умножения можно продолжить. Перемножим мантиссы 1 способом. Результат в таблице 10.

Таблица 10 – Умножение операндов с неустранимым временным ПРС

|  |  |  |  |
| --- | --- | --- | --- |
| **Множитель** | **Множимое** | **СЧП** | **Комментарий** |
| 0,11000000 | 0,11100000 | 0,00000000 00000000 | Исх. данные |
| 0,11000000 | 0,11100000 | 0,00000000 00000000 | - |
| 0,**0**1100000 | 0,11100000 | 0,00000000 00000000 | Сдвиги |
| 0,**0**1100000 | 0,11100000 | 0,00000000 00000000 | - |
| 0,**00**110000 | 0,11100000 | 0,00000000 00000000 | Сдвиги |
| 0,**00**110000 | 0,11100000 | 0,00000000 00000000 | - |
| 0,**000**11000 | 0,11100000 | 0,00000000 00000000 | Сдвиги |
| 0,**000**11000 | 0,11100000 | 0,00000000 00000000 | - |
| 0,**0000**1100 | 0,11100000 | 0,00000000 00000000 | Сдвиги |
| 0,**0000**1100 | 0,11100000 | 0,00000000 00000000 | - |
| 0,**00000**110 | 0,11100000 | 0,00000000 00000000 | Сдвиги |
| 0,**00000**110 | 0,11100000 | 0,00000000 00000000 | - |
| 0,**000000**11 | 0,11100000 | 0,00000000 00000000 | Сдвиги |
| 0,**000000**11 | 0,11100000 | 0,00000000 00000000  0,01110000 00000000  0,01110000 00000000 | Сложение |
| 0,**0000000**1 | 0,11100000 | 0,00111000 00000000 | Сдвиги |
| 0,**0000000**1 | 0,11100000 | 0,00111000 00000000  0,01110000 00000000  0,10101000 00000000 | Сложение |
| 0,**00000000** | 0,11100000 | 0,10101000 00000000 | Сдвиги |
| 0,**00000000** | 0,11100000 | 0,10101000 00000000 | Результат! |

В ходе умножения мантисс временная ПРС стала неустранимой ПРС, выдаем ошибку результата.

**7.** ПМР при нормализации:

ma=0,5=0,10000000 ca = 0011 = -3

mb=0,75=0,11000000 cb = 0101 = -5

Знак результата

Определим характеристику произведения сложением характеристик сомножителей в таблице 11:

Таблица 11 – Определение характеристики при ПМР после нормализации

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| РА |  | 0 | 0 | 1 | 1 |
| РВ |  | 0 | 1 | 0 | 1 |
| РА+РВ | 0 | 1 | 0 | 0 | 0 |

Временное ПРС, доведем умножение до конца. Результат представлен в таблице 12.

Таблица 12 – Умножение операндов с появлением ПМР после нормализации

|  |  |  |  |
| --- | --- | --- | --- |
| **Множитель** | **Множимое** | **СЧП** | **Комментарий** |
| 0,10000000 | 0,11000000 | 0,00000000 00000000 | Исх.данные |
| 0,10000000 | 0,11000000 | 0,00000000 00000000 | - |
| 0,**0**1000000 | 0,11000000 | 0,00000000 00000000 | Сдвиги |
| 0,**0**1000000 | 0,11000000 | 0,00000000 00000000 | - |
| 0,**00**100000 | 0,11000000 | 0,00000000 00000000 | Сдвиги |
| 0,**00**100000 | 0,11000000 | 0,00000000 00000000 | - |
| 0,**000**10000 | 0,11000000 | 0,00000000 00000000 | Сдвиги |
| 0,**000**10000 | 0,11000000 | 0,00000000 00000000 | - |
| 0,**0000**1000 | 0,11000000 | 0,00000000 00000000 | Сдвиги |
| 0,**0000**1000 | 0,11000000 | 0,00000000 00000000 | - |
| 0,**00000**100 | 0,11000000 | 0,00000000 00000000 | Сдвиги |
| 0,**00000**100 | 0,11000000 | 0,00000000 00000000 | - |
| 0,**000000**10 | 0,11000000 | 0,00000000 00000000 | Сдвиги |
| 0,**000000**10 | 0,11000000 | 0,00000000 00000000 | - |
| 0,**0000000**1 | 0,11000000 | 0,00000000 00000000 | Сдвиги |
| 0,**0000000**1 | 0,11000000 | 0,00000000 00000000  0,01100000 00000000  0,01100000 00000000 | Сложение |
| 0,**00000000** | 0,11000000 | 0,01100000 00000000 | Сдвиги |
| 0,**00000000** | 0,11000000 | 0,01100000 00000000 | Результат! |

В ходе умножения получился ненормализованный результат, следует нормализовать результат сдвигом на один разряд влево и вычесть единицу из характеристики. Результат нормализации представлен в таблице 13.

mc = 0,01100000 00000000 - > 0,11000000 00000000

Cc = Cc – 1

Таблица 13 – Нормализация характеристики произведения в ситуации с появлением ПМР

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| РА | 0 | 1 | 0 | 0 | 0 |
| РВ | 1 | 1 | 1 | 1 | 1 |
| РА+РВ | 0 | 0 | 1 | 1 | 1 |

В ходе нормализации мантиссы появилось ПМР, выводим нулевой результат.

# Разработка функциональной схемы операционной части устройства

## Состав операционного автомата

Операционный автомат должен содержать следующие элементы:

* 24–х разрядный сдвиговый регистр RG1 для хранения мантиссы множителя со знаком;
* 23–х разрядный не сдвиговый регистр RG2 для хранения мантиссы множимого со знаком;
* 23–х разрядный сдвиговый регистр RG3 для хранения суммы частичных произведений (СЧП);
* 23–х разрядный сумматор SM1 для сложения множимого с СЧП;
* D–триггер T1 для хранения знака характеристики множителя;
* RS–триггер T2 для хранения и выдачи сигнала ПРС;
* 8–и разрядный не сдвиговый регистр RG4 для хранения характеристик;
* 8–и разрядный счетчик СТ1 для хранения характеристики результата;
* 7–и разрядный счетчик циклов умножения CT2;
* 8–и разрядный сумматор SM2 для сложения характеристик;
* 7–и разрядный элемент «ИЛИ» для определения признака ПРС;
* Двухвходовой элемент «ИЛИ» для определения признака ПМР;
* D–триггер T3для запоминания единицы переноса при перемножении мантисс;
* Двухвходовой элемент «И» для определения временного ПРС;
* Двухвходовой элемент «И» для определения истинного ПРС;
* Двухвходовой элемент «сложение по модулю два» для определения знака результата;
* Усилитель–формирователь для выдачи результата на выходную шину;
* Инвертор для формирования знака характеристики.

## Описание работы операционного автомата

Операнды разрядностью 32 байта поступают в операционный автомат (ОА) в ПК по входной шине. Первый операнд - множимое. Запись мантиссы множимого производится в RG1 и RG2. Знак числа записывается в Т1 и в 23-й разряд RG1. 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 «001001». Вторым операндом приходит множитель. Запись знака и мантиссы множителя 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 цикле умножения производится анализ младшего разряда множителя, если он равен единице, то в RG3 заносим сумму множимого и частичной суммы. После этого производится сдвиг RG1 и RG3 вправо, а значение СТ2 увеличивается на 1. Цикл умножения заканчивается, когда в старшем разряде СТ2 появится «1». Если старший разряд RG3 равен нулю, 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, RG1 и счетчик CT1 и выдаем на выходную шину ноль.

Для выдачи результата на выходную шину содержимое RG3, 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, T1, установка Т2 в положение «0», обнуление RG3 и CT1, запись СТ2;

y2 – запись в СТ1 значения выхода SM2;

y3 – сдвиг RG3 влево, СТ1:=СТ1-1;

y4 – запись в RG3 значения выхода SM1, запись в Т2 значения выхода переноса SM1;

y5 – сдвиг RG1 и RG3 вправо, СТ2:=СТ2+1;

y6 – обнуление Т1 и RG1;

y7 – установка Т2 в положение «1»;

y8 – выдача результата на выходную шину.

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

Х – проверка наличия операндов на входной шине;

р1 – проверка на ноль;

р2 – проверка на временное ПРС;

р3 – младший разряд RG1 (проверка очередной цифры множителя);

р4 – старший разряд RG3 (проверка нормализации результата);

р5 – проверка на ПМР;

р6 – проверка на окончание операции умножения;

р7 – проверка на ПРС;

Z – проверка возможности выдачи результата на шину выхода.

Таким образом, УА должен вырабатывать 9 управляющих сигналов и посылать их в ОА в нужные такты машинного времени в соответствии с алгоритмом выполнения операции умножения, учитывая 8 осведомительных сигналов, поступающих из ОА.

