

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Ильшат Ринатович Мухаметзянов

Должность: директор

Дата подписания: 14.07.2023 09:36:08

Уникальный программный ключ:

aba80b84033c9ef19b188e7ea0434f90a83a40954ba270e84b5e6402d1d8d0

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение высшего образования «Казанский национальный исследовательский технический университет

им. А.Н. Туполева-КАИ»

(КНИТУ-КАИ)

Чистопольский филиал «Восток»

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ЛАБОРАТОРНЫМ РАБОТАМ**

по дисциплине

**ТЕОРИЯ АВТОМАТОВ**

Индекс по учебному плану: **Б1.В.08**

Направление подготовки: **09.03.01 Информатика и вычислительная техника**

Квалификация: **Бакалавр**

Профиль подготовки: **Автоматизированные системы обработки информации и управления**

Типы задач профессиональной деятельности: **проектный,  
производственно-технологический**

Рекомендовано УМК ЧФ КНИТУ-КАИ

Чистополь  
2023 г.

# Лабораторная работа №1

## *СИНТЕЗ КОМБИНАЦИОННЫХ СХЕМ НА УНИВЕРСАЛЬНЫХ ЛОГИЧЕСКИХ ЭЛЕМЕНТАХ*

Цель работы. Изучение методов логического синтеза комбинационных схем на элементах И-НЕ и ИЛИ-НЕ.

### **Комбинационные схемы**

В ЦВМ информация кодируется в двоичном коде и представляется в виде множества двоичных сигналов. Каждому из последних соответствует двоичная переменная, принимающая лишь два значения: 0 и 1. Отсюда следует, что любую схему в ЦВМ можно представить как функциональный преобразователь, в котором появление на входах какой-либо комбинации из нулей и единиц вызывает появление на выходах определенной комбинации из нулей и единиц. При этом выделяются два основных класса схем - комбинационные схемы (КС) и конечные автоматы (КА).

В КС значения выходных сигналов в момент времени  $t$  однозначно определяются значениями входных сигналов в тот же момент времени. Выходные сигналы в конечном автомате зависят также и от состояния автомата, которое, в свою очередь, зависит от входных сигналов, поступивших в предыдущие моменты времени.

Технические вопросы синтеза КС решаются с помощью аппарата алгебры логики, в котором основным понятием является понятие переключательной (или булевой) функции. Переключательной функцией (ПФ) называется функция, способная принимать лишь два значения, и такая, что все ее аргументы также могут принимать только два значения. Если значения ПФ отожде-

ствить с выходными сигналами схемы, а значения ее аргументов с входными сигналами, то функция будет описывать процесс преобразования электронной схемой входных сигналов в выходные. На этом основано приложение алгебры логики к синтезу КС.

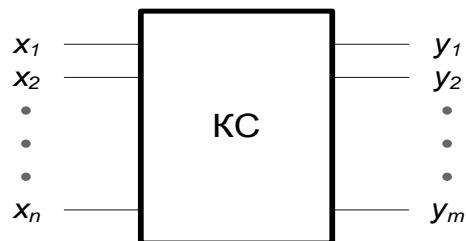


Рис.1.1

Общий вид КС представлен на рис.1.1. Можно сказать, что эта схема реализует  $m$  переключательных функций от  $n$  аргументов ( $y_i=f_i(x_1, x_2, \dots, x_n), i=1-m$ ).

Любая сколь угодно сложная КС строится из более простых схем с одним выходом, реализующих элементарные ПФ. Эти схемы, называемые логическими элементами, должны реализовывать базис, т.е. систему ПФ, из которых с помощью операций алгебры логики может быть образована ПФ любой сложности[1]. В цифровой технике наиболее широко используется булевый базис, включающий переключательные функции или операции: отрицания (операция НЕ), дизъюнкции (операция ИЛИ), конъюнкции (операция И), и универсальный базис, включающий либо операцию отрицания дизъюнкции (операция ИЛИ-НЕ), либо операцию отрицания конъюнкции (операция И-НЕ).

Операция отрицания реализуется элементом НЕ (или инвертором), значение сигнала на выходе которого равно обратному значению входного сигнала ( $y=\bar{x}$ ). Таблица истинности и условное обозначение элемента представлены на рис.1.2.

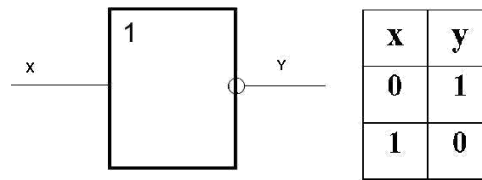


Рис.1.2

Операция дизъюнкции над значениями входных сигналов ( $y = x_1 \vee x_2 \vee \dots \vee x_n$ ,  $n \geq 2$ ) выполняется элементом ИЛИ. Сигнал на выходе элемента ИЛИ принимает значение 0 только в том случае, если ни один из входных сигналов не имеет в данный момент времени значения 1. Таблица истинности и условное обозначение элемента ИЛИ ( $n = 2$ ) приведены на рис.1.3

Операция конъюнкции ( $y = x_1 x_2 \dots x_n$ ,  $n \geq 2$ ) выполняется элементом И, сигнал на выходе которого равен 1, если все входные сигналы одновременно равны 1. На рис.1.4 приведены таблица истинности и условное обозначение элемента И ( $n = 2$ ).

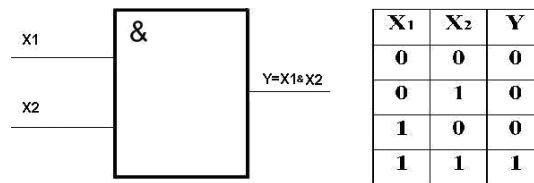


Рис.1.3

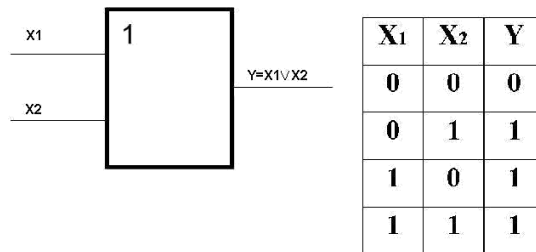
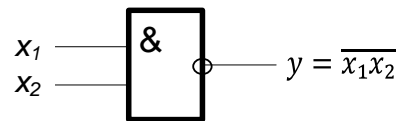


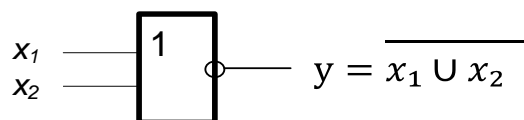
Рис.1.4

Операцию отрицания дизъюнкции  $y = \overline{x_1 \vee x_2 \vee x_3 \vee \dots \vee x_n} = \downarrow(x_1, x_2, \dots, x_n), n \geq 2$  реализует элемент ИЛИ-НЕ, представляющий собой элемент ИЛИ с инверсным выходом. Операцию отрицания конъюнкции ( $y = \overline{x_1 x_2 \dots x_n} = \uparrow(x_1, x_2, \dots, x_n), n \geq 2$ ) реализует элемент И-НЕ, являющийся элементом И с инверсным выходом. Таблицы истинности и условные обозначения элементов ИЛИ-НЕ и И-НЕ приведены на рис.1.5 и 1.6 соответственно.



X <sub>1</sub>	X <sub>2</sub>	Y
0	0	1
0	1	0
1	0	0
1	1	1

Рис.1.5



X <sub>1</sub>	X <sub>2</sub>	Y
0	0	1
0	1	1
1	0	0
1	1	0

Рис.1.6

По теореме де Моргана  $\overline{x_1 x_2 \dots x_n} = \overline{x_1} \vee \overline{x_2} \vee \dots \vee \overline{x_n}$  и  $\overline{x_1 \vee x_2 \vee \dots \vee x_n} = \overline{x_1} \overline{x_2} \dots \overline{x_n}$ . Отсюда следует, что элемент И-НЕ выполняет операцию ИЛИ над инверсными значениями входных сигналов (рис.6, в), а элемент

ИЛИ-НЕ операцию И над инверсными значениями входных сигналов (рис.1.5, в).

Введем далее важное понятие полярности логики, являющееся связующим звеном между реальным элементом и его логической функцией. Под полярностью логики понимается уровень сигнала (положительный или отрицательный, высокий или низкий), который соответствует логической единице. Уровни электрического потенциала, представляющие 1 и 0, не относятся к существенным характеристикам элементов, а выбираются перед началом логического синтеза. Так, например, элементы диодно-транзисторной (ДТЛ) и транзисторно-транзисторной логики (ТТЛ) выполняют операцию И-НЕ только в том случае, если в качестве уровня единицы принят более высокий потенциал (положительная логика). Если же выбрать в качестве сигнала единицы низкий потенциал (отрицательная логика), то те же элементы будут реализовывать операцию ИЛИ-НЕ. Пусть, например, имеется элемент, выполняющий в положительной логике операцию И-НЕ  $y = \overline{x_1 x_2 \dots x_n}$ . При изменении полярности логики (в качестве сигнала 1 выбирается более низкий уровень потенциала) на входы элемента будут поступать инверсные (обратные) значения ( $\overline{x_1 x_2 \dots x_n}$ ) переменных  $x_1, x_2, x_n$ , а с выхода будет снижаться обратное значение  $\overline{y}$  функции  $y$ . Так как функция элемента, определяемая его структурой, сохраняется, то справедлива следующая подстановка в выражение  $y = \overline{x_1 x_2 \dots x_n} : y = \overline{\overline{x_1, x_2, \dots, x_n}}$ . Преобразуя последнее выражение в следующем порядке:  $y = \overline{\overline{x_1 x_2 \dots x_n}} = \overline{x_1 \vee x_2 \vee \dots \vee x_n}$ , убеждаемся, что элемент реализует операцию ИЛИ-НЕ в отношении переменных  $x_1, x_2, \dots, x_n$ .

Характеристика комбинационных схем. На рис.1.7 представлена КС, реализующая на элементах булевого базиса систему двух ПФ:  $y_1 = x_1 x_2 \vee \bar{x}_2 x_3$ ;  $y_2 = (x_1 \vee x_2 \vee x_3) \bar{y}_1$ . Как видно, в КС значения входных сигналов определяются путем последовательного преобразования входных сигналов в промежуточные и промежуточных - в выходные, т.е. путем многоуровневого преобразования.

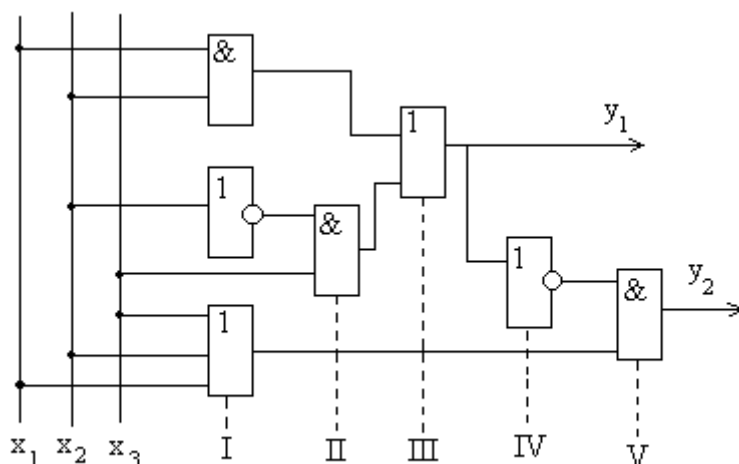


Рис.1.7

При определении уровней КС используется правило: каждый элемент  $i$ -го уровня ( $i > I$ ) должен иметь хотя бы один вход, подключенный к выходу элемента  $(i - 1)$ -го уровня. Нулевой уровень составляют входы КС, первый уровень - элементы, на все входы которых поступают сигналы непосредственно с входных проводов КС. Обратная связь, т.е. подключение выхода элемента какого-либо уровня ко входу элемента того же или младшего уровня, не допускается. Таким образом, число уровней  $r$  в КС равно максимальному числу элементов, проходя через которые сигнал от входа КС достигает ее выхода. Если на любом элементе сигнал задерживается на время  $\tau_0$ , то значение  $D = r \tau_0$  будет определять быстродействие КС.

Сложность КС характеризуется суммарным числом  $S$  входов элементов, составляющих схему. В схеме рис.1.7  $r = 5$ ,  $S =$

13. Важными характеристиками системы элементов, используемых при построении КС, являются коэффициент объединения  $J$  и коэффициент разветвления  $F$ . Коэффициент объединения  $J$  задает максимальное число входов элемента, т.е. максимальное число элементов, выходы которых могут быть объединены через входы данного. Одной из мер, позволяющих удовлетворить требуемое значение  $J$  в синтезируемой КС, является разделение входов с помощью дополнительных элементов. Процесс разделения входов при  $J=2$  поясняется схемами на рис.1.8, а (для случая реализации в булевом базисе функции  $y_1 = x_1 x_2 x_3 x_4$ ) и 1.8, б (для случая реализации в базисе И-НЕ функции  $y_2 = \overline{x_1 x_2 x_3 x_4}$ ). В первом случае используется преобразование:  $y_1 = x_1 x_2 x_3 x_4 = (x_1 x_2)(x_3 x_4)$ , и вместо элемента И с четырьмя входами в КС вводятся 3 элемента И с двумя входами. Во втором случае вместо одного элемента И-НЕ

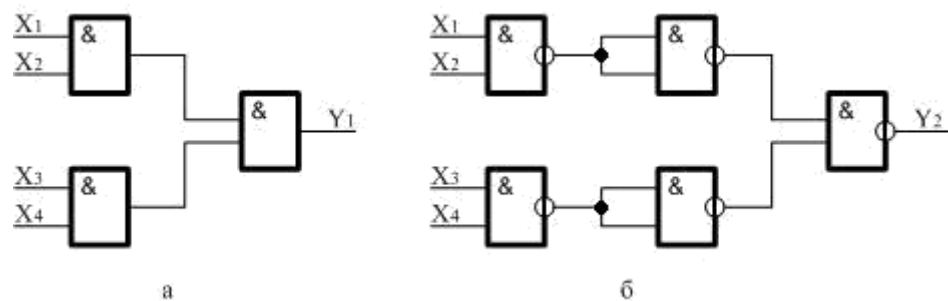


Рис.8

с четырьмя входами применены 5 элементов И-НЕ с двумя входами. Здесь

$$y_2 = \overline{x_1 x_2 x_3 x_4} = \overline{x_1 x_2} \vee \overline{x_3 x_4} = \overline{x_1 x_2} \vee \overline{x_3 x_4} = \overline{x_1 x_2} \cdot \overline{x_3 x_4} = \overline{(x_1 x_2)(x_3 x_4)}$$

Коэффициент разветвления задает максимальное число входов элементов, которые можно соединить с выходом данного элемента, не вызывая искажений сигналов 0 и 1, превышающих за-



данные пределы. Если в КС оказался перегруженным какой-либо элемент, то принимаются меры к его разгрузке.