Функциональная схема (ФС) ОА изображена на рисунке 2 и представлена в приложении А.

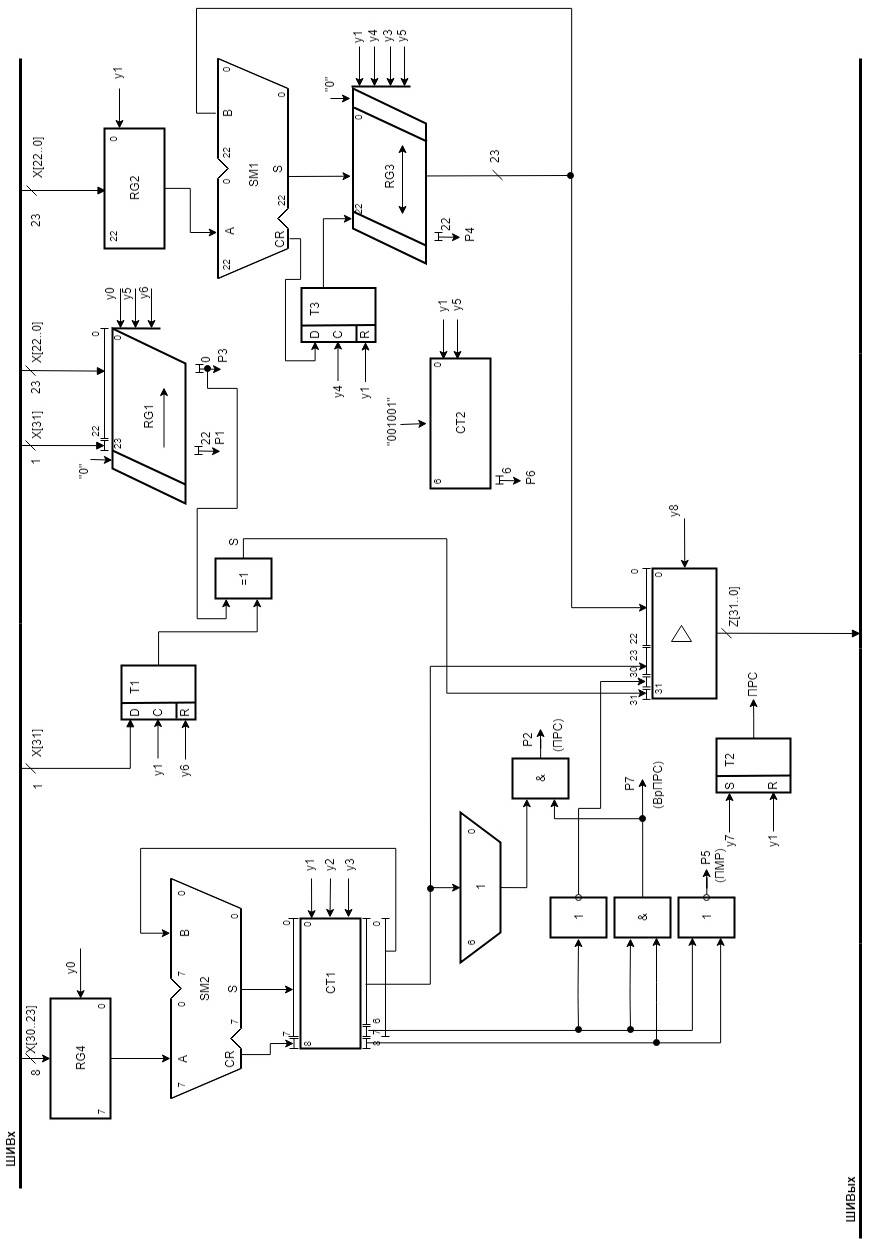


Рисунок 2 - Функциональная схема операционного автомата

# Разработка содержательной граф-схемы алгоритма

В первом такте производится проверка наличия на входной шине множимого (блок 1). При поступлении множимого, его мантисса заносится в RG2 и RG1, знак заносится в Т1, в RG4 заносится значение характеристики, RG3, CT1 и Т2 обнуляются, а в СТ2 заносится значение «001001» (блок 2).

Затем производится проверка на ноль мантиссы множимого (блок 3). Если Р1=1, то выполняется обнуление СТ1, RG3, Т1, RG1 (блок 19) и переход к блоку 21, иначе в СТ1 записывается значение выхода SM2. (блок 4). Производится проверка наличия на входной шине множителя (блок 5). При поступлении множителя, знак и мантисса заносятся в RG1, характеристика записывается в RG4 (блок 6). Производится проверка мантиссы на ноль (блок 7). Если P1=1, осуществляется переход к блоку 19, иначе в СТ1 заносится значение выхода сумматора SM2 (блок 8). Производится проверка на ПРС (блок 9). Если P2=1, то триггер Т2 устанавливаем в единицу (Блок 20) и переходим к блоку 21, иначе происходит проверка на ПМР (блок 10), если Р5=1 то осуществляется переход к блоку 19, иначе начинается цикл умножения. Производится проверка младшего разряда регистра множителя RG1 (блок 11). Если Р3=0, то осуществляются сдвиги на 1 разряд вправо RG1 и RG3, а также увеличение значения СТ2 на единицу (блок 13), иначе в RG3 заносится результат суммы значений регистра множимого и регистра частичных сумм (блок 12) и осуществляется переход к блоку 13. Далее проверяется условие окончания цикла умножения (блок 14). Если Р6=0, то осуществляется переход к блоку 11, иначе заканчивается цикл умножения и проверяется условие нормализации мантиссы (блок 15). Если Р4=1, то выполняется проверка на ПРС (блок 17), если Р2=1 то переход к блоку 20, иначе проверка условия ПМР (блок 18); если Р4=0 то выполняется нормализация (блок 16) и переход к блоку 18. Если Р5=1, то переход к блоку 19, иначе выполняем переход к блоку 21. Выполняется проверка возможности выдачи результата на выходную шину (блок 21) и выдача результата на выходную шину (блок 22)

Содержательная граф–схема алгоритма представлена на рисунке 3 и в приложении Б.

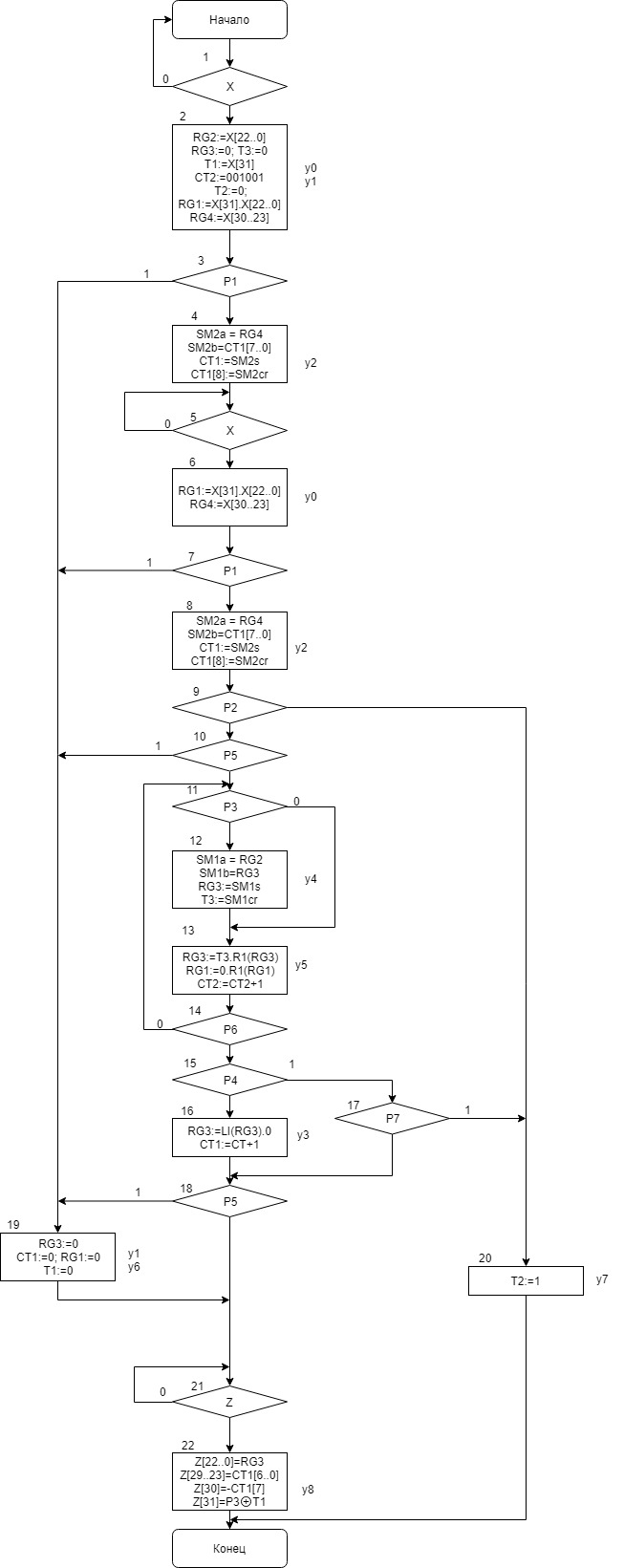


Рисунок 3 – ГСА алгоритма умножения

# 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 14. Каждой условной вершине содержательной ГСА ставится в соответствие один из входных сигналов управляющего автомата X1…Xm.

Таблица 14 – Список микрокоманд

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Y0 | Y1 | Y2 | Y3 | Y4 | Y5 | Y6 | Y7 | Y8 | Y9 |
| y0, y1 | y2, | y0 | y2 | y4 | y5 | y3 | y1,y6 | y7 | 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 15). Во всех операторных вершинах ГСА проставляют символы из множества выходных сигналов УА – У1, У2, …, УN (таблица 14). Удобно в каждой операторной вершине ГСА вслед за символом МК указать в скобках набор МО, образующих каждую МК.

Таблица 15 – Список входных сигналов для УА

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Х1 | Х2 | Х3 | Х4 | Х5 | Х6 | Х7 | Х8 | Х9 |
| Х | P1 | P2 | P3 | P4 | P5 | P6 | P7 | 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..a8 , для модели Мура символами b0..b12.

Отмеченная граф–схема алгоритма представлена на рисунке 4 и в приложении В.

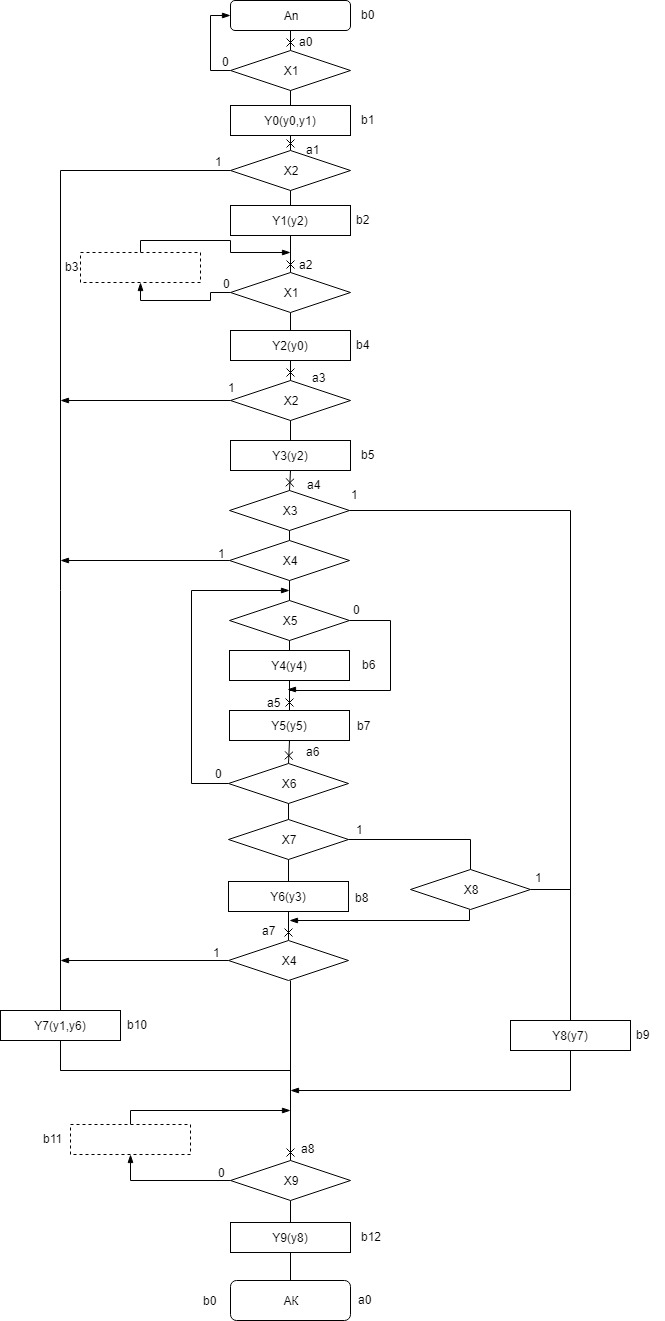


Рисунок 4 - Отмеченная ГСА для алгоритма умножения

# Построение графов автоматов моделей Мили и Мура

Граф автомата модели Мили имеет 9 вершин, соответствующих состояниям автомата а0…a8. Дуги его отмечены входными сигналами X1…X9, действующими на каждом переходе, и набором выходных сигналов y0…y8, вырабатываемых управляющим автоматом на данном переходе. Граф автомата модели Мили представлен на рисунке 5 и в приложении Г.

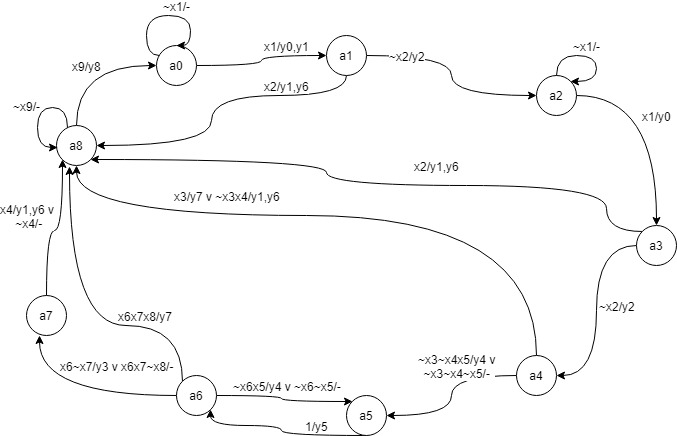


Рисунок 5 - Граф автомата модели Мили

Раскрытие пустых переходов позволяет нам сохранить время работы автомата. Пустой переход из а4 в а5 раскрывать не имеет смысла, так как это не приносит нам выгоды – мы сохраняем только один такт ( анализ младшего разряда - складывать множимое с СЧП или нет). Раскрытие этого перехода приведет к увеличению аппаратурных затрат за счет появления нестандартного перехода.

Переход из а5 в а6 находится в цикле, раскрыв его мы сэкономим половину 23 тактов (около 12), поэтому следует его раскрыть. Также из а6 в а8 уже есть переход, поэтому мы можем раскрыть пустой переход из а6 в а7 без аппаратурных затрат. Минимизированный граф автомата модели Мили представлен на рисунке 6 и в приложении Д.

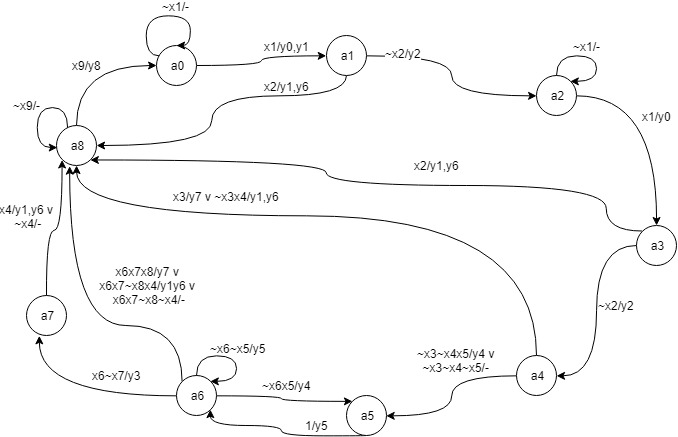


Рисунок 6 – Минимизированный граф автомата модели Мили

Граф автомата Мура имеет 13 вершин, соответствующих состояниям автомата b0…b12. Каждое состояние определяет наборы выходных сигналов y0…y8 управляющего автомата, а дуги графа отмечены входными сигналами X1…X9, действующими на данном переходе. Граф автомата модели Мура представлен на рисунке 7 и в приложении Е.