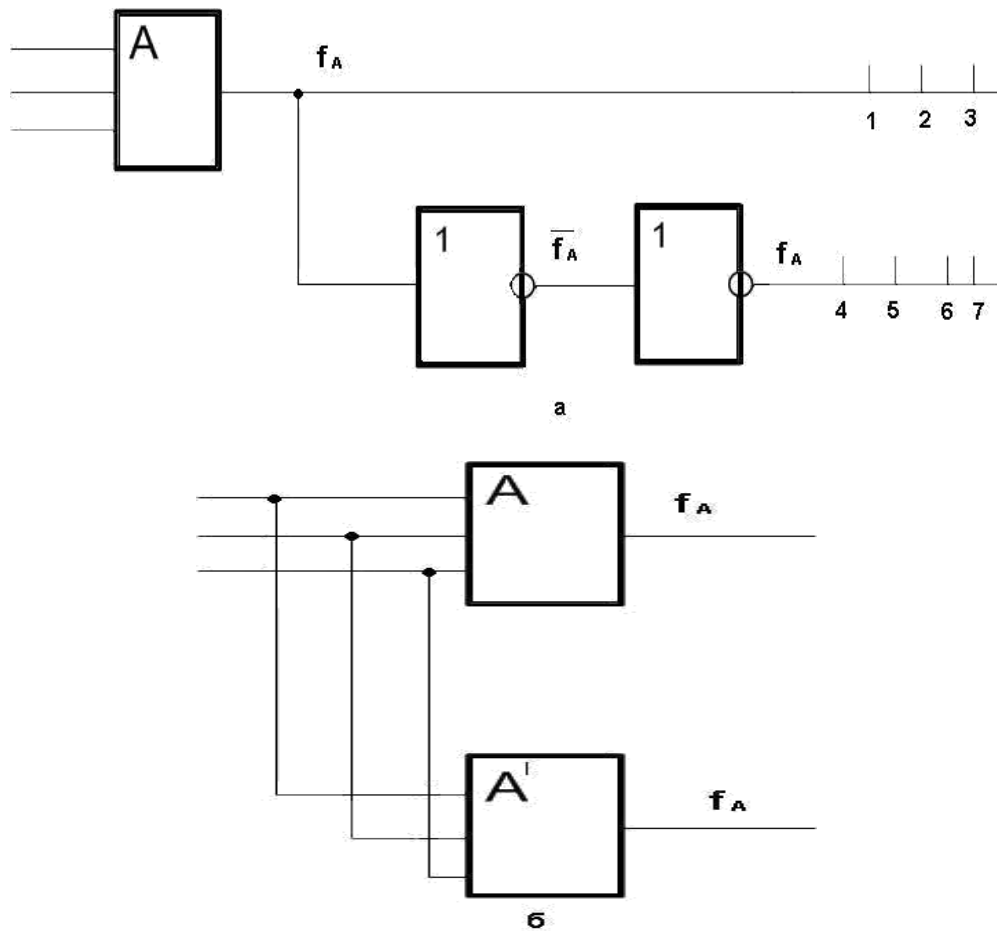


Рис.1.9

Разгрузка может выполняться дублированием выходного сигнала или дублированием элемента. Пусть  $F=4$ , и к выходу некоторого элемента  $A$  необходимо подключить 7 входов других элементов. Дублирование сигнала осуществляется по схеме рис.1.9,а. с помощью двух инверторов  $I$  и  $II$ . Недостатком способа является увеличение задержки сигналов в схеме. От этого недостатка свободен способ разгрузки дублированием элемента (рис.1.9.б).

Синтез КС на элементах универсального базиса сводится к выполнению следующих этапов: представление реализуемых ПФ в булевом базисе; минимизации ПФ; перевод минимальных

выражений в базис И-НЕ либо ИЛИ-НЕ; построение КС с учетом требуемых значений  $J, F, D$ .

Синтез КС на элементах универсального базиса сводится к выполнению следующих этапов: представление реализуемых ПФ в булевом базисе; минимизации ПФ; перевод минимальных выражений в базис И-НЕ либо ИЛИ-НЕ; построение КС с учетом требуемых значений  $J, F, D$ .

### **Задания на лабораторную работу № 1**

1.  $F(A, B, C, D) = \vee (1, 3, 6, 7, 9, 12, 13) = 1, \vee (0, 4, 8, 14) = 0$   
На остальных наборах переключательная функция не определена

2.  $F(A, B, C, D) = \vee (0, 2, 4, 5, 8, 10, 15) = 1, \vee (1, 7, 9, 14) = 0$   
На остальных наборах переключательная функция не определена

3.  $F(A, B, C, D) = \vee (1, 2, 5, 7, 14, 15) = 1, \vee (0, 4, 8, 13) = 0$   
На остальных наборах переключательная функция не определена

4.  $F(A, B, C, D) = \vee (3, 4, 6, 7, 11, 12) = 1, \vee (1, 5, 8, 14) = 0$   
На остальных наборах переключательная функция не определена

5.  $F(A, B, C, D) = \vee (0, 3, 6, 8, 10, 14) = 1, \vee (1, 4, 11, 13) = 0$   
На остальных наборах переключательная функция не определена

6.  $F(A, B, C, D) = \vee (0, 1, 7, 9, 11, 15) = 1, \vee (3, 5, 10, 14) = 0$   
На остальных наборах переключательная функция не определена

7.  $F(A, B, C, D) = \vee (2, 3, 6, 13, 14, 15) = 1, \vee (1, 7, 9, 12) = 0$   
На остальных наборах переключательная функция не определена

8.  $F(A, B, C, D) = \vee (5, 6, 8, 9, 11, 14) = 1, \vee (2, 7, 10, 13) = 0$   
На остальных наборах переключательная функция не определена

9.  $F(A, B, C, D) = \vee (4, 5, 7, 10, 12, 13) = 1, \vee (1, 6, 11, 14) = 0$   
На остальных наборах переключательная функция не определена

10.  $F(A, B, C, D) = \vee (0, 5, 8, 11, 13, 14) = 1, \vee (2, 7, 10, 15) = 0$   
На остальных наборах переключательная функция не определена

11.  $F(A, B, C, D) = \vee (1, 3, 4, 10, 11, 15) = 1, \vee (5, 8, 12, 13) = 0$   
На остальных наборах переключательная функция не определена

12.  $F(A, B, C, D) = \vee (0, 4, 6, 12, 13, 15) = 1, \vee (1, 3, 7, 14) = 0$   
На остальных наборах переключательная функция не определена

13.  $F(A, B, C, D) = \vee (1, 2, 6, 8, 10, 11, 14) = 1, \vee (0, 3, 9, 13) = 0$   
На остальных наборах переключательная функция не определена

14.  $F(A, B, C, D) = \vee (2, 4, 5, 7, 12, 13, 15) = 1, \vee (1, 6, 8, 14) = 0$   
На остальных наборах переключательная функция не определена

15.  $F(A, B, C, D) = \vee (1, 5, 6, 8, 10, 11, 14) = 1, \vee (0, 4, 9, 12) = 0$   
На остальных наборах переключательная функция не определена

16.  $F(A, B, C, D) = \vee (2, 5, 7, 8, 11, 13, 15) = 1, \vee (0, 6, 10, 14) = 0$   
На остальных наборах переключательная функция не определена

17.  $F(A, B, C, D) = \vee (1, 4, 8, 10, 11, 14, 15) = 1, \vee (2, 7, 9, 13) = 0$   
На остальных наборах переключательная функция не определена

18.  $F(A, B, C, D) = \vee (1, 3, 6, 7, 9, 12, 13) = 1, \vee (0, 4, 8, 14) = 0$   
На остальных наборах переключательная функция не определена

19.  $F(A, B, C, D) = \vee (0, 1, 3, 7, 12, 14, 15) = 1, \vee (2, 5, 11, 13) = 0$   
На остальных наборах переключательная функция не определена

20.  $F(A, B, C, D) = \vee (2, 3, 6, 8, 9, 11, 15) = 1, \vee (1, 4, 10, 14) = 0$   
На остальных наборах переключательная функция не определена

21.  $F(A, B, C, D) = \vee (2, 4, 7, 8, 11, 14) = 1, \vee (0, 5, 13) = 0$   
На остальных наборах переключательная функция не определена

22.  $F(A, B, C, D) = \vee (1, 5, 6, 7, 10, 13, 15) = 1, \vee (2, 4, 8, 14) = 0$   
На остальных наборах переключательная функция не определена

23.  $F(A, B, C, D) = \vee (0, 6, 7, 10, 11, 15) = 1, \vee (1, 8, 9, 14) = 0$   
На остальных наборах переключательная функция не определена

24.  $F(A, B, C, D) = \vee (2, 3, 7, 10, 12, 14) = 1, \vee (0, 8, 13) = 0$   
На остальных наборах переключательная функция не определена

25.  $F(A, B, C, D) = \vee (1, 6, 7, 9, 10, 14) = 1, \vee (0, 5, 12) = 0$   
На остальных наборах переключательная функция не определена

26.  $F(A, B, C, D) = \vee (1, 2, 8, 10, 11, 12) = 1, \vee (3, 4, 9, 14) = 0$   
На остальных наборах переключательная функция не определена

27.  $F(A, B, C, D) = \vee (0, 5, 7, 13, 14, 15) = 1, \vee (1, 4, 8, 12) = 0$   
 На остальных наборах переключательная функция не определена

28.  $F(A, B, C, D) = \vee (3, 5, 6, 10, 11, 12, 15) = 1, \vee (0, 2, 7, 14) = 0$   
 На остальных наборах переключательная функция не определена

29.  $F(A, B, C, D) = \vee (1, 6, 7, 9, 13, 14) = 1, \vee (0, 5, 12, 15) = 0$   
 На остальных наборах переключательная функция не определена

30.  $F(A, B, C, D) = \vee (2, 4, 11, 13, 14, 15) = 1, \vee (0, 5, 10, 12) = 0$   
 На остальных наборах переключательная функция не определена

### Задания на лабораторную работу № 1

Вариант	$F = V(\dots\dots\dots)$	Не определены
1.	0,2,4,8,10,12,14	1,9,13
2.	0,3,4,8,10,13	2,6,14,15
3.	0,2,5,8,11,12,14	1,4,7
4.	0,2,4,9,10,12	3,7,13,15
5.	0,2,3,5,7,8,11	4,6,14
6.	0,3,5,7,12,15	6,9,10,13
7.	0,4,8,9,11,13,15	2,3,10
8.	0,4,6,8,10,13	2,7,11,14
9.	0,3,5,8,10,12,13	2,6,11
10.	0,4,5,7,9,13	3,6,8,11
11.	0,1,3,5,8,11,15	7,9,12
12.	0,1,4,7,10,12	2,3,8,13
13.	0,1,5,6,10,12,14	3,7,9
14.	0,1,5,7,9,13	4,6,14,15
15.	0,2,5,9,10,12,15	3,8,13
16.	1,6,9,11,13,15	2,4,12,14

17.	1,2,6,9,11,14,15	3,10,12
18.	1,3,7,9,10,14	2,5,11,15
19.	1,3,5,8,10,12,14	4,9,13
20.	1,3,6,11,13,15	2,6,9,14
21.	1,4,6,8,10,12,13	3,7,11
22.	1,5,6,7,9,15	2,3,8,10
23.	1,6,7,8,10,11,15	3,8,14
24.	1,2,4,5,7,13	6,8,10,15
25.	1,3,8,9,11,14,15	4,6,10
26.	1,7,9,12,14,15	2,3,5,10
27.	1,5,7,10,11,13,14	0,3,8
28.	1,2,7,9,10,12	0,5,6,13
29.	1,2,7,8,10,12,14	3,11,15
30.	1,3,5,9,13,15	0,6,10,12
31.	2,3,5,8,11,13,14	1,4,10
32.	2,6,8,9,11,13	7,10,12,15
33.	2,5,7,8,10,12,15	0,6,11
34.	2,4,8,9,12,13	1,5,10,14
35.	2,3,7,9,11,14,15	4,5,10
36.	2,5,6,9,10,12	1,7,8,15
37.	2,4,6,8,13,14,15	0,9,10
38.	2,5,9,11,13,15	1,6,7,10
39.	2,3,4,6,8,10,12	0,5,9
40.	2,3,6,9,12,13	1,4,8,14
41.	2,3,7,9,11,13,15	0,8,10

42.	2,4,7,9,11,14	5,8,10,15
43.	2,5,8,9,11,13,15	1,6,7
44.	2,4,8,10,13,15	0,5,9,14
45.	3,5,8,10,11,13,15	1,6,12
46.	2,6,7,10,12,14	0,8,11,15
47.	3,4,6,7,9,12,14	5,8,10
48.	3,5,7,8,13,15	4,6,9,11
49.	3,4,6,9,10,13,14	1,5,8
50.	3,6,8,9,11,13	0,7,10,15
51.	3,6,7,10,12,13,15	4,5,9
52.	3,7,8,10,12,15	1,4,9,13
53.	3,4,5,7,9,11,13	6,10,14
54.	3,7,9,11,12,15	0,5,10,13
55.	3,5,6,9,10,12,14	1,7,11
56.	4,5,7,9,11,12	1,3,8,14
57.	4,6,7,9,12,14,15	0,2,5
58.	4,8,10,12,13,15	1,3,7,11
59.	4,6,8,10,11,13,14	1,3,9
60.	4,8,9,11,13,14	0,3,10,15

**\*Приведены два варианта заданий для удобства преподавателя.**

Порядок выполнения работы

1. Произвести кодирование состояний, входных и выходных сигналов абстрактного автомата, заданного графическим или табличным способом.