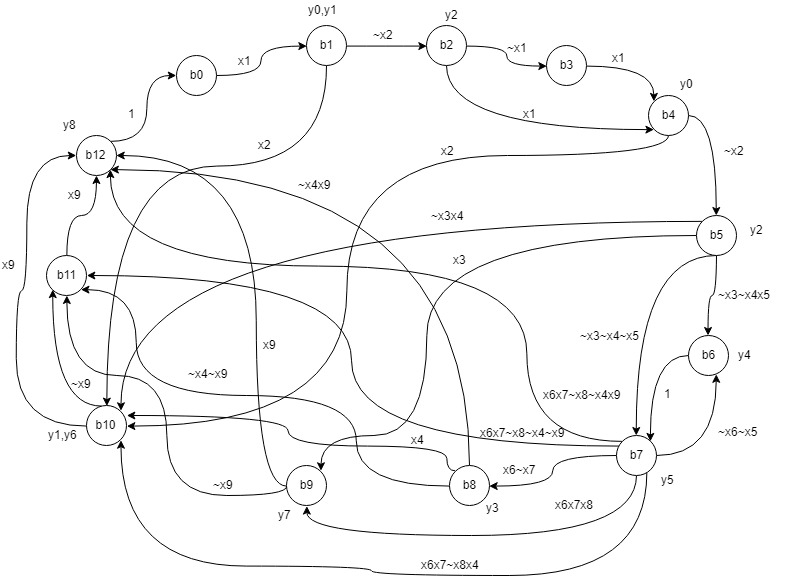


Рисунок 7 - Граф автомата модели Мура

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

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.

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 9 состояний a0...a8 графа автомата по модели Мили 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 16.

Таблица 16 - Коды состояний для модели Мили на D–триггерах

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Состояние | a0 | a1 | a2 | a3 | a4 | a5 | a6 | a7 | a8 |
| Переходы | a0,a8 | а0 | a1,a2 | a2 | a3 | a4,a6 | a5,а6 | a6 | a1,a3,a4,a6,a7,a8 |
| Число  переходов | 2 | 1 | 2 | 1 | 1 | 2 | 2 | 1 | 6 |
| Код | 0001 | 0011 | 0010 | 0101 | 0110 | 0100 | 1000 | 1100 | 0000 |

Далее составляется прямая структурная таблица переходов и выходов автомата по модели Мили, результаты которой представлены в таблице 17, и формируются логические выражения для функций возбуждения.

Таблица 17 - 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  возбуж-  дения  D  триггера |
| a0 | 0001 | a0 | 0001 | ~x1 | - | D0 |
| a1 | 0011 | x1 | y0y1 | D1D0 |
| a1 | 0011 | a2 | 0010 | ~x2 | y2 | D1 |
| а8 | 0000 | x2 | у1у6 | - |
| a2 | 0010 | a2 | 0010 | ~x1 | - | D1 |
| a3 | 0101 | x1 | y0 | D2D0 |
| a3 | 0101 | a4 | 0110 | ~x2 | y2 | D2D1 |
| a8 | 0000 | x2 | y1y6 | - |
| a4 | 0110 | a5 | 0100 | ~x3~x4x5 | y4 | D2 |
| a5 | 0100 | ~x3~x4~x5 | - | D2 |
| a8 | 0000 | ~x3x4 | y1y6 | - |
| а8 | 0000 | x3 | y7 | - |
| a5 | 0100 | a6 | 1000 | 1 | y5 | D3 |
| a6 | 1000 | a5 | 0100 | ~x6x5 | y4 | D2 |
| a6 | 1000 | ~x6~x5 | y5 | D3 |
| a7 | 1100 | x6~x7 | y3 | D3D2 |
| а7 | 1100 | x6x7~x8 | - | D3D2 |
| a8 | 0000 | x6x7x8 | y7 | - |
| a7 | 1100 | a8 | 0000 | ~x4 | - | - |
| a8 | 0000 | x4 | y1y6 | - |
| a8 | 0000 | a0 | 0001 | x9 | y8 | D0 |
| a8 | 0000 | ~x9 | - | - |

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 = a0~x1 V a0x1 V a2x1 V a8x9

D1 = a0x1 V a1~x2 V a2~x1 V a3~x2

D2 = a2x1 V a3~x2 V a4~x3~x4x5 V a4~x3~x4~x5 V a6~x6x5 V a6x6~x7 V a6x6x7~x8

D3 = a5 V a6~x6~x5 V a6x6~x7 V a6x6x7~x8

Аналогично составляются логические выражения для функций выходов:

y0 = a0x1 V a2x1

y1 = a0x1 V a3x2 V a1x2 V a4~x3x4 V a7x4

y2 = a1~x2 V a3~x2

y3 = a6x6~x7

y4 = a4~x3~x4x5 V a6~x6x5

y5 = a5 V a6~x6~x5

y6 = a3x2 V a4~x3x4 V a7x4 V a1x2

y7 = a4x3 V a6x6x7x8

Выделяем общие части:

k = a0x1;

s = а2х1;

g = а1~x2;

h = a3~x2;

m = a3x2;

n = a6~x6x5;

l = a6~x6~x5;

o = a6x6x7~x8;

j = a6x6~x7;

c = a4~x3~x4x5;

e = a4~x3x4;

i = a7x4;

f = a5;

u = a1x2;

(2)

(2)

(2)

(2)

(2)

(3)

(3)

(4)

(3)

(4)

(3)

(2)

(0)

(2)

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

D0 = a0~x1 V k V s V a8x9

D1 = k V g V a2~x1 V h

D2 = s V h V c V a4~x3~x4~x5 V n V j V o

D3 = f V l V j V o

(8)

(6)

(11)

(4)

y0 = k V s

y1 = k V m V u V e V i

y2 = g V h

y3 = a6x6~x7

y4 = c V n

y5 = f V l

y6 = m V e V I V u

y7 = a4x3 V a6x6x7x8

(2)

(5)

(2)

(3)

(2)

(2)

(4)

(8)

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

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

∑ = КС + ИНВ + ЭП + НУ + 𝐷𝐶 = 91+8+16+0+4=119

Схема начальной установки (НУ) для D–триггеров в качестве ЭП приведена на рисунке 8, где D0, D1, D2, D3 – функции возбуждения соответствующих ЭП.

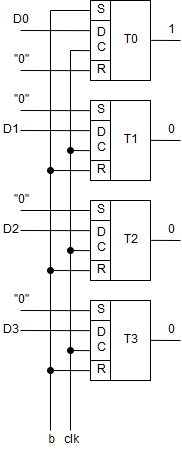


Рисунок 8 - Схема формирования начальной установки на D–триггерах

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

Для кодирования 9 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. Строим матрицу, состоящую из различных пар номеров таких, что в автомате S есть переход abk -> alk

М =

1. Закодируем состояние первой строки: kb1 = 00..00; kl1 = 00..01
2. Вычеркиваем из матрицы М первую строку. Получим матрицу М’
3. В начальной (верхней) строке матрицы М’ один элемент уже закодирован. Выберем незакодированный элемент первой строки матрицы и обозначим его – q
4. Построим матрицу Мq, выбрав из M’ все строки, содержащие элемент q
5. Пусть множество Bq = {q1, … , qF} – множество всех элементов матрицы Мq, которые уже закодированы. Для каждого кода kq найдем множество кодов С1q, соседних с кодом kq и еще не занятых для кодирования состояний автомата. Построим множество всех возможных кодов, соседних с kq и еще незакодированных:

D1q = 1q

Если нет ни одного множества с незакодированными элементами, то количество разрядов для кодирования (кол-во ЭП) выбрано неправильно

1. Находим Wgf = |kqi ⊕ kqf| - кодовое расстояние для пар переходов («сколько триггеров переключается»)
2. Находим сумму всех кодовых расстояний Wg = gf
3. Выбираем код для состояния, у которого сумма кодовых расстояний Wg – минимальна
4. Из матрицы М’ вычеркиваем строки, в которых оба элемента закодированы, получаем матрицу М’’, если матрица М’’ – пустая, переходим к пункту 11, иначе 4.
5. Вычисляем W = ms , сумму всех кодовых расстояний. Оценкой качества кодирования рассмотренного алгоритма может служить число К, где p – число переходов данного автомата. Чем меньше К, тем ближе полученное кодирование к соседнему: K =

Эксперименты показали, что К при хорошем кодировании лежит в пределах 1.4 ≤ К ≤ 2.1

Выпишем матрицу Т – матрицу всех возможных переходов автомата.

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 5 |
| 1 | 2 | 5 |
| 1 | 8 | 9 |
| 2 | 3 | 5 |
| 3 | 4 | 6 |
| 3 | 8 | 9 |
| 4 | 5 | 6 |
| 4 | 8 | 9 |
| 5 | 6 | 7 |
| 6 | 5 | 7 |
| 6 | 7 | 6 |
| 6 | 8 | 10 |
| 7 | 8 | 8 |
| 8 | 0 | 8 |

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

|T|=

2) Упорядочим строки матрицы |𝑇|, для чего строим матрицу |𝑀|

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

*|M|=*

3) Закодируем первые 2 состояния:

**𝑎6=0001; 𝑎8=0000;**

4)

**a4=1000**

Эффективность кодирования: 𝑘 = 19/14 = 1,36;

Получившиеся коды состояний представлены в таблице 18.

Таблица 18 - Коды состояний для модели Мили на RS–триггерах

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| а0 | а1 | а2 | а3 | а4 | а5 | а6 | а7 | а8 |
| 0110 | 0010 | 1010 | 0100 | 1000 | 1001 | 0001 | 0101 | 0000 |

Далее составляется прямая структурная таблица переходов и выходов автомата модели Мили, представленная в таблице 19, и по известному правилу формируются логические выражения для функций возбуждения. Прямая структурная таблица переходов и выходов автомата модели Мили на RS–триггерах.

Таблица 19 - 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  возбуж-  дения  RS  триггера |
| a0 | 0110 | a0 | 0110 | ~x1 | - | - |
| a1 | 0010 | x1 | y0y1 | R2 |
| a1 | 0010 | a2 | 1010 | ~x2 | y2 | S3 |
| а8 | 0000 | х2 | у1у6 | R1 |
| a2 | 1010 | a2 | 1010 | ~x1 | - | - |
| a3 | 0100 | x1 | y0 | R3S2R1 |
| a3 | 0100 | a4 | 1000 | ~x2 | y2 | S3R2 |
| a8 | 0000 | x2 | y1y6 | R2 |
| a4 | 1000 | a5 | 1001 | ~x3~x4x5 | y4 | S0 |
| a5 | 1001 | ~x3~x4~x5 | - | S0 |
| a8 | 0000 | ~x3x4 | y1y6 | R3 |
| а8 | 0000 | x3 | y7 | R3 |
| a5 | 1001 | a6 | 0001 | 1 | y5 | R3 |
| a6 | 0001 | a5 | 1001 | ~x6x5 | y4 | S3 |
| a6 | 0001 | ~x6~x5 | y5 | - |
| a7 | 0101 | x6~x7 | y3 | S2 |
| а7 | 0101 | x6x7~x8 | - | S2 |
| a8 | 0000 | x6x7x8 | y7 | R0 |
| a7 | 0101 | a8 | 0000 | ~x4 | - | R2R0 |
| a8 | 0000 | x4 | y1y6 | R2R0 |
| a8 | 0000 | a0 | 0110 | x9 | y8 | S2S1 |
| a8 | 0000 | ~x9 | - | - |

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 = a4~x3~x4x5 V a4~x3~x4~x5

S1 = a8x9

S2 = a8x9 V a6x6~x7 V a6x6x7~x8 V a2x1

S3 = a1~x2 V a3~x2 V a6~x6x5

R0 = a6x6x7x8 V a7~x4 V a7x4

R1 = a2x1 V a1x2

R2 = a0x1 V a3~x2 V a3x2 V a7~x4 V a7x4

R3 = a2x1 V a4~x3x4 V a4x3 V a5

Аналогично составляются логические выражения для функций выходов:

y0 = a0x1 V a2x1

y1 = a0x1 V a3x2 V a4~x3x4 V a7x4 V a1x2

y2 = a1~x2 V a3~x2

y3 = a6x6~x7

y4 = a4~x3~x4x5 V a6~x6x5

y5 = a5 V a6~x6~x5

y6 = a3x2 V a4~x3x4 V a7x4 V a1x2

y7 = a4x3 V a6x6x7x8

Выделяем общие части:

k = a0x1;

s = а2х1;

g = а1~x2;

h = a3~x2;

m = a3x2;

n = a6~x6x5;

l = a6x6x7x8;

o = a8x9;

j = a6x6~x7;

c = a4~x3~x4x5;

e = a4~x3x4;

i = a7x4;

f = a5;

d = a4x3;

z = a7~x4;

u = a1x2;

(2)

(2)

(2)

(2)

(2)

(3)

(4)

(2)

(3)

(4)

(3)

(2)

(0)

(2)

(2)

(2)

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

S0 = c V a4~x3~x4~x5

S1 = o

S2 = o V j V a6x6x7~x8 V s

S3 = g V h V n

R0 = l V z V i

R1 = s V u

R2 = k V h V m V z V i

R3 = s V e V d V f

(6)

(0)

(8)

(3)

(3)

(2)

(5)

(4)

y0 = k V s

y1 = k V m V e V i V u

y2 = g V h

y3 = j

y4 = c V n

y5 = f V a6~x6~x5

y6 = m V e V i V u

y7 = d V l

(2)

(5)

(2)

(0)

(2)

(5)

(4)

(2)

Инверторы (ИНВ): X̅2, X̅3, X̅4, X̅5, X̅6, X̅7, X̅8 (7)

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

∑ = КС + ИНВ + ЭП + НУ + 𝐷𝐶 =90+7+12+17+4 = 130;

Схема начальной установки для RS–триггеров в качестве ЭП приведена на рисунке 9, где S0, R0, S1, R1, S2, R2, S3, R3 – функции возбуждения соответствующих ЭП, а b – сигнал начальной установки.

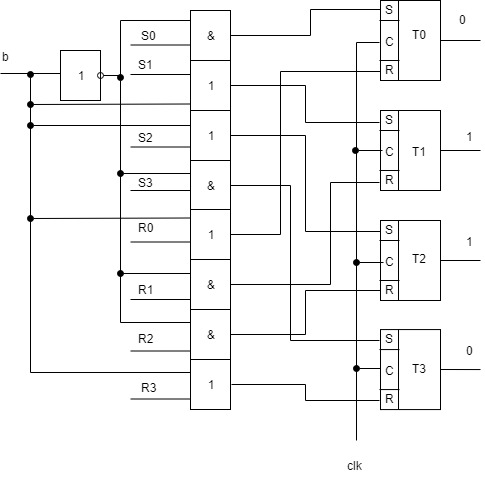


Рисунок 9 – Схема начальной установки для RS-триггеров

## 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», «R». Если при кодировании возникает ситуация, когда происходит переход в состояние отличное от предыдущего состояния более чем на единицу, необходимо заносить нестандартное состояние сразу же на элементы памяти счетчика. Для кодирования 9 состояний автомата по модели Мили потребуется 4–х разрядный счетчик. Получившиеся коды состояний представлены в таблице 20.

Таблица 20 - Коды состояний для модели Мили на счетчике

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

Далее составляется прямая структурная таблица переходов и выходов автомата Мили на счетчике, результаты которой представлены в таблице 21, и по известному правилу формируются логические выражения для функций возбуждения.

Таблица 21 - 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  возбуж-  дения |
| a0 | 0001 | a0 | 0001 | ~x1 | - | - |
| a1 | 0010 | x1 | y0y1 | +1 |
| a1 | 0010 | a2 | 0011 | ~x2 | y2 | +1 |
| a8 | 0000 | x2 | y1y6 | R |
| a2 | 0011 | a2 | 0011 | ~x1 | - | - |
| a3 | 0100 | x1 | y0 | +1 |
| a3 | 0100 | a4 | 0101 | ~x2 | y2 | +1 |
| a8 | 0000 | x2 | y1y6 | R |
| a4 | 0101 | a5 | 0110 | ~x3~x4x5 | y4 | +1 |
| a5 | 0110 | ~x3~x4~x5 | - | +1 |
| a8 | 0000 | ~x3x4 | y1y6 | R |
| а8 | 0000 | x3 | y7 | R |
| a5 | 0110 | a6 | 0111 | 1 | y5 | +1 |
| a6 | 0111 | a5 | 0110 | ~x6x5 | y4 | -1 |
| a6 | 0111 | ~x6~x5 | y5 | - |
| a7 | 1000 | x6~x7 | y3 | +1 |
| а8 | 0000 | x6x7~x8x4 | y1y6 | R |
| a8 | 0000 | x6x7x8 | y7 | R |
| a8 | 0000 | x6x7~x8~x4 | - | R |
| a7 | 1000 | a8 | 0000 | ~x4 | - | R |
| a8 | 0000 | x4 | y1y6 | R |
| a8 | 0000 | a0 | 0001 | x9 | y8 | +1 |
| a8 | 0000 | ~x9 | - | - |

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 a1~x2 V a2x1 V a3~x2 V a4~x3~x4x5 V a4~x3~x4~x5 V a5 V

a6 x6~x7 V a8x9

R = a3x2 V a4~x3x4 V a4x3 V a6x6x7x8 V a7~x4 V a7x4 V a1x2 V a6x6x7~x8x4 V a6x6x7~x8~x4

-1 = a6~x6x5

Аналогично составляются логические выражения для функций выходов:

y0 = a0x1 V a2x1

y1 = a0x1 V a3x2 V a1x2 V a4~x3x4 V a7x4 V a6x6x7~x8x4

y2 = a1~x2 V a3~x2

y3 = a6x6~x7

y4 = a4~x3~x4x5 V a6~x6x5

y5 = a5 V a6~x6~x5

y6 = a3x2 V a4~x3x4 V a7x4 V a1x2 V a6x6x7~x8x4

y7 = a4x3 V a6x6x7x8

Выделяем общие части:

k = a0x1;

s = а2х1;

g = а1~x2;

h = a3~x2;

m = a3x2;

n = a6~x6x5;

l = a6x6x7x8;

j = a6x6~x7;

c = a4~x3~x4x5;

e = a4~x3x4;

i = a7x4;

f = a5;

d = a4x3;

u = a1x2;

w = a8x9;

p = a6~x6~x5;

q = a6x6x7~x8x4;

(2)

(2)

(2)

(2)

(2)

(3)

(4)

(3)

(4)

(3)

(2)

(0)

(2)

(2)

(2)

(3)

(5)

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

+1 = k V g V s V h V c V a4~x3~x4~x5 V f V j V w

R = m V e V d V l V a7~x4 V i V u V q V a6x6x7~x8~x4

-1 = n

WR = b

Логическую функцию для счетного входа «+1» можно еще упростить.