2. Составить кодированную таблицу переходов, таблицу

кодированных выходов и таблицу функций возбуждения триггеров заданного типа.

3. Получить минимальные выражения для функций кодированных выходов и функций возбуждения триггеров в булевом базисе.

4. Полученные выражения перевести в заданный универсальный базис.

5. Построить на эмуляторе синхронизированную схему автомата в заданном базисе.

7. Проверить работу автомата в соответствии с таблицами переходов и выходов.

#### Примеры выполнения заданий на лабораторную работу

Синтез КС с одним выходом.

Представление реализуемой ПФ в булевом базисе. По таблице истинности ПФ может быть получена совершенная дизъюнктивная нормальная форма (СДНФ) записи функции [1]. Для получения СДНФ нужно выписать произведения, соответствующие наборам значений аргументов, при которых функция обращается в 1. При этом, если в данном наборе аргумент  $x_i = 1$ , то он вписывается в соответствующее произведение без изменения; если же  $x_i = 0$ , то в произведение вписывается его отрицание  $\bar{x}_i$ . Все полученные произведения соединяются между собой знаками дизъюнкций. Для ПФ, заданной таблицей, можно записать СДНФ:

$$y = \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2 x_3 \vee \bar{x}_1 x_2 \bar{x}_3$$

x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	x <sub>4</sub>
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Каждое из полученных произведений равно 1 только при определенном наборе значений всех аргументов функции и носит название *минтерма*.

Преобразуем полученную СДНФ следующим образом:

$$y = \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2 (x_3 \vee \bar{x}_3) = \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2$$

Такая дизъюнкция произведений, члены которой могут и не быть минтермами, называется дизъюнктивной нормальной формой (ДНФ). ДРФ, содержащая минимальное число букв, называется минимальной ДНФ (МДНФ).

ПФ может быть определена также и нулевыми ее значениями. Так, рассматриваемая функция (см. таблицу) равна 0, если равно 1 любое из произведений:

$$\bar{x}_1 \bar{x}_2 \bar{x}_3, \bar{x}_1 \bar{x}_2 x_3, \bar{x}_1 x_2 \bar{x}_3, \bar{x}_1 x_2 x_3, x_1 \bar{x}_2 \bar{x}_3, \text{ т.е.}$$

Или, в соответствии с правилом де Моргана,

$$y = (x_1 \vee x_2 \vee x_3)(x_1 \vee x_2 \vee \bar{x}_3)(x_1 \vee \bar{x}_2 \vee x_3)(x_1 \vee \bar{x}_2 \vee \bar{x}_3)(\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3)$$

Каждая дизъюнкция полученного произведения равна 0 только при определенном наборе значений всех аргументов функ-



ции и носит название *макстерма*. Форма представления ПФ в виде произведения макстермов есть совершенная конъюнктивная нормальная форма (СКНФ). Путем склеивания членов полученной СКНФ [1], можно получить выражение:  $y = (x_1 \vee x_2)(\bar{x}_2 \vee x_3)(\bar{x}_1 \vee \bar{x}_2)$ .

Такое произведение дизъюнкций, члены которого могут и не быть макстермами, есть конъюнктивная нормальная форма (КНФ). КНФ, содержащая минимальное число букв, называется минимальной КНФ (МКНФ).

Минимизация ПФ. Существуют различные методы, позволяющие получать МДНФ и МКНФ функций [1 - 3]. Для функций с небольшим числом переменных (до 5-6) наиболее удобным методом минимизации является метод диаграмм Вейча, основанный на использовании операции склеивания:  $x y \vee x \bar{y} = x(y \vee \bar{y}) = x$ . Диаграмма Вейча является графическим представлением совокупности всех минтермов для данного числа переменных. Каждый минтерм изображается на диаграмме в виде клетки. Диаграмма образуется путем такого расположения клеток, при котором минтермы, находящиеся в соседних клетках, отличаются значением одной переменной.

Диаграммы Вейча для функций двух, трех и четырех переменных представлены на рис.1.11. В каждой из диаграмм наборы значений переменных, указанных по обе стороны диагональной черты в левом верхнем углу диаграммы, представлены с внешней стороны соответствующего столбца или строки, вдоль которых эти переменные остаются постоянными. Минтермы функции отмечаются единицами в соответствующих клетках диаграммы. Минтермы, не входящие в функцию, отмечаются в клетках нулями (либо клетки оставляются пустыми).

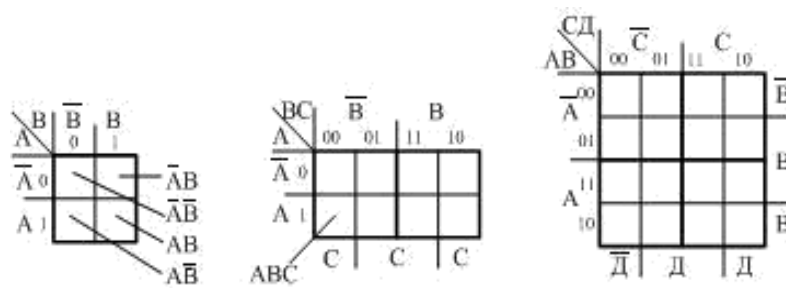


Рис.11

На основании операции склеивания два минтерма, находящиеся в соседних клетках, можно заменить одним произведением, содержащим на одну переменную меньше. Если соседними являются две пары минтермов, то такая группа из четырех минтермов заменяется произведением, уменьшенным на две переменные. В общем случае наличие единиц в  $2^n$  соседних клетках позволяет исключить  $n$  переменных.

Следует заметить, что в диаграммах соседними считаются также крайние клетки каждого столбца или строки, так как расположенные в них минтермы отличаются значением одной переменной.

Минимизация ПФ с помощью диаграммы сводится к такому объединению всех соседних единиц в группы, при котором каждая группа содержит максимальное число единиц, а количество групп минимально. Практически минимизацию выполняют следующим образом. Каждую группу единиц накрывают овалом (рис.1.12). Каждому овалу ставят в соответствие произведение, в котором отсутствуют переменные, изменяющие в данном овале свои значения. Переменная или ее отрицание остается в произведении, если она сохраняет в овале единичное или нулевое значение соответственно. При этом удобно пользоваться обозначениями переменных и их отрицаний, проставленными вне поля диаграммы. При минимизации следует иметь в виду,

что одна и та же клетка диаграммы может покрываться несколькими различными овалами.

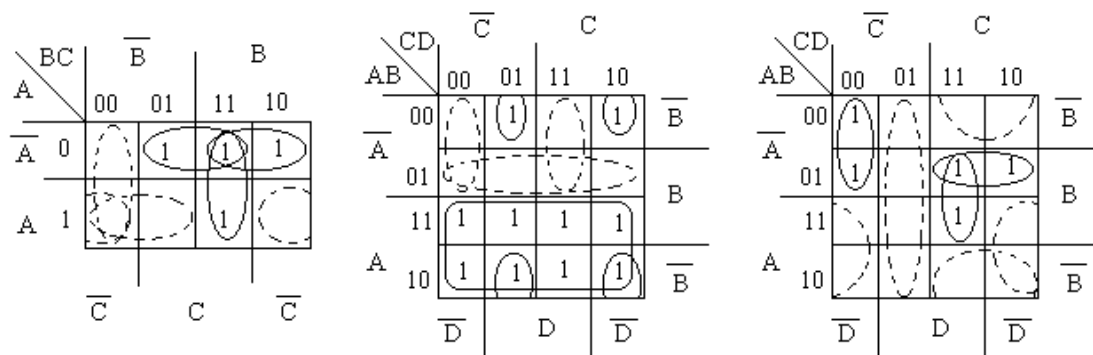


Рис.1.12

Для функций, диаграммы которых представлены на рис.1.12, можно получить следующие МДНФ:

$$f_a = \overline{A}C \vee \overline{A}B \vee BC; f_b = A \vee \overline{B}C \vee \overline{B}C \vee \overline{B}C; f_g = \overline{A}C \vee \overline{A}C \vee \overline{A}B \vee \overline{A}B$$

Диаграмма Вейча позволяет получить и МКНФ функции. Для этого сначала получают МДНФ для отрицания функции  $\overline{f}$ , объединяя на диаграмме пустые клетки. Так, из рис.1.12,б следует, что  $\overline{f_b} = \overline{C}D \vee \overline{A}D \vee \overline{B}C$ . Далее, с помощью правила де Моргана, получают МКНФ функции  $f$ . Для функции рис.12,в МКНФ имеет вид  $\overline{f_b} = \overline{C}D \vee \overline{A}D \vee \overline{B}C = (C \vee \overline{D})(\overline{A} \vee D)(B \vee \overline{C})$  и является более простой, чем МДНФ.

Реализация ПФ на универсальных элементах. После нахождения МДНФ и МКНФ функции, реализуемой на элементах универсального базиса, в полученных выражениях осуществляется переход к базисам И-НЕ либо ИЛИ-НЕ.

Переход к базису И-НЕ. Представим операции НЕ, И и ИЛИ в базисе И-НЕ:

$$\overline{A} = \overline{A \cdot A} = \overline{A}; AB = \overline{\overline{A \cdot B}} = \overline{A \cdot B}; A \vee B = \overline{\overline{A \cdot B}} = \overline{A \cdot B}$$

Отсюда следуют правила перехода: инверсия осуществляется подачей аргумента на элемент И-НЕ; конъюнкция реализуется подачей аргументов на элемент И-НЕ с последующей инверсией; дизъюнкция выполняется инвертированием аргументов с последующей подачей их на элемент И-НЕ.

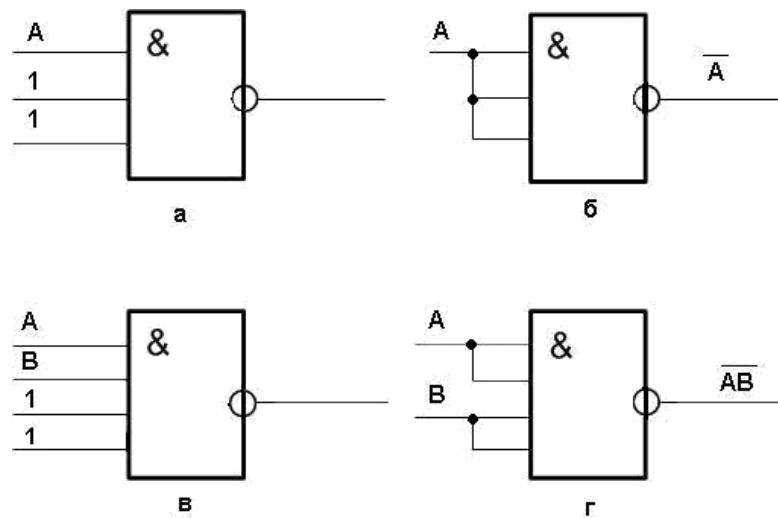


Рис.1.13

В общем случае на свободные входы элемента И-НЕ должны быть поданы либо константы "1", либо подключены уже используемые в элементе входные сигналы (рис.1.13). В лабораторном макете сигналы на свободных входах элементов И-НЕ соответствуют константе "1", поэтому эти входы можно оставить незадействованными (на схемах свободные входы можно не показывать).

Преобразуем МДНФ и МКНФ функции (рис.1.12,б). Здесь и в дальнейшем предполагается, что исходные переменные КС поступают в парафазном коде (т.е. каждая переменная  $x_i$  поступает по двум проводам: по одному – прямое значение  $x_i$ , по другому - инверсное  $\overline{x_i}$ ).

МДНФ:

$$f_{\delta} = A\nu\overline{BCD}\nu\overline{BCD} = \overline{\overline{A\nu\overline{BCD}\nu\overline{BCD}}} = \overline{A \cdot \overline{BCD}\nu\overline{BCD}} = \overline{[A, /(\overline{B,C,D}), /(\overline{B,C,D})]}.$$

Соответствующая КС приведена на рис.1.14, а. Здесь число уровней  $r = 2$ , число входов элементов  $S=9$ .