Возьмем из таблицы те пункты, где не порождается никакой функции возбуждения, также возьмем сигнал R и -1 под знак инверсии.

В итоге получим:

+1 = ~(-1 V R V a0~x1 V a2~x1 V p V a8~x9) (13)

-1 = n (0)

R = m V e V d V l V a7~x4 V i V u V q V a6x6x7~x8~x4 (16)

WR = b (0)

y0 = k V s

y1 = k V m V u V e V i V q

y2 = g V h

y3 = j

y4 = c V n

y5 = f V p

y6 = m V e V i V u V q

y7 = d V l

y8 = w

(2)

(6)

(2)

(1)

(2)

(2)

(5)

(2)

(0)

Инверторы (ИНВ): X̅2, X̅3, X̅4, X̅5, X̅6, X7, X8 (7)

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

∑ = КС + ИНВ + ЭП + НУ + 𝐷𝐶 =94+7+9+0+4=114;

Схема начальной установки для счетчика в качестве ЭП приведена на рисунке 10, где b – сигнал начальной установки.

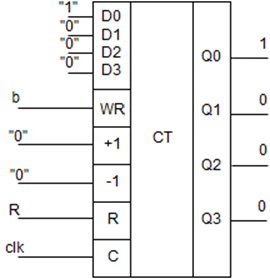


Рисунок 10 - Схема формирования начальной установки на счетчике

# 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 22.

Таблица 22 - Коды состояний для модели Мура на D–триггерах

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Исходное  состояние | b0 | b1 | b2 | b3 | b4 | b5 | b6 | b7 | b8 | b9 | b10 | b11 | b12 |
| Состояния перехода | b12 | b0 | b1 | b2 | b2  b3 | b4 | b5  b7 | b5  b6 | b7 | b5  b7 | b1  b4  b5  b7  b8 | b7  b8  b9  b10 | b7  b8  b9  b10  b11 |
| Число переходов | 1 | 1 | 1 | 1 | 2 | 1 | 2 | 2 | 1 | 2 | 5 | 4 | 5 |
| Код состояния | 0110 | 1001 | 1010 | 1100 | 0100 | 0111 | 1000 | 0011 | 1011 | 0101 | 0001 | 0010 | 0000 |

Далее составляется прямая структурная таблица переходов и выходов автомата модели Мура на D–триггерах, результаты которой представлены в таблице 23 и формируются логические выражения для функций возбуждения.

Таблица 23 - 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 | 0110 | - | b1 | 1001 | x1 | D3D0 |
| b1 | 1001 | y0y1 | b2 | 1010 | ~x2 | D3D1 |
| b10 | 0001 | x2 | D0 |
| b2 | 1010 | y2 | b3 | 1100 | ~x1 | D3D2 |
| b4 | 0100 | x1 | D2 |
| b3 | 1100 | - | b4 | 0100 | x1 | D2 |
| b4 | 0100 | y0 | b5 | 0111 | ~x2 | D2D1D0 |
| b10 | 0001 | x2 | D0 |
| b5 | 0111 | y2 | b6 | 1000 | ~x3~x4x5 | D3 |
| b7 | 0011 | ~x3~x4~x5 | D1D0 |
| b10 | 0001 | ~x3x4 | D0 |
| b9 | 0101 | x3 | D2D0 |
| b6 | 1000 | y4 | b7 | 0011 | 1 | D1D0 |
| b7 | 0011 | y5 | b6 | 1000 | ~x6~x5 | D3 |
| b8 | 1011 | x6~x7 | D3D1D0 |
| b12 | 0000 | x6x7~x8~x4x9 | - |
| b11 | 0010 | x6x7~x8~x4~x9 | D1 |
| b10 | 0001 | x6x7~x8x4 | D0 |
| b9 | 0101 | x6x7x8 | D2D0 |
| b8 | 1011 | y3 | b12 | 0000 | ~x4x9 | - |
| b11 | 0010 | ~x4~x9 | D1 |
| b10 | 0001 | x4 | D0 |
| b9 | 0101 | y7 | b12 | 0000 | x9 | - |
| b11 | 0010 | ~x9 | D1 |
| b10 | 0001 | y1y6 | b12 | 0000 | x9 | - |
| b11 | 0010 | ~x9 | D1 |
| b11 | 0010 | - | b12 | 0000 | x9 | - |
| b12 | 0000 | y8 | b0 | 0110 | 1 | D2D1 |

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 b4~x2 V b4x2 V b5~x3~x4~x5 V b5~x3x4 V b5x3 V b6 V b7x6~x7 V b7x6x7~x8x4 V b7x6x7x8 V b8x4

D1 = b1~x2 V b4~x2 V b5~x3~x4~x5 V b6 V b7x6~x7 V b7x6x7~x8~x4~x9 V b8~x4~x9 V b12 V b10~x9 V b9~x9

D2 = b2~x1 V b2x1 V b3x1 V b4~x2 V b5x3 V b7x6x7x8 V b12

D3 = b0x1 V b1~x2 V b2~x1 V b5~x3~x4x5 V b7~x6~x5 V b7x6~x7

Аналогично составляются логические выражения для функций выходов:

y0 = b1 V b4

y1 = b1 V b10

y2 = b2 V b5

y3 = b8

y4 = b6

y5 = b7

y6 = b10

y7 = b9

y8 = b12

Выделяем общие части:

k = b0x1

s = b1~x2

g = b2~x1

h = b4~x2

m = b5~x3~x4~x5

n = b6

l = b7x6~x7

j = b12

c = b7x6x7x8

(2)

(2)

(2)

(2)

(4)

(0)

(3)

(0)

(4)

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

y0 = b1 V b4

y1 = b1 V b10

y2 = b2 V b5

y3 = b8

y4 = n

y5 = b7

y6 = b10

y7 = b9

y8 = j

(2)

(2)

(2)

(0)

(0)

(0)

(0)

(0)

(0)

D0 = k V b1x2 V h V b4x2 V m V b5~x3x4 V b5x3 V n V l V b7x6x7~x8x4 V c V b8x4 (28)

D1 = s V h V m V n V l V b7x6x7~x8~x4~x9 V b8~x4~x9 V j V b10~x9 V b9~x9 (23)

D2 = g V b2x1 V b3x1 V h V b5x3 V c V j (13)

D3 = k V s V g V b5~x3~x4x5 V b7~x6~x5 V l (13)

Инверторы: X̅1, X̅2, X̅3, X̅4, X̅5, X̅6, X̅7, X̅8, X̅9 (9)

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

∑ = КС + ИНВ + ЭП + НУ + 𝐷𝐶 =102+9+16+0+4=131;

Схема формирования начальной установки на D–триггерах представлена на рисунке 11.

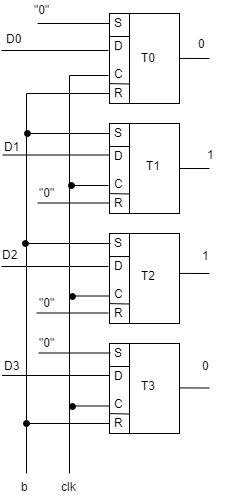


Рисунок 11 - Схема формирования начальной установки на D–триггерах автомата Мура

Цена по Квайну автомата модели Мура на D–триггерах получилась больше, 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.