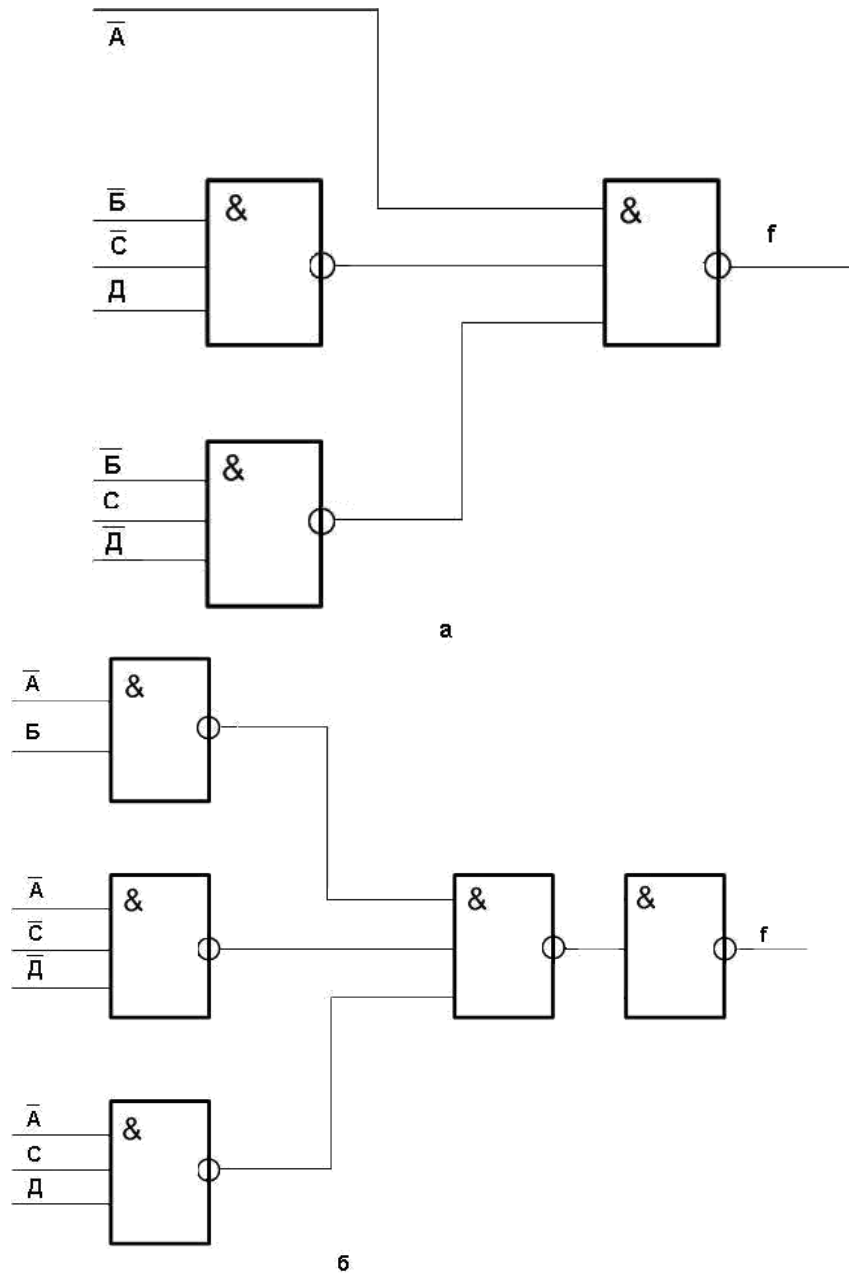


Рис. 14

$$f_{\delta} = (A\nu\overline{B})(A\nu C\nu D)(A\nu\overline{C}\nu\overline{D}) = \overline{\overline{A\nu\overline{B}}}\overline{\overline{A\nu C\nu D}}\overline{\overline{A\nu\overline{C}\nu\overline{D}}} = \overline{\overline{AB}} \cdot \overline{\overline{ACD}} \cdot \overline{\overline{ACD}} = \text{МКНФ: } = / \{ /(\overline{A}, B), /(\overline{A}, \overline{C}, \overline{D}), /(\overline{A}, C, D) \}$$

Соответствующая КС приведена на рис.1.14,б. Здесь  $r = 3$ ,  $S = 12$ .

Из примера видно, что реализация на элементах И-НЕ МДНФ функции приводит к двухуровневой схеме, реализация МКНФ – трехуровневой схеме.

Переход к базису ИЛИ-НЕ

$$\begin{aligned} \bar{A} &= \overline{A \vee A} = \downarrow(A); \overline{AB} = \overline{A \vee B} = \\ &= \downarrow(\bar{A}, \bar{B}) = \downarrow[\downarrow(A), \downarrow(B)]; \\ A \vee B &= \overline{\overline{A \vee B}} = \downarrow[\downarrow(A), \downarrow(B)] \end{aligned}$$

Правила перехода: инверсия реализуется подачей аргумента на элемент ИЛИ-НЕ; конъюнкция получается инвертированием аргументов с последующей подачей их на элемент ИЛИ-НЕ; дизъюнкция выполняется подачей аргументов на элемент ИЛИ-НЕ с последующей инверсией.

В общем случае на свободные входы элемента ИЛИ-НЕ должны быть поданы либо константы 0, либо подключены уже используемые в элементе входные сигналы.

Преобразуем МДНФ и МКНФ функции рис.1.12,б.

$$\begin{aligned} f_s &= A \vee \overline{B \overline{C} D} \vee \overline{B \overline{C} D} = \overline{\overline{A \vee \overline{B \overline{C} D} \vee \overline{B \overline{C} D}}} = \overline{A \vee (B \vee C \vee \overline{D})} \vee (B \vee C \vee \overline{D}) = \\ \text{МДНФ: } &= \downarrow \left\{ \downarrow [A, \downarrow (B, C, \overline{D})], \downarrow (B, \overline{C}, D) \right\} \end{aligned}$$

В соответствующей схеме (рис.1.15,а)  $r = 3, S = 10$ .

МКНФ:

$$\begin{aligned} f_s &= (A \vee \bar{B})(A \vee C \vee D)(A \vee \bar{C} \vee \bar{D}) = \overline{\overline{(A \vee \bar{B})(A \vee C \vee D)(A \vee \bar{C} \vee \bar{D})}} = \overline{(A \vee \bar{B})(A \vee C \vee D)(A \vee \bar{C} \vee \bar{D})} = \\ &= \downarrow \left\{ \downarrow (A, \bar{B}), \downarrow (A, C, D), \downarrow (A, \bar{C}, \bar{D}) \right\} \end{aligned}$$

В схеме (рис.1.15,б)  $r = 2, S = 11$ . Как видно, реализация на элементах ИЛИ-НЕ МДНФ функции приводит к трехуровневой схеме, реализация МКНФ - к двухуровневой схеме.

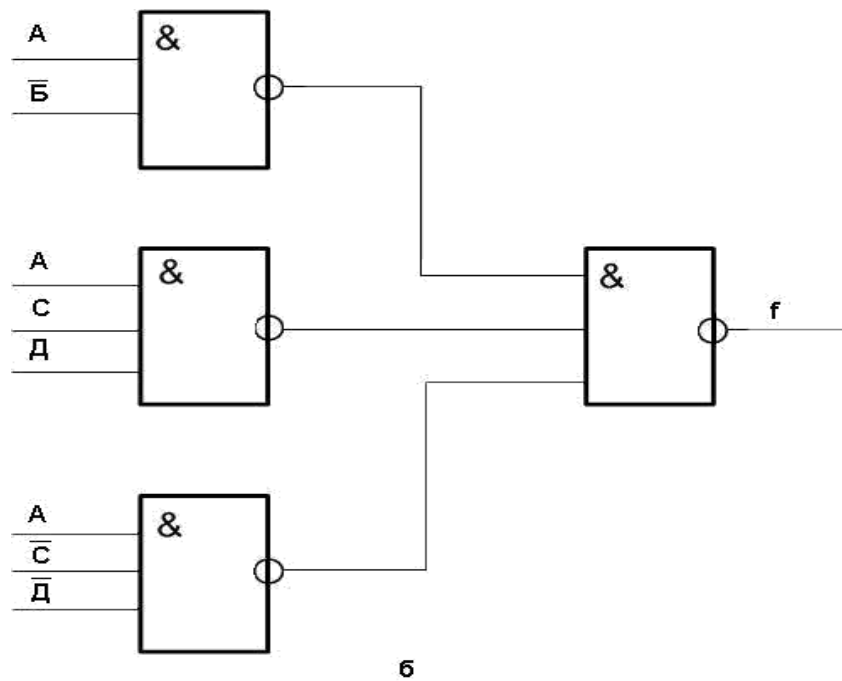
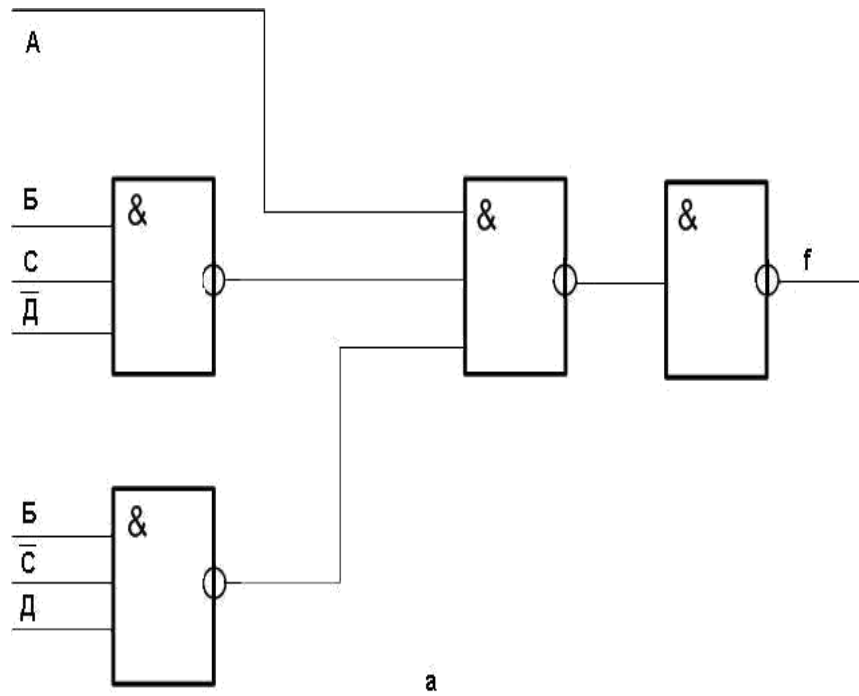


Рис.1.15

Из четырех схем, реализующих функцию рис.1.12,б предпочтителен вариант на рис.1.14, а ( $\gamma = 2$ ,  $S = 9$ ). Если же в КС необходимо использовать элементы ИЛИ-НЕ и допускается использование трехуровневой схемы, то следует выбрать вариант рис.1.15, а. ( $\gamma = 3$ ,  $S = 10$ ), и т.д. В общем случае выбирается ва-

риант КС, который при заданных значениях  $J$  и  $F$  является оптимальным в смысле минимума используемого в схеме оборудования при выполнении ограничений на быстродействие КС. Следует также иметь в виду, что синтезируемые схемы часто можно дополнительно упростить, применяя скобочные формы представления функций [3]. Скобочные формы в булевом базисе получают путем вынесения за скобки общих частей нескольких произведений (в ДНФ) или нескольких дизъюнкций (в КНФ). Так, например, получим скобочную форму из МДНФ функции рис. 1.12,а:

$$f_a = \bar{A}Cv\bar{A}BvBC = \bar{A}(CvB)vBC.$$

Переведем МДНФ и скобочную форму в базис И-НЕ.

$$f_a = \overline{\overline{\bar{A}Cv\bar{A}BvBC}} = \overline{\overline{\bar{A}C}v\overline{\overline{\bar{A}B}v\overline{\overline{BC}}}} = \overline{/\overline{(\bar{A}, C)}, /(\bar{A}, B)}, /(\overline{BC})}$$

Скобочная форма:

$$f_a = \bar{A}(\overline{\overline{CvB}})vBC = \overline{\overline{\bar{A} \cdot \overline{\overline{CB}}v\overline{\overline{BC}}}} = \overline{\overline{\bar{A} \cdot \overline{\overline{CB}} \cdot \overline{\overline{BC}}}} = \overline{/\overline{[\bar{A}, /(\overline{CB})]}, /(\overline{BC})}$$

Схемы, реализующие МДНФ ( $r = 2, S = 9$ ) и скобочную форму ( $r = 3, S = 8$ ), приведены на рис. 1.16. Схема рис. 1.16,б не только проще, но и требует лишь двухвходовых элементов. Но в любом случае скобочные формы приводят к увеличению числа уровней  $r$  в схеме.

На практике к скобочным формам прибегают, наряду с разделением входов, для получения схем, удовлетворяющих требуемому



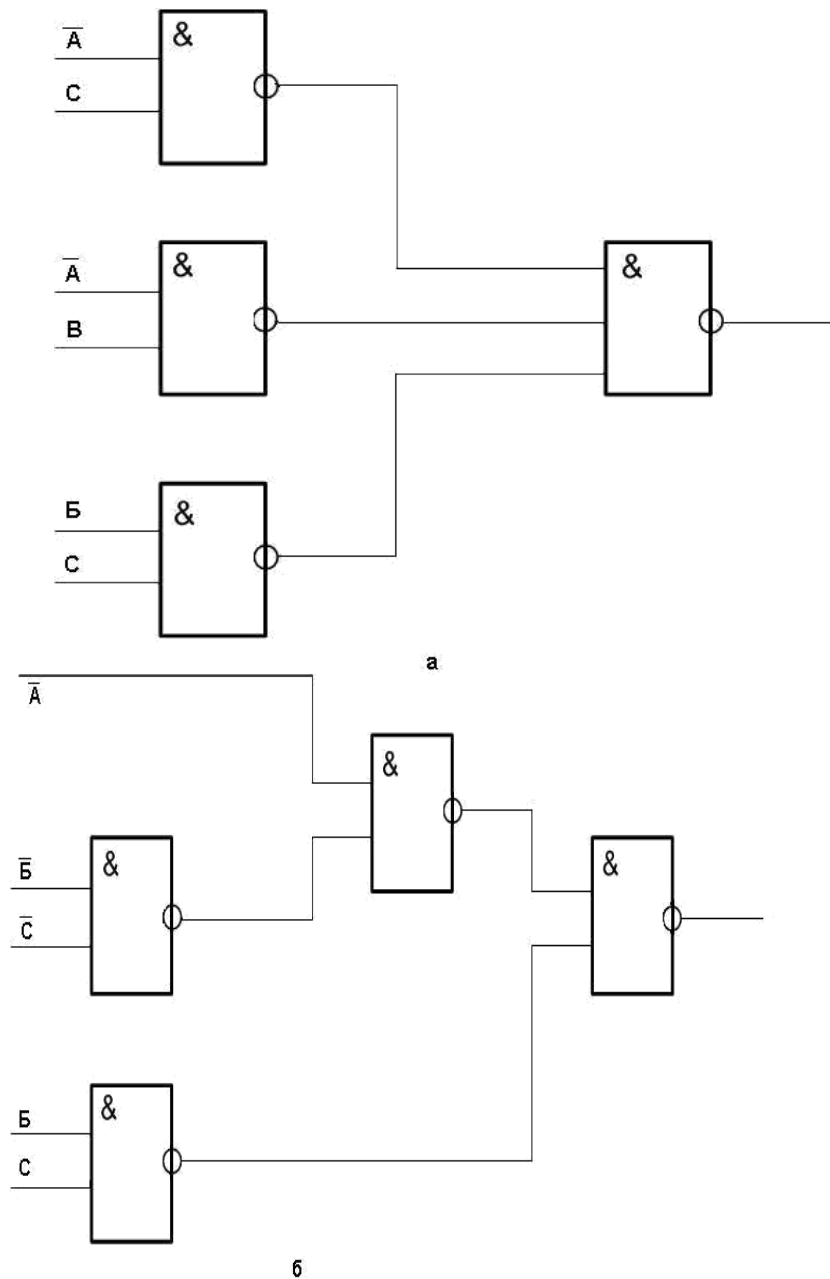


Рис.1.16

На практике к скобочным формам прибегают, наряду с разделением входов, для получения схем, удовлетворяющих требуемому значению коэффициента  $J[3]$ . При этом разделение входов используется в последнюю очередь, либо сопровождается увеличением числа входов  $S$  в КС.