# Построение схемы управляющего микропрограммного автомата

Наиболее оптимальной ценой по Квайну, равной 114, и быстродействием обладает модель автомата модели Мили на счетчике поэтому микропрограммный автомат будет строиться для этой модели. В приложении Ж приведена функциональная схема проектируемого управляющего автомата, выполняющего операцию умножения чисел в двоичной системе счисления с плавающей запятой в прямом коде первым способом с характеристиками. 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

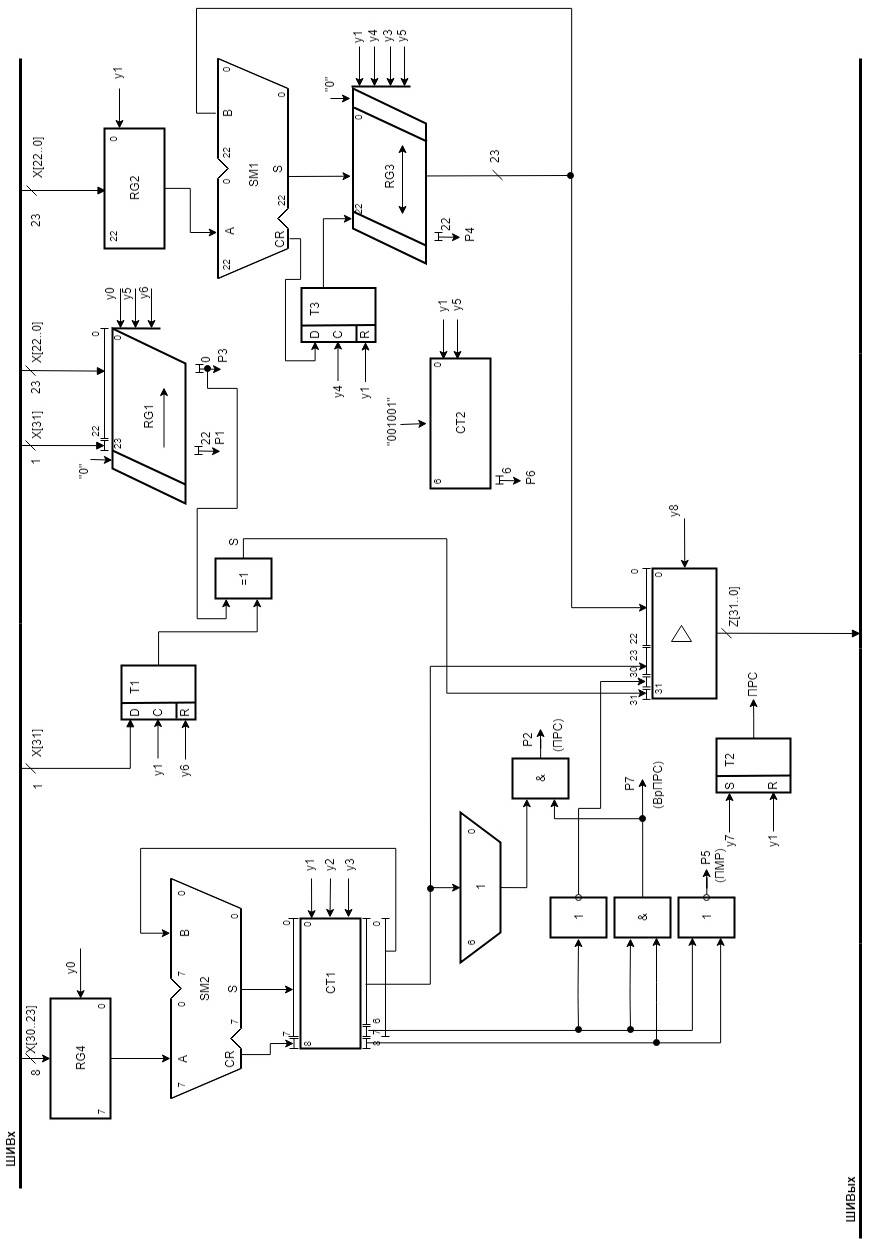
В ходе курсового проекта был синтезирован автомат, выполняющий умножение первым способом в двоичной системе счисления с плавающей запятой с использованием прямого кода при умножении мантисс. Управляющий автомат был синтезирован по модели Мили с использованием 4–х разрядного счетчика в качестве элемента памяти, так как цена по Квайну, равная 114, и быстродействие данного автомата получились наиболее оптимальными. Автомат, полученный в ходе выполнения курсового проекта, задан множеством внутренних состояний a0–a8, множеством входных сигналов X1–X9, множеством выходных сигналов y0–y8, функциями переходов и выходов, заданными в таблице 21, начальным состоянием a0.

# Приложение А

(обязательное)

**Функциональная схема операционного автомата**

Рисунок 12 - Функциональная схема операционного автомата

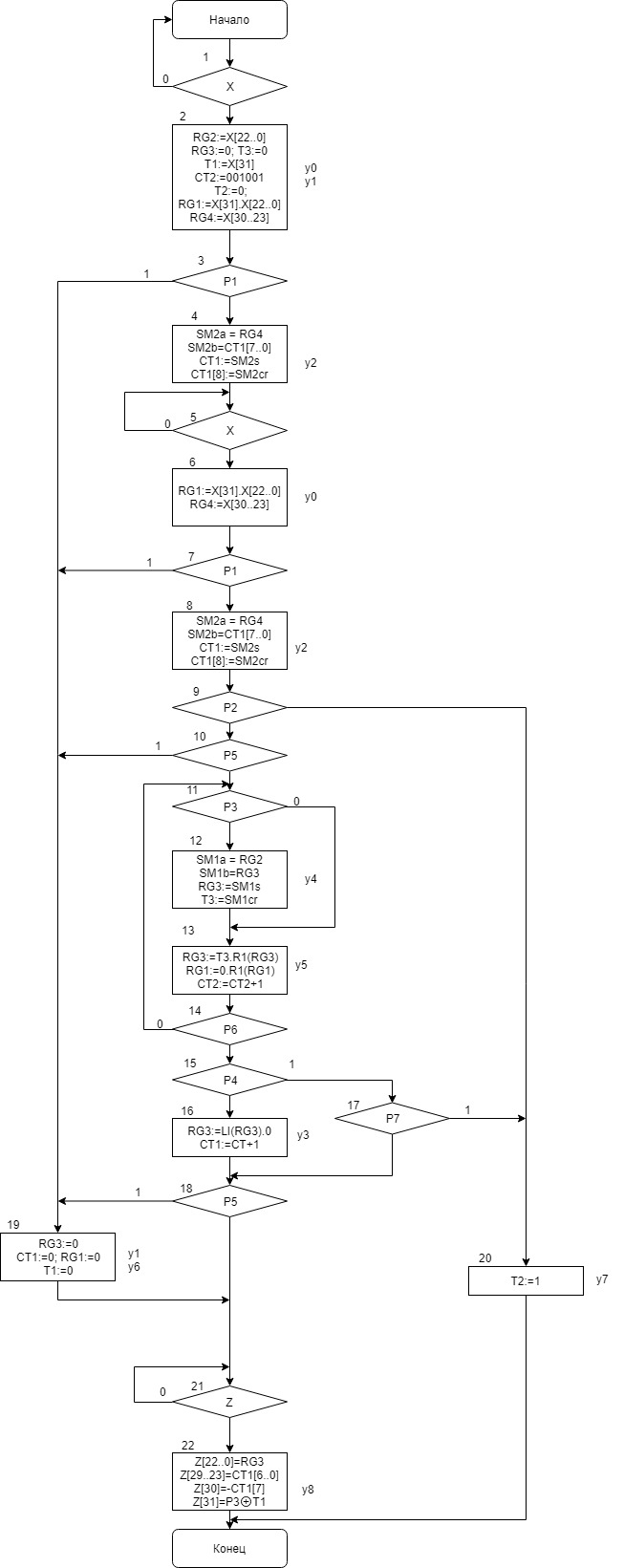


# Приложение Б

(обязательное)