Синтез КС со многими выходами Задачу синтеза схемы с  $m$  выходами, работа которой описывается системой из  $m$  функций, можно свести к задаче синтеза КС с одним выходом, если каждую

из функций реализовать отдельно. При этом получаем искомую схему с  $t$  выходами, состоящую из  $t$  независимых КС. Однако в общем случае схему можно существенно упростить, объединив участки схемы, реализующие одинаковые выражения в уравнениях нескольких функций.

Общая идея минимизации схем со многими выходами сводится к получению таких выражений для совокупности переключательных функций, в которых оптимально используются члены, общие для нескольких функций [1].

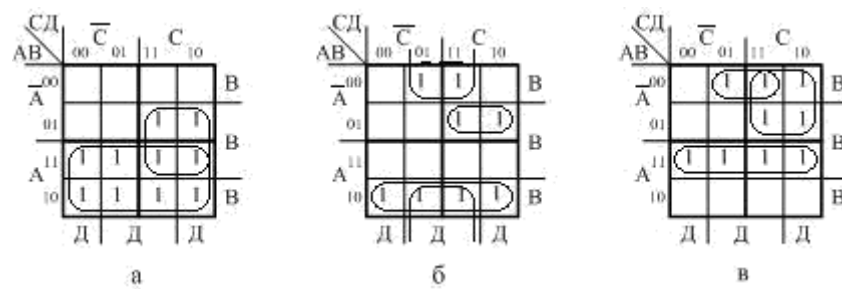


Рис.1.17

Пусть, например, требуется построить двухуровневую КС, реализующую на элементах булевого базиса систему трех функций, заданных в ДНФ. Диаграммы Вейча функций представлены на рис.1.17. При отдельной минимизации можно получить следующие МДНФ функций:  $f_a = AvBC; f_b = \bar{A}\bar{B}\bar{D}v\bar{A}BC; f_c = Av\bar{A}\bar{B}Dv\bar{A}C$ .

Схема, построенная по этим выражениям, будет содержать 24 входа элементов ( $S = 24$ ). В то же время можно построить искомую схему с  $S = 21$ , если согласно рис.1.18 представить функции в виде следующих ДНФ:

$$f_a = \bar{A}BCvAv\bar{A}\bar{B}$$

$$f_b = \bar{A}\bar{B}Dv\bar{A}BCv\bar{A}\bar{B}; f_c = \bar{A}\bar{B}Dv\bar{A}CvAv$$

При этом предполагается, что любой из дизъюнктивных членов  $(\bar{A}BC, \bar{A}\bar{B}, \bar{A}\bar{B}D, Av)$ , повторяющихся в выражениях для не-

скольких функций, будет реализовываться одним элементом, общим для этих функций.

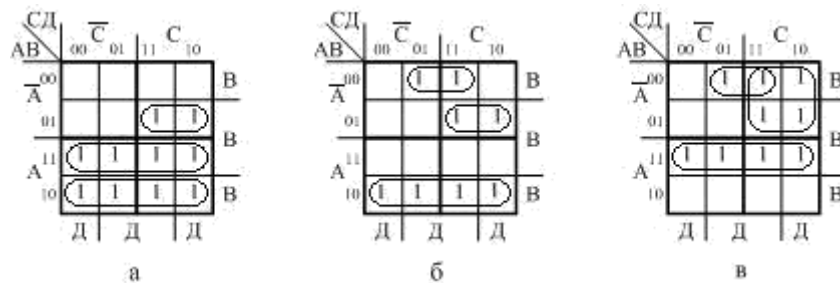


Рис.1.18

Синтез схем с многими выходами, оптимальных по количеству оборудования, связан с практически неосуществимым перебором большого числа вариантов схемы. Известные методы синтеза [1,3] позволяют сократить перебор, однако при этом получаются не оптимальные, а только в той или иной степени близкие к ним схемы.

### Содержание отчёта

1. Представление реализуемых ПФ таблицами истинности и диаграммами Вейча.
2. Получение минимальных нормальных форм ПФ.
3. Переход к базисам И-НЕ и ИЛИ-НЕ.
4. Функциональные схемы на элементах И-НЕ и ИЛИ-НЕ, собранные в среде эмулятора.

### Приложение

#### 1.2. Функции в алгебре логики

Рассмотрение понятия функции в алгебре логики (АЛ) можно начать с функций одной переменной. Нетрудно видеть, что таких функций можно построить четыре (табл. 1.1).

Таблица 1.1. Функции одной переменной в АЛ

Переменная	Функции			
$x$	$g_0$	$g_1$	$g_2$	$g_3$
0	0	0	1	1
1	0	1	0	1

Очевиден и содержательный смысл этих функций:  $g_0$  - константа нуля,  $g_1$  - повторение  $x$ ,  $g_2$  - инверсия  $x$ ,  $g_3$  - константа единицы.

Для двух переменных может быть введено уже 16 функций (табл. 1.2).

Таблица 1.2. Функции двух переменных в АЛ

Переменные	$x_1$	0	0	1	1
	$x_2$	0	1	0	1
Функции					
$f_0$		0	0	0	0
$f_1$		0	0	0	1
$f_2$		0	0	1	0
$f_3$		0	0	1	1
$f_4$		0	1	0	0
$f_5$		0	1	0	1
$f_6$		0	1	1	0
$f_7$		0	1	1	1
$f_8$		1	0	0	0
$f_9$		1	0	0	1

$f_{10}$	1	0	1	0
$f_{11}$	1	0	1	1
$f_{12}$	1	1	0	0
$f_{13}$	1	1	0	1
$f_{14}$	1	1	1	0
$f_{15}$	1	1	1	1

Продолжая этот ряд, получим таблицу 1.3, показывающую, что количество логических функций вычисляется как два в степени количества возможных входных наборов.

Таблица 1.3. Зависимость числа логических функций от числа входных переменных

Количество входных переменных	1	2	3	...	n
Число входных наборов	$2^1$	$2^2$	$2^3$	...	$2^n$
Число логических функций	$2^1$	$2^4$	$2^8$	...	$2^{2^n}$

Логическая функция определяется как n-местная функция, определенная на множестве истинных значений <Истина (True), Ложь(False)> и принимающая значения в этом множестве.

Если последовательность логических переменных обозначить как  $X=(x_1, x_2, \dots, x_n)$  и назвать двоичным набором, то под функцией алгебры логики следует понимать однозначное отображение множества всевозможных наборов \* на множество  $Y=<0,1>$ .

Если две функции алгебры логики  $f_1(x_1, x_2, \dots, x_n)$  и  $f_2(x_1, x_2, \dots, x_n)$  принимают на всех возможных наборах одинаковые значения, то они называются равными (эквивалентными). Функции двух пе-

ременных, рассмотренные в табл. 1.4 играют важную роль в алгебре логики и могут быть названы элементарными.

Таблица 1.4. Содержательная таблица функций двух переменных в АЛ

Функция в аналитическом выражении	Наименование	Словесное выражение	Выражение в элементарном базисе	Функциональное обозначение
$f_0=0$	Константа "0"	Всегда ложно	$x_1x_1 \vee x_2x_2$	<u>рис. 1.1.а</u>
$f_1=x_1x_2$ $f_1=x_1 \& x_2$	Конъюнкция, И	$x_1$ и $x_2$	$x_1 \& x_2$	<u>рис. 1.1.б</u>
$f_2=x_1 \neg x_2$	Запрет $x_1$	Запрет по $x_2$	$x_1x_2$	<u>рис. 1.1.в</u>
$f_3=x_1$	Повторение $x_1$	Повторение $x_1$	$x_1$	<u>рис. 1.1.г</u>
$f_4=x_2 \neg x_1$	Запрет $x_2$	Запрет по $x_1$	$x_1x_2$	<u>рис. 1.1.д</u>
$f_5=x_2$	Повторение $x_2$	Повторение $x_2$	$x_2$	<u>рис. 1.1.е</u>
$f_6=x_1 \oplus x_2$	Сложение по модулю 2, неравнозначность, исключающее ИЛИ	$x_1$ неравнозначно $x_2$	$x_1x_2 \vee x_1\neg x_2$	<u>рис. 1.1.ж</u>
$f_7=x_1 \vee x_2$ $f_7=x_1 + x_2$	Дизъюнкция, ИЛИ	$x_1$ или $x_2$	$x_1 \vee x_2$	<u>рис. 1.1.з</u>
$f_8=x_1 \nabla x_2$	Стрелка Пирса, ИЛИ-НЕ	не $x_1$ и не $x_2$	$x_1 \vee x_2$	<u>рис. 1.1.и</u>
$f_9=x_1 \boxplus x_2$	Равнозначность, эквивалентность	$x_1$ равнозначно $x_2$	$x_1x_2 \vee x_1\neg x_2$	<u>рис. 1.1.к</u>
$f_{10}=\neg x_2$	Инверсия, отрицание $x_2$	Не $x_2$	$x_2$	<u>рис. 1.1.л</u>

$f_{11}=x_2 \square x_1$	Импликация $x_1$	Если $x_2$ , то $x_1$	$x_1 \vee x_2$	<u>рис. 1.1.м</u>
$f_{12}=x_1$	Инверсия, отрицание $x_1$	Не $x_1$	$x_1$	<u>рис. 1.1.н</u>
$f_{13}=x_1 \square x_2$	Импликация $x_2$	Если $x_1$ , то $x_2$	$x_1 \vee x_2$	<u>рис. 1.1.о</u>
$f_{14}=x_1   x_2$	<u>Штрих Шеффера</u> , И-НЕ	Не $x_1$ или не $x_2$	$x_1 \vee x_2$	<u>рис. 1.1.п</u>
$f_{15}=1$	Константа “1”	Всегда истинно	$(x_1 \vee x_1) (x_2 \vee x_2)$	<u>рис. 1.1.р</u>

К элементарным функциям обычно относят: функцию инверсии (отрицания), конъюнкцию, дизъюнкцию, импликацию, штрих Шеффера и стрелку Пирса.

Новые функции АЛ можно получить из известных функций либо путем перенумерации аргументов, либо путем подстановки в функцию новых функций вместо аргументов.

Функция АЛ, полученная из функций  $f_1, f_2, \dots, f_k$  с помощью этих правил, называется суперпозицией функций  $f_1, f_2, \dots, f_k$ . В табл. 1.4 приведено представление различных функций через суперпозицию конъюнкции, дизъюнкции и отрицания.

# Лабораторная работа 2

## *СИНТЕЗ КОНЕЧНЫХ АВТОМАТОВ*

**Цель работы:** изучение процедуры структурного синтеза конечных автоматов.

### I. Конечные автоматы

Автоматом называется дискретное устройство, способное принимать различные состояния, под воздействием входных сигналов переходить из одного состояния в другое и вырабатывать выходные сигналы. Математической моделью устройства является абстрактный автомат, который задается совокупностью пяти объектов  $S(A, X, Y, \delta, \lambda)$ , где

$A = \{a_0, a_1, a_2, \dots, a_m, \dots, a_M\}$  - множество состояний автомата, причем,  $a_0$ - исходное (начальное) состояние;

$X = \{x_1, x_2, \dots, x_f, \dots, x_F\}$  - множество входных сигналов;

$Y = \{y_1, y_2, \dots, y_g, \dots, y_G\}$  - множество выходных сигналов;

$\delta$  - функция переходов, обеспечивающая выработку последующего состояния  $a_s$  автомата в зависимости от существующего состояния  $a_t$  и входного сигнала  $x_f$ , т.е.  $a_s = \delta(a_t, x_f)$ ;

$\lambda$  - функция выходов, обеспечивающая выработку выходного сигнала автомата в зависимости от  $a_m$ , и  $x_f$ , т.е.  $y_g = \lambda(a_m, x_f)$ .

Если множества  $A, X, Y$  конечны, то автомат называется конечным.



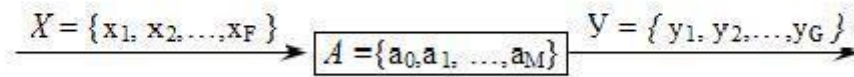


рис.1

Абстрактный автомат имеет один входной и один выходной каналы (рис.1) и функционирует в дискретные моменты времени, которые обычно обозначаются натуральными числами:  $t = 0, 1, 2, \dots, n, \dots$ . В каждый момент дискретного времени  $t$  автомат находится в определенном состоянии  $a(t)$ , причем в момент  $t = 0$  он всегда находится в исходном состоянии  $a(0) = a_0$ . В момент  $t$  автомат, находясь в состоянии  $a(t)$ , воспринимает сигнал  $x(t)$  на входном канале, вырабатывает на выходе сигнал  $y(t) = \lambda[a(t), x(t)]$  и переходит в новое состояние, которое к следующему моменту дискретного времени определяется как  $a(t+1) = \delta[a(t), x(t)]$ .

Наибольшее распространение получили автоматы Мили и Мура. Закон функционирования автомата Мили задается следующими уравнениями:

$$a(t+1) = \delta[a(t), x(t)]; y(t) = \lambda[a(t), x(t)].$$

Работа автомата Мура определяется уравнениями:

$$a(t+1) = \delta[a(t), x(t)]; y(t) = \lambda[a(t)].$$

Как видно, в автомате Мура выходные сигналы зависят лишь от состояния автомата.

## Способы задания конечных автоматов

**Табличный способ.** Автомат Мили может быть задан таблицей переходов, определяющей функцию переходов  $\delta$ , и таблицей выходов, определяющей функцию выходов  $\lambda$ . Строки этих таблиц соответствуют возможным входным сигналам  $x_f$ , а столбцы - возможным состоянием  $a_m$  автомата. В таблице переходов (рис.2, а) на пересечении столбца  $a_m$  и строки  $x_f$  находится состояние  $a_s = \delta(a_m, x_f)$ . В таблице, выходов (рис. 2,б) в аналогичной клетке помещается выходной сигнал  $y_g = \lambda(a_m, x_f)$ . Как видно, здесь задан автомат, имеющий множества  $A = \{a_0, a_1, a_2\}$ ;  $X = \{x_1, x_2\}$ ;  $Y = \{y_1, y_2, y_3\}$ .