**Содержательная граф – схема алгоритма**

Рисунок 13 – ГСА алгоритма умножения

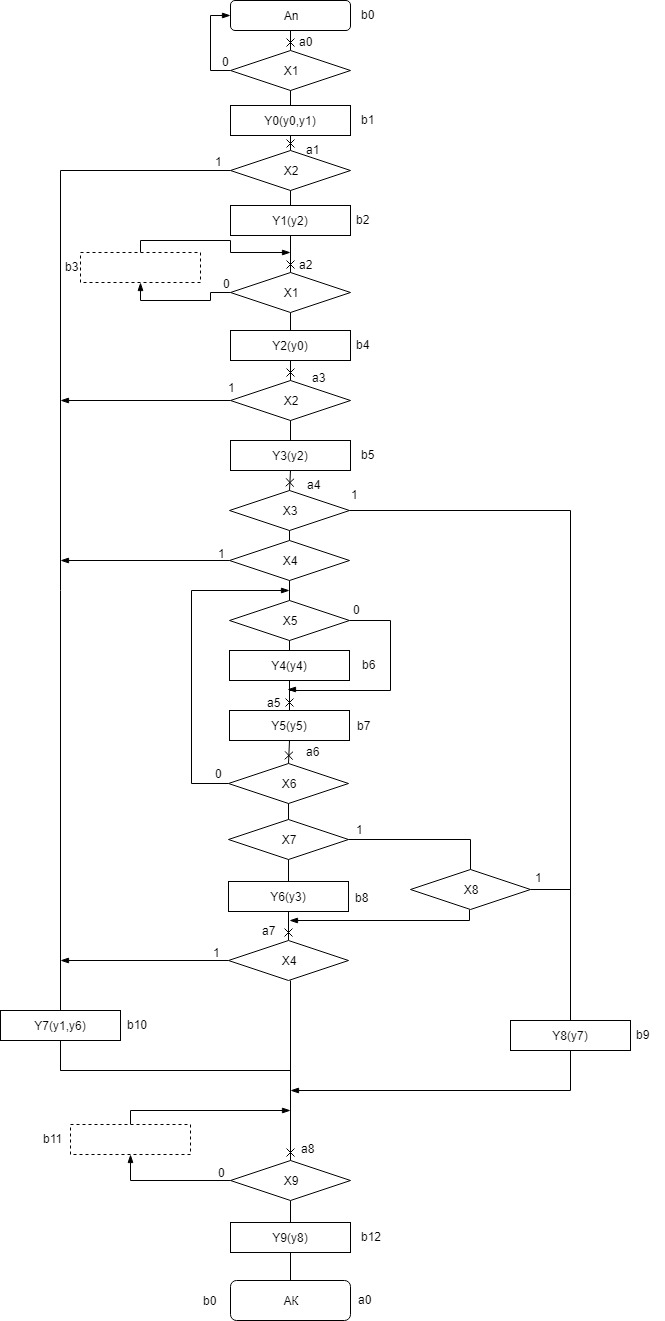


# Приложение В

(обязательное)

**Отмеченная граф – схема алгоритма**

Рисунок 14 - Отмеченная ГСА для алгоритма умножения



# Приложение Г

(обязательное)

**Граф автомата модели Мили**

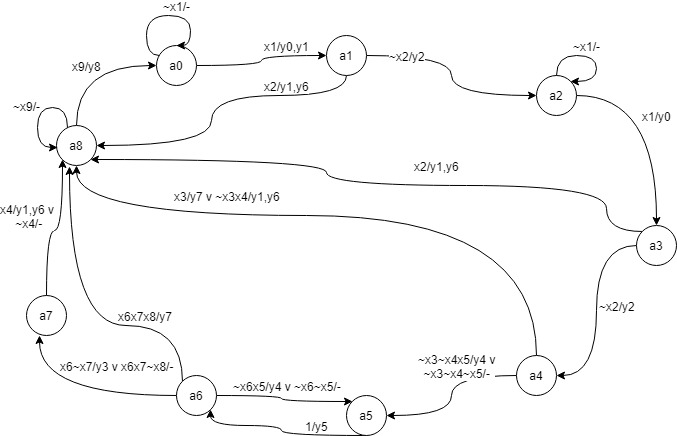


Рисунок 15– Граф автомата модели Мили

Рисунок 13 - Граф автомата модели Мили

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

# Приложение Д

(обязательное)

**Минимизированный граф автомата модели Мили**

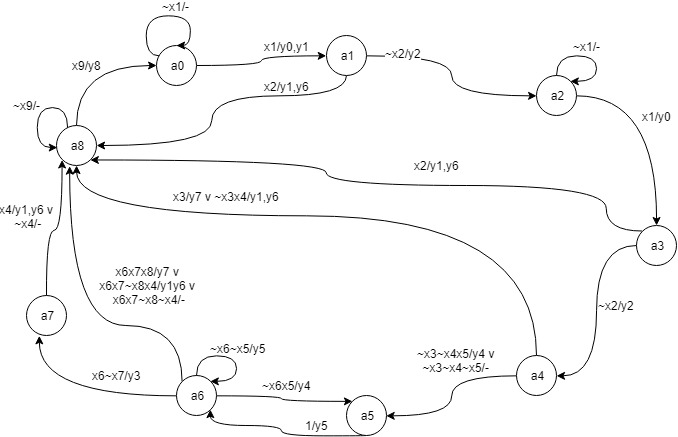


Рисунок 16 – Минимизированный граф автомата модели Мили

# Приложение Е

(обязательное)

**Граф автомата модели Мура**

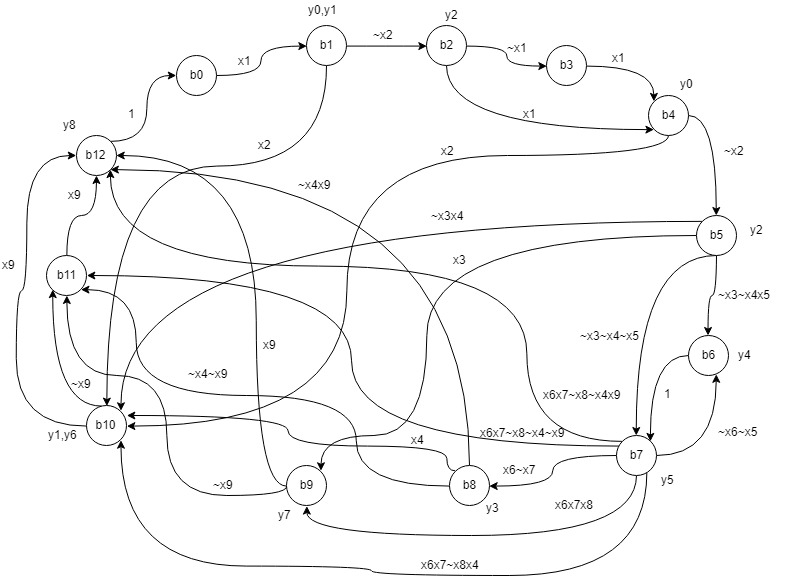


Рисунок 17 – Граф автомата модели Мура

Рисунок 15 - Граф автомата модели Мура

# Приложение Ж

(обязательное)

**Функциональная схема управляющего автомата**

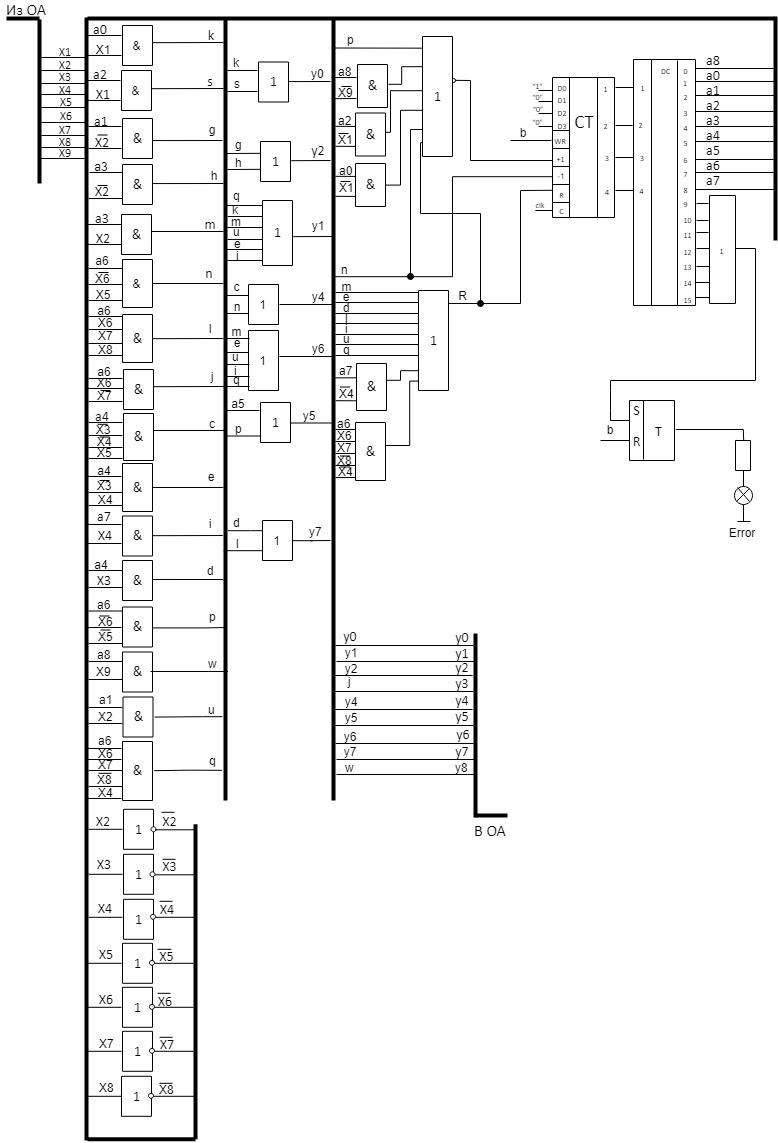


Рисунок 18 - Функциональная схема управляющего автомата

# 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. 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*