Так как в автомате Мура выходные сигналы зависят лишь от состояния, то он задается одной так называемой отмеченной таблицей переходов (рис.2,в). В этой таблице над каждым состоянием  $a$  автомата, обозначающим тот или иной столбец таблицы, стоит соответствующий этому состоянию выходной сигнал  $y_g = \lambda(a_m)$ .

$a_m \backslash x_f$	$a_0$	$a_1$	$a_2$
$x_1$	$a_1$	$a_1$	$a_1$
$x_2$	$a_2$	$a_2$	$a_0$

а

$a_m \backslash x_f$	$a_0$	$a_1$	$a_2$
$x_1$	$y_1$	$y_2$	$y_2$
$x_2$	$y_3$	$y_3$	$y_2$

б

$y_g$	$y_1$	$y_3$	$y_2$	$y_2$	$y_1$
$a_m \backslash x_f$	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$
$x_1$	$a_1$	$a_0$	$a_4$	$a_0$	$a_4$
$x_2$	$a_3$	$a_2$	$a_3$	$a_2$	$a_3$

в

Рис.2

**Графический способ.** Этот способ основан на использовании направленных графов. Вершины графов соответствуют состояниям, а дуги - возможным переходам между ними. Две

вершины  $a_m$ , и  $a_s$  графа соединяются дугой, направленной от  $a_m$ , к  $a_s$ , если существует переход  $a_s = \delta(a_m, x_f)$ . Дуге автомата Мили приписывается входной сигнал  $x_f$ , и выходной сигнал  $y_g = \lambda(a_m)$ . В автомате Мура выходной сигнал  $y_g = \lambda(a_m)$  записывается внутри вершины  $a_m$ . Графы автоматов Мили и Мура, заданных таблицами рис.2, представлены на рис.3.

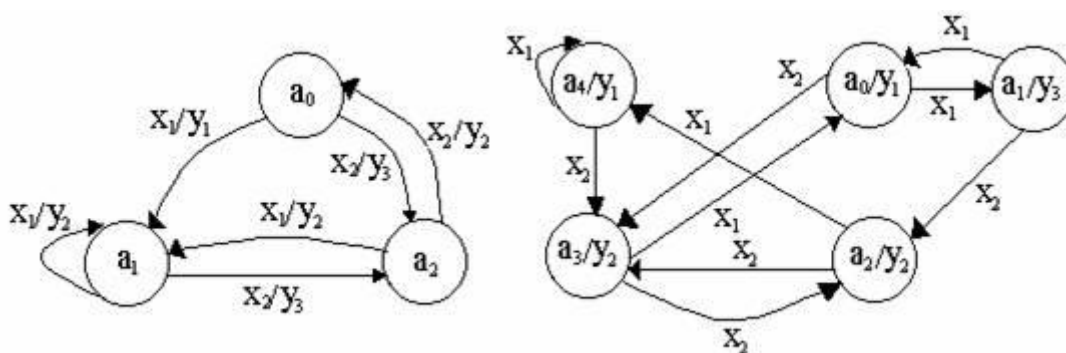


Рис.3

Процедура синтеза конечного автомата в общем случае делится на два этапа: этап абстрактного синтеза и этап структурного синтеза. Содержанием первого этапа является построение графа или таблиц переходов и выходов и получение множества состояний  $A$  с минимальным числом членов [1]. Цель этапа структурного синтеза - построение схемы автомата на заданных логических элементах и элементах памяти.

Согласно теореме о структурной полноте [1] система элементов, из которых может быть построена схема любого конечного автомата, должна содержать логические элементы ба-

зиса и в качестве элемента памяти - автомат Мура, обладающий полнотой переходов и выходов. Полнота переходов в автомате означает, что для любой пары состояний существует входной сигнал, который переводит автомат из одного состояния в другое. Иначе говоря, в каждом столбце таблицы переходов должны встречаться все состояния автомата. Полнота выходов означает, что каждое возможное состояние автомата отмечет отличным от других выходным сигналом.

В качестве элементов памяти широко используются элементарные автоматы Мура с двумя состояниями. Каждый из таких автоматов выдает два различных выходных сигнала, соответствующих двум его различным состояниям. В дальнейшем указанные состояния и соответствующие им выходные сигналы будут обозначаться одной буквой  $Q$  и кодироваться цифрами 0 и 1 ( $Q \in \{0,1\}$ ).

Элементарный автомат может иметь в общем случае несколько физических входов, на каждый из которых могут подаваться двоичные сигналы. Условное обозначение элементарного автомата приведено на рис.4. Здесь  $u_{ji}$ ,  $j=1, 2, \dots, K$  – сигналы возбуждения элементарного автомата ( $u_{ji} \in \{0, 1\}$ ).

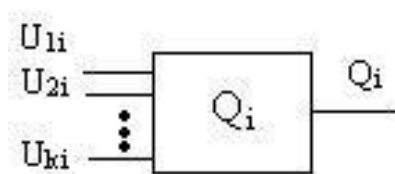


рис. 4

**Структурная схема конечного автомата** (рис.5) состоит из двух основных частей: комбинационной части (КЧ) и запоминающей части (ЗЧ). КЧ строится из логических элементов базиса (ЗЧ) представляет собой набор элементарных автоматов  $Q_1, Q_2, Q_3$ .

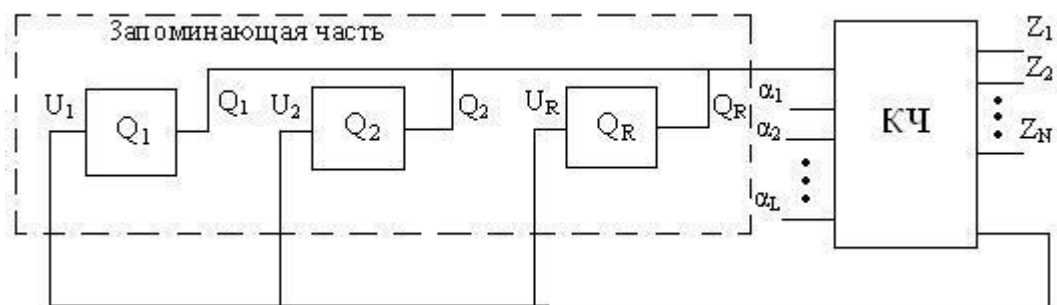


Рис.5

Каждое состояние абстрактного автомата  $a_m (m=0,1,\dots,M)$  кодируется в структурном автомате набором состояний элементарных автоматов  $(Q_1, Q_2, \dots, Q_k)$ , где  $R = \lceil \log_2(M+1) \rceil$ ;  $\lceil a \rceil$  означает ближайшее целое число, большее  $a$  или равное  $a$ , (если  $a$  целое). Каждый входной  $x_f, f=1,2,\dots,F$  и выходной  $y_g, g=1,2,\dots,G$  сигналы абстрактного автомата кодируются наборами значений двоичных сигналов на  $L$ , входных  $(\alpha_1, \alpha_2, \dots, \alpha_L)$  и  $N$  выходных  $(Z_1, Z_2, \dots, Z_N)$  каналах структурного автомата. Здесь  $L = \lceil \log_2 F \rceil$ ;  $N = \lceil \log_2(G) \rceil$ .

Изменение состояний элементарных автоматов происходит под действием сигналов возбуждения  $u_{jr}$ , где  $j=1, 2, \dots, K_r$ ;  $r = 1, 2, \dots, R$ . В схеме рис.5 для простоты у каждого элементарного автомата показано по одному входу  $u_r$ .

После выбора элементов памяти и кодирования состояний, входных и выходных сигналов абстрактного автомата произво-

дится синтез комбинационной части структурного автомата, реализующей систему функций кодированных выходов  $Z_n$  и функций возбуждения элементарных автоматов  $u_{jr}$ . Здесь

$$Z_n(t) = Z_n[Q_1(t), Q_2(t), \dots, Q_R(t), \alpha_1(t), \alpha_2(t), \dots, \alpha_L(t)],$$

$$n=1, 2, \dots, N$$

$$u_{jr}(t) = u_{jr}[Q_1(t), Q_2(t), \dots, Q_R(t), \alpha_1(t), \alpha_2(t), \dots, \alpha_L(t)],$$

$$j=1, 2, \dots, K_r; r=1, 2, \dots, R.$$

### Типы элементарных автоматов

Роль элементарных автоматов Мура в схемах ЦВМ выполняют триггеры различных типов. Рассмотрим некоторые, часто используемые типы.

I. T-триггер (триггер со счетным входом). Условное обозначение, граф и отмеченная таблица переходов T-триггера представлены на рис.6. Триггер имеет два выхода; прямой Q и инверсный  $\bar{Q}$ . Из графа триггера видно, что для перевода триггера в противоположное состояние нужно на вход T подать единичный сигнал. Согласно таблице переходов T-триггер обладает полнотой переходов и выходов.

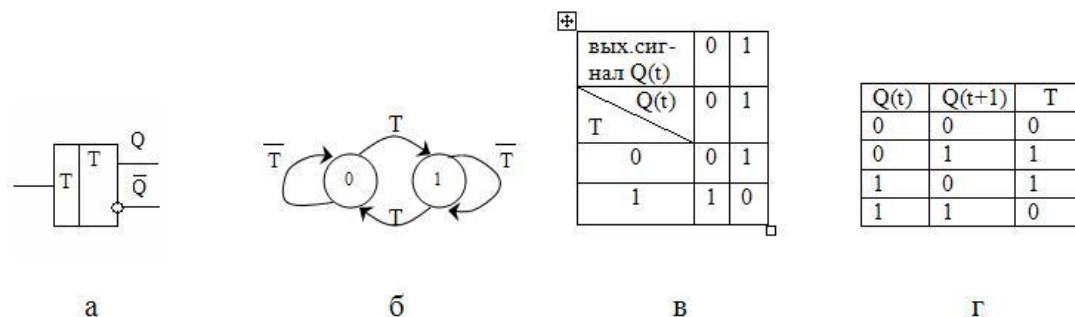


Рис.6

На практике более удобно вместо таблицы переходов использовать матрицу переходов триггера, которая определяет значения входных сигналов, обеспечивающие каждый из четырех возможных переходов в триггере ( $0 \rightarrow 0$ ,  $0 \rightarrow 1$ ,  $1 \rightarrow 0$ ,  $1 \rightarrow 1$ ). Матрица переходов T-триггера представлена на рис.6,г.

Из матрицы рис.6,г может быть получена аналитическая запись закона функционирования T-триггера:  $Q(t+1) = \overline{Q(t)}T \vee Q(t)\overline{T}$ . Это формула функции логического сложения по модулю 2: поэтому иначе T-триггер называют триггером со счетным входом.

2. RS-триггер (триггер с отдельными входами). Условное обозначение, граф, матрица переходов RS-триггера представлены на рис.7. Из графа следует, что переход из 0 в 0 может

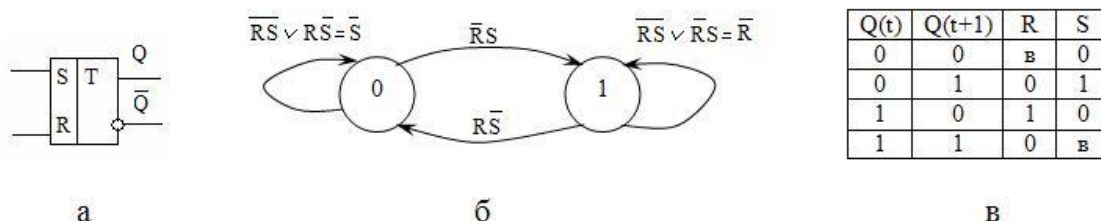


Рис.7

происходить при двух комбинациях входных сигналов:  $\overline{SR} \vee \overline{S}\overline{R} = \overline{S}$ , т.е. переход не зависит от значения R. Необходимо лишь, чтобы  $S = 0$ . Аналогично переход из 1 в 1 не зависит от значения, и при этом  $R = 0$ . Граф составлен с учетом того, что одновременная подача единичных сигналов на входы R и S запрещена ( $RS=0$ ). В матрице переходов коэффициентами  $b_i$  обозначены безразличные значения соответствующих сигналов ( $b_i \in \{0,1\}$ )

3. Триггер с дублированными переходами (рис.8). В отличие от RS-триггера в рассматриваемом триггере переходы из 0 в 1 и из 1 в 0 могут также осуществляться при одновременном поступлении единичных входных сигналов, т.е. при  $R'S'=1$ .

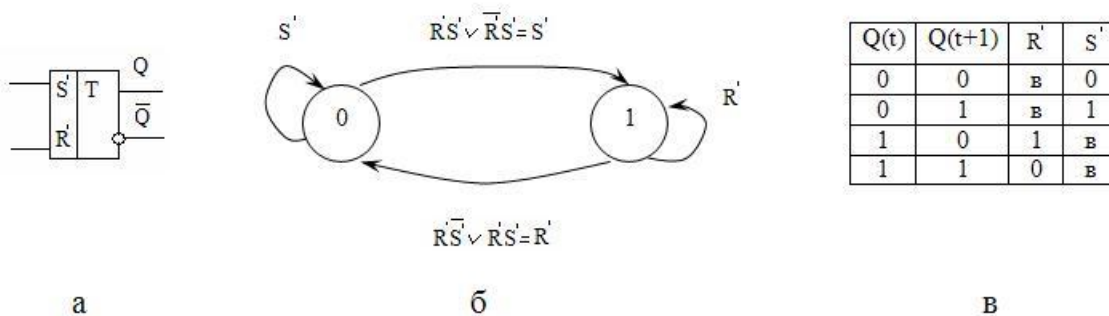


Рис.8

### Задания

#### для лабораторной работы №2

#### по теории автоматов (специальность 230101)

1. Синтезировать автомат на RS-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах:

0 – 2 – 4 – 6 – 7 – 10 – 8 – 9 – 1 – 15 – 13 – 14 – 11 – 12 – 3 – 0

2. Синтезировать автомат на RS-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах:

0 – 3 – 7 – 6 – 2 – 10 – 12 – 9 – 14 – 15 – 13 – 1 – 11 – 8 – 4 – 5 – 0

3. Синтезировать автомат на RS-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах:

0 – 4 – 7 – 5 – 2 – 11 – 8 – 9 – 12 – 13 – 15 – 14 – 1 – 10 – 3 – 6 – 0

4. Синтезировать автомат на RS-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах:

0 – 5 – 4 – 2 – 10 – 8 – 9 – 1 – 15 – 13 – 14 – 11 – 6 – 7 – 12 – 3 – 0

5. Синтезировать автомат на RS-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах:



0 – 6 – 4 – 12 – 7 – 10 – 2 – 1 – 15 – 13 – 14 – 11 – 8 – 9 – 3 – 0

6. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 7 – 12 – 10 – 4 – 6 – 13 – 8 – 9 – 1 – 15 – 14 – 11 – 5 – 3 – 0

7. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 9 – 4 – 6 – 8 – 7 – 10 – 12 – 1 – 15 – 14 – 11 – 13 – 2 – 3 – 0

8. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 1 – 10 – 8 – 4 – 6 – 12 – 15 – 13 – 7 – 9 – 5 – 14 – 2 – 11 – 3 – 0

9. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 2 – 5 – 8 – 9 – 14 – 10 – 15 – 6 – 7 – 13 – 11 – 1 – 12 – 3 – 0

10. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 7 – 2 – 4 – 6 – 10 – 5 – 15 – 13 – 8 – 9 – 1 – 14 – 11 – 3 – 0

11. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 2 – 6 – 7 – 4 – 10 – 8 – 9 – 1 – 15 – 13 – 14 – 11 – 5 – 3 – 0

12. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 3 – 8 – 9 – 2 – 10 – 5 – 11 – 6 – 15 – 13 – 4 – 1 – 7 – 12 – 14 – 0

13. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 3 – 5 – 4 – 7 – 6 – 11 – 15 – 2 – 8 – 14 – 1 – 9 – 12 – 10 – 13 – 0

14. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 4 – 3 – 7 – 1 – 14 – 15 – 13 – 8 – 9 – 11 – 5 – 10 – 2 – 12 – 6 – 0

15. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 14 – 2 – 13 – 6 – 7 – 10 – 4 – 9 – 1 – 15 – 3 – 8 – 11 – 12 – 5 – 0

16. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 13 – 7 – 6 – 2 – 10 – 12 – 8 – 14 – 15 – 3 – 1 – 11 – 9 – 4 – 5 – 0

17. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 12 – 7 – 1 – 2 – 11 – 8 – 9 – 15– 13 – 4 – 14 – 10 – 5 – 3 – 6 – 0

18. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 11 – 4 – 5 – 12 – 8 – 9 – 1 – 14 – 13 – 15 – 2 – 6 – 7 –10 – 3 – 0

19. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 10 – 4 – 12 – 7 – 13 – 2 – 1 – 15 – 6 – 14 – 11 – 5 – 9 – 8 – 0

20. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 9 – 12 – 10 – 4 – 6 – 8 – 5 – 1 – 15 – 13– 14 – 11 – 7 – 3 – 2 –0

21. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 8 – 4 – 6 – 3– 7 – 10 – 15 – 1 – 11 – 13 – 14 – 12 – 2 – 9 – 0

22. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 7 – 10 – 1 – 4 – 6 – 12 – 15 – 13 – 11 – 9 – 5– 14 – 2 – 8 – 3 – 0

23. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 6 – 5 – 2 – 13 – 14 – 10 – 15 – 9 – 7 – 4 – 11 – 1 – 12 – 3 – 0

24. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 5 – 2 – 7 – 6 – 10 – 11 – 15 – 13 –8 – 9 – 1 – 4 – 12 – 3 – 0

25. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 4 – 15 – 6 – 12 – 10 – 8 – 9 – 1 – 7 – 2 – 13 – 14 – 11 – 5 – 3 – 0

26. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 15 – 8 – 9 – 5 – 11 – 12 – 10 – 6 – 3 – 13 – 4 – 1 – 7 – 2 – 14 – 0

27. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 13 – 5 – 6 – 7 – 10 – 11 – 12 – 2 – 8 – 14 – 1 – 9 – 3 – 15 – 4 – 0

28. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 2 – 1 – 4 – 7 – 12 – 10 – 8 – 13 – 14 – 6 – 11 – 9 – 15 – 3 – 0

29. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 3 – 11 – 6 – 12 – 9 – 2 – 10 – 8 – 4 – 14 – 15 – 13 – 1 – 7 – 5 – 0

30. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 4 – 5 – 2 – 7 – 11 – 8 – 12 – 9 – 13 – 15 – 1 – 10 – 14 – 3 – 6 – 0

31. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 5 – 2 – 7 – 10 – 4 – 8 – 9 – 13 – 12 – 14 – 1 – 15 – 11 – 6 – 3 – 0

32. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 6 – 12 – 3 – 10 – 2 – 15 – 13 – 1 – 4 – 14 – 11 – 8 – 7 – 9 – 0

33. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 7 – 10 – 4 – 14 – 12 – 6 – 9 – 1 – 8 – 15 – 13 – 3 – 11 – 5 – 0

34. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 9 – 2 – 3 – 8 – 10 – 14 – 11 – 4 – 6 – 12 – 1 – 15 – 7 – 13 – 0

35. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 1 – 2 – 11 – 3 – 4 – 6 – 10 – 8 – 12 – 7 – 9 – 5 – 15 – 13 – 14 – 0

36. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 2 – 10 – 15 – 1 – 12 – 9 – 14 – 6 – 7 – 13 – 11 – 5 – 8 – 3 – 0

37. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 7 – 6 – 13 – 8 – 10 – 2 – 4 – 5 – 15 – 14 – 11 – 9 – 1 – 3 – 0

38. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 2 – 10 – 6 – 7 – 11 – 3 – 9 – 1 – 4 – 8 – 15 – 5 – 13 – 14 – 0

39. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 3 – 4 – 1 – 2 – 10 – 5 – 12 – 14 – 8 – 9 – 11 – 13 – 7 – 6 – 15 – 0

40. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 3 – 7 – 1 – 9 – 6 – 5 – 4 – 11 – 10 – 13 – 8 – 14 – 15 – 2 – 12 – 0

41. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 4 – 5 – 3 – 7 – 1 – 14 – 8 – 10 – 11 – 12 – 15 – 13 – 6 – 2 – 9 – 0

42. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 14 – 6 – 12 – 5 – 2 – 13 – 4 – 7 – 10 – 9 – 3 – 8 – 1 – 15 – 11 – 0

43. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 13 – 2 – 7 – 6 – 10 – 14 – 12 – 8 – 15 – 3 – 4 – 1 – 11 – 9 – 5 – 0

44. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 12 – 1 – 11 – 10 – 5 – 9 – 15 – 2 – 13 – 7 – 8 – 4 – 14 – 3 – 6 – 0

45. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 11 – 12 – 4 – 5 – 8 – 3 – 14 – 9 – 1 – 6 – 7 – 10 – 13 – 15 – 2 – 0

46. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 10 – 7 – 4 – 13 – 2 – 11 – 5 – 9 – 12 – 1 – 14 – 8 – 15 – 6 – 0

47. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 9 – 12 – 10 – 4 – 6 – 8 – 5 – 1 – 15 – 13 – 14 – 11 – 7 – 3 – 2 – 0

48. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 8 – 3 – 12 – 2 – 4 – 6 – 7 – 1 – 11 – 13 – 10 – 15 – 14 – 9 – 0

49. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 7 – 4 – 10 – 1 – 6 – 8 – 3 – 12 – 11 – 9 – 5 – 14 – 2 – 15 – 13 – 0

50. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 4 – 5 – 1 – 3 – 7 – 14 – 8 – 9 – 15 – 13 – 11 – 2 – 12 – 10 – 6 – 0

51. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 3 – 9 – 12 – 2 – 10 – 4 – 14 – 11 – 6 – 8 – 15 – 7 – 5 – 13 – 1 – 0

52. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 4 – 7 – 11 – 8 – 3 – 6 – 12 – 5 – 2 – 15 – 1 – 10 – 9 – 13 – 14 – 0

53. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 5 – 4 – 2 – 9 – 13 – 7 – 10 – 8 – 12 – 11 – 6 – 15 – 3 – 14 – 1 – 0

54. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 6 – 10 – 14 – 5 – 11 – 2 – 12 – 3 – 7 – 9 – 1 – 4 – 8 – 15 – 13 – 0

55. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 7 – 14 – 12 – 13 – 3 – 6 – 8 – 15 – 10 – 11 – 5 – 9 – 1 – 4 – 0

56. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 9 – 8 – 11 – 2 – 3 – 6 – 7 – 13 – 12 – 1 – 4 – 15 – 10 – 14 – 0

57. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 1 – 3 – 2 – 8 – 7 – 9 – 11 – 4 – 6 – 15 – 13 – 12 – 10 – 5 – 14 – 0

58. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 2 – 1 – 12 – 11 – 5 – 8 – 15 – 14 – 10 – 4 – 7 – 13 – 3 – 6 – 9 – 0

59. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 7 – 8 – 6 – 13 – 2 – 4 – 5 – 15 – 10 – 14 – 1 – 3 – 11 – 9 – 0

60. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 2 – 7 – 10 – 6 – 3 – 9 – 11 – 4 – 8 – 15 – 5 – 1 – 13 – 14 – 12 – 0

61. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 3 – 2 – 10 – 5 – 7 – 6 – 12 – 9 – 4 – 1 – 11 – 13 – 15 – 14 – 8 – 0

62. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 3 – 9 – 6 – 15 – 2 – 10 – 13 – 8 – 14 – 5 – 4 – 12 – 7 – 1 – 11 – 0

63. Синтезировать автомат на RS-триггерах и элементах ИЛИ–НЕ. Схему построить на JK- триггерах:

0 – 4 – 10 – 11 – 7 – 1 – 13 – 6 – 14 – 12 – 15 – 2 – 8 – 5 – 3 – 9 – 0

64. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 8 – 5 – 6 – 12 – 4 – 2 – 13 – 7 – 3 – 10 – 9 – 14 – 11 – 1 – 15 – 0

65. Синтезировать автомат на RS-триггерах и элементах И–НЕ. Схему построить на JK- триггерах:

0 – 13 – 6 – 2 – 7 – 12 – 8 – 10 – 4 – 1 – 15 – 9 – 5 – 3 – 11 – 14 – 0

## Задания на лабораторную работу № 2

1-3	a0 a1 a2 a3	4-6.	a0 a1 a2 a3	7-9.	a0 a1 a2 a3
x1	a3 a2 a0 a1		a2 a0 a3 a0		a1 a2 a3 a0
x2	a2 a3 a1 a3		a3 a1 a0 a1		a1 a3 a0 a3

a0 a1 a2 a3	a0 a1 a2 a3
y2 y3 y0 y1	y1 y0 y3 y2
y0 y3 y2 y0	y3 y2 y0 y1

10-12 .	13-15	16-18
a0 a1 a2 a3	a0 a1 a2 a3	a0 a1 a2 a3
x1 a1 a3 a0 a3	a0 a3 a1 a0	a1 a1 a0 a2
x2 a2 a0 a3 a1	a1 a0 a3 a2	a3 a2 a1 a0
	y2 y3 y1 y0	
	y1 y2 y0 y1	

19-21.	22-24	25-27
a0 a1 a2	a0 a1 a2	a0 a1 a2
x1 a1 a2 a0	a2 a1 a0	a0 a2 a1
x2 a2 a0 a1	a1 a0 a2	a1 a2 a2
x1 y1 y0 y2		y1 y0 y1
x2 y2 y1 y0		y0 y1 y2

Например: 13-15 (13,14,15)

Первая цифра- триггеры R-S

Вторая цифра- триггеры T

Третья цифра- триггеры J-K

Выходные сигналы автоматов Мура назначаются самим студентом.

**\* Два вида заданий представлены для удобства преподавателя.**

## **Порядок выполнения работы**

1. Произвести кодирование состояний, входных и выходных сигналов абстрактного автомата, заданного графическим или табличным способом.
2. Составить кодированную таблицу переходов, таблицу кодированных выходов и таблицу функций возбуждения триггеров заданного типа.
3. Получить минимальные выражения для функций кодированных выходов и функций возбуждения триггеров в булевом базисе.
4. Полученные выражения перевести в заданный универсальный базис.
5. Построить синхронизированную схему автомата в заданном базисе.
6. Набрать на эмуляторе схему автомата и проверить его работоспособность.
7. Проверить на эмуляторе работу автомата в соответствии с таблицами переходов и выходов.

## **Пример выполнения задания на лабораторную работу**

Рассмотрим процедуру структурного синтеза на примере построения (на Т-триггерах и элементах И-НЕ) автомата Мили, таблицы переходов и выходов которого приведены на рис.2,и,с соответственно. Здесь  $(M+1)=3$ ,  $F=2$ ,  $G=3$ .



1. Определяются минимально необходимые количества триггеров  $R$ , физических входов  $L$  и выходов  $N$  структурного автомата:  $R = \lceil \log_2(M+1) \rceil = 2$ ;  $L = \lceil \log_2 F \rceil = 1$ ;  $N = \lceil \log_2 G \rceil = 2$ .

Как видно, в автомате требуется два триггера, состояния и выходные сигналы которых обозначим через  $Q_1$  и  $Q_2$ ; в автомате должен быть один вход  $\alpha$  и два выхода  $Z_1$  и  $Z_2$ . 2. Составляются таблицы кодирования состояний, входных и выходных сигналов абстрактного автомата. Один из возможных вариантов кодирования приведен в табл. 1 - 3.

	$Q_1$	$Q_2$
$a_0$	0	0
$a_1$	0	1
$a_2$	1	0

Таблица 1

	$\alpha$
$x_1$	0
$x_2$	1

Таблица 2

	$Z_1$	$Z_2$
$y_1$	0	0
$y_2$	0	1
$y_3$	1	0

Таблица 3

3. С помощью полученных таблиц переходов и выходов (рис. 2, а, б) строится кодированная таблица переходов и таблица кодированных выходов.

Кодированная таблица переходов определяет зависимость состояний триггеров в момент времени  $(t + 1)$  от состояний триггеров и входных сигналов автомата в момент времени  $t$ . В табл. 4 кодированную таблицу переходов составляют первые пять столбцов. Комбинация единичных состояний триггеров является запрещённой, поэтому в соответствующих клетках таблицы стоят прочерки [1].

Таблица кодированных выходов определяет зависимость значений выходных сигналов автомата в момент времени  $t$  от состояний триггеров и входных сигналов автомата в тот же момент

времени. В табл.4 ее составляют столбцы 1,2,3,6,7. В клетках 6 и 7-го столбцов, соответствующих запрещенной комбинации состояний триггеров, помещены коэффициента  $b_i$ , значения которых безразличны ( $b_i \in \{0,1\}$ ).

Таблица 4

1	2	3	4	5	6	7	8	9
$\alpha(t)$	$Q_1(t)$	$Q_2(t)$	$Q_1(t+1)$	$Q_2(t+1)$	$Z_1(t)$	$Z_2(t)$	$T_1(t)$	$T_2(t)$
0	0	0	0	1	0	0	0	1
0	0	1	0	1	0	1	0	0
0	1	0	0	1	0	1	1	1
0	1	1	-	-	в	в	в	в
1	0	0	1	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	0	0	1	1	0
1	1	1	-	-	в	в	в	в

4. Строится таблица функций возбуждения триггеров, которая определяет значения сигналов на входах триггеров, обеспечивающих необходимые переходы автомата из одного состояния в другое (табл.4, столбцы 8 и 9). При этом используется матрица переходов триггера и ранее построенная кодированная таблица переходов (матрица переходов Т-триггера представлена на рис.6,2). Например, в первой строке табл.4  $T_1=0$ ,  $T_2=1$ , так как здесь первый триггер переходит из 0 в 0, а второй – из 0 в 1 и т.д.

5. Проводится синтез комбинационной части автомата, реализующей систему функций возбуждения триггеров и функций кодированных выходов автомата. В данном случае реализуется система из четырех функций:

$$Z_1(t) = f_1[\alpha(t), Q_1(t), Q_2(t)]; Z_2(t) = f_2[\alpha(t), Q_1(t), Q_2(t)];$$

$$T_2(t) = f_3[\alpha(t), Q_1(t), Q_2(t)]; T_2(t) = f_4[\alpha(t), Q_1(t), Q_2(t)];$$

Для минимизации этих функций воспользуемся диаграммами Вейча. Согласно заданию комбинационная часть автомата строится на элементах И-НЕ. Примем также для определенности, что комбинационные схемы должны быть двухуровневые. Отсюда следует, что мы должны исходить из минимальных дизъюнктивных нормальных форм (МДНФ) записи синтезируемых функций [3].

Диаграммы Вейча для функций  $Z_1, Z_2, T_1, T_2$ , составленные на основе 1-3-го и 6-9-го столбцов табл.4, представлены на рис.10. Из диаграмм можно получить следующие МДНФ синтезируемых функций (коэффициентам  $b_i$ , в диаграмме приписываются такие значения, чтобы упростить выражение для минимизируемой функции):  $Z_1 = \alpha \bar{Q}_1; Z_2 = Q_1 \vee \alpha \bar{Q}_2; T_1 = Q_1 \vee \alpha; T_2 = \alpha Q_2 \vee \alpha \bar{Q}_2$ .

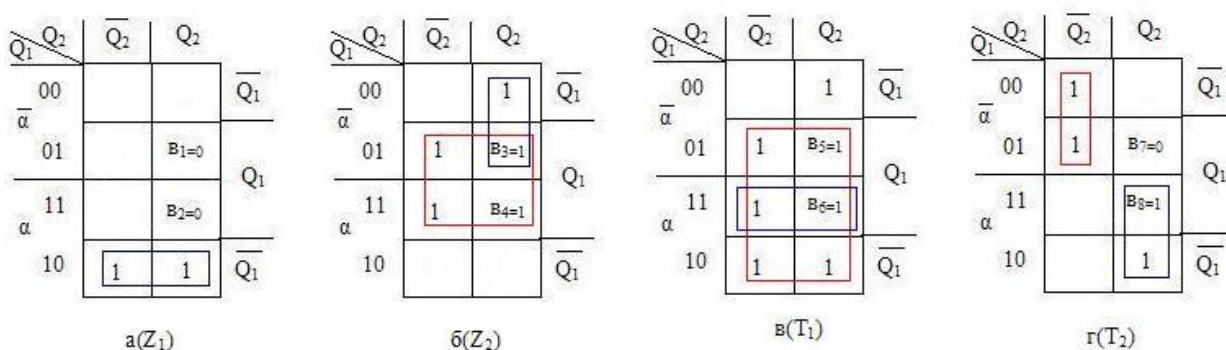


Рис.10

6. Для отметки моментов дискретного времени  $t_0, 1, 2, \dots, n, \dots$  в схему автомата подаются синхронизирующие сигналы  $S$ , следующие через равные интервалы времени  $T$ ;  $T$  определяет такт работы устройства (рис.11,а). Синхронизированная структурная схема автомата на  $T$ -триггерах приведена на рис.11,б.

В течение такта  $T$  формируются выходные сигналы автомата  $Z_n(t) = \lambda[\alpha(t), x(t)]$  (по сигналу  $C$ ), и триггеры ЗЧ одновременно переключаются в состояния, соответствующие очередному состоянию автомата  $0, (t + 1)$  (по сигналу  $\bar{C}$ ). Как видно на входах всех триггеров ЗЧ поставлены двухвходовые элементы И. На практике триггеры часто выполняются в синхронном варианте (синхронные триггеры), когда упомянутые элементы И включаются в схемы триггеров. Число входов у каждого из таких триггеров увеличивается на единицу; добавляется вход  $C$  для синхронизирующих сигналов.

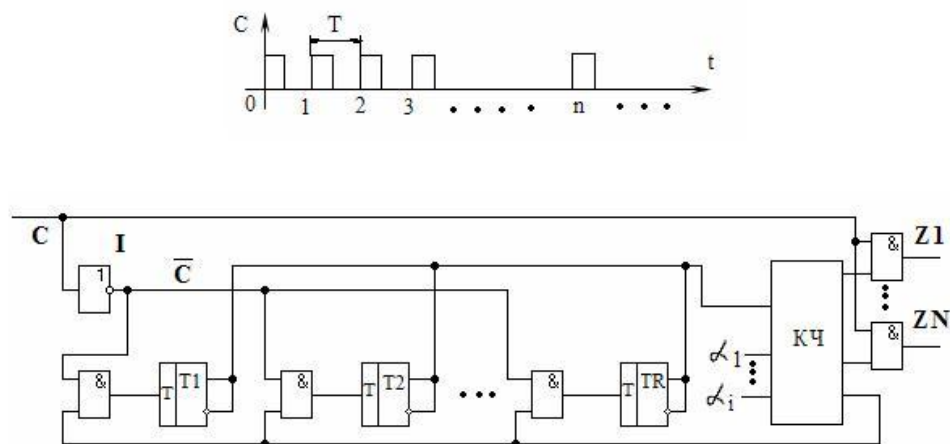


Рис.11

7. Обеспечивается устойчивость состояний и устраняется эффект гонок в автомате. Понятие устойчивости состояний автомата заключается в следующем.

Пусть на графе автомата имеется два перехода ( $a_m \rightarrow a_s$  и  $a_s \rightarrow a_k$ ), выполняемые под действием одного и того же сигнала  $x_f$ . Если длительность сигнала  $\bar{C}$  (рис.11,б) больше времени перехода автомата из  $a_m$  в  $a_s$ , то автомат в данном такте может

перескочить состояние  $a_s$  и к моменту  $(t+I)$  оказаться в состоянии  $a_k$ . Состояние  $a_s$  в этом случае будет неустойчивым.

Другой неприятный момент связан с тем, что триггеры ЗЧ имеют, как правило, различные времена переключения, различны также времена задержек сигналов обратной связи, поступающих от выходов триггеров на их входы (через КЧ). Поэтому, если при переходе автомата из  $a_m$  в  $a_s$  должны измениться состояния нескольких триггеров, то между их выходными сигналами начинаются состязания (гонки). Тот триггер, который перейдет в новое состояние раньше других, может по цепям обратной связи изменить сигналы на входах других триггеров до того, как они переключатся в новые состояния. В результате триггеры могут оказаться в иных состояниях, чем это требуется по графу автомата.

Устойчивость состояний будет обеспечена, и эффект гонок устранен, если выходные сигналы триггеров, фиксирующие их текущие состояния  $Q_r(t)$ , не будут меняться во время действия синхросигнала на их входах. Этим свойством обладают двухступенчатые триггеры, которые переключаются в новые состояния в момент окончания синхросигнала. Например, условное обозначение и логически эквивалентная схема двухступенчатого Т - триггера приведены на рис.12. Как видно, в состав двухступенчатого триггера входят два триггера – I и II. Во время действия сигнала С происходит лишь запись информация в триггер; с момента окончания сигнала С состояние триггера переписывается в триггер. Аналогично строятся двухступенчатые триггеры других типов. При использовании двухступенчатых триггеров в схеме рис.11,б отпадает необходимость в инверторе; синхросигналы

подаются непосредственно на входы синхронизации триггеров ЗЧ (см. далее рис.14).

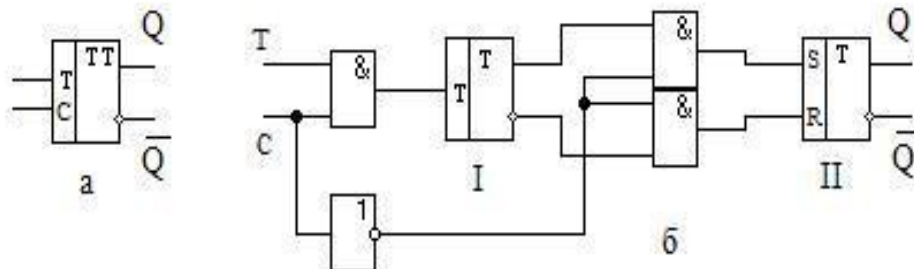


Рис.12

В лабораторном макете используются двухступенчатые JK-триггеры. Условное обозначение, логически эквивалентная схема и матрица переходов JK-триггера приведены на рис.13,а,б,в соответственно. Здесь J – вход синхронной установки триггера в "1", K- вход синхронной установки в "0". По сигналу JCS =1 триггер переключается в противоположное состояние.

JK-триггер называется универсальным, так как может выполнять функции других типов триггеров. Так, на рис.13,г представлены способы использования JK -триггера как T-триггера и как RS -триггера.

8. Для построения синхронизированной схемы автомата выражения для функций выходов автомата логически умножаются на символ синхросигнала C. В данном примере получаем:

$$Z_1 = C \bar{Q}_1; Z_2 = C(Q_1 \bar{Q}_2) = C Q_1 \bar{Q}_2$$

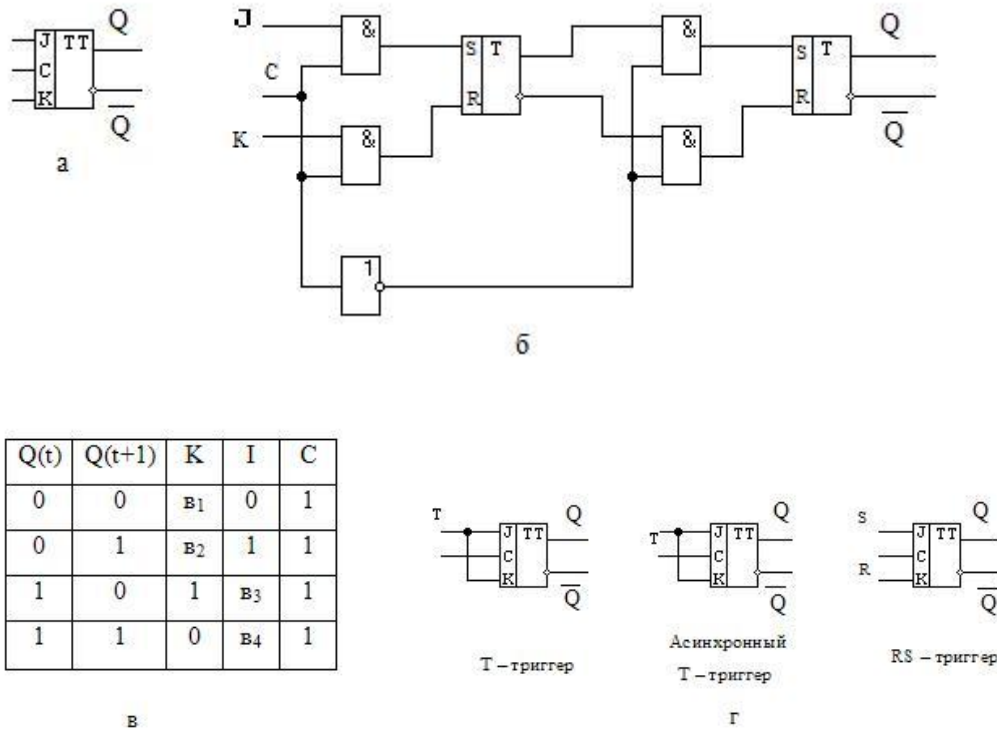


Рис.13

Если в синтезируемой схеме автомата используются синхронные триггеры, то полученные минимальные выражения для функций возбуждения триггеров на С не умножаются. В этом случае синхросигналы подаются непосредственно на входы синхронизации триггеров автомата.

Переведём далее полученные МНДФ для функций  $Z_1, Z_2, T_1, T_2$  в заданный базис И-НЕ;

$$\begin{aligned}
 Z_1 &= \overline{C \alpha Q_1} = \overline{I(C, \alpha, Q_1)}; T_1 = \overline{\alpha \nu Q_1} = \overline{\alpha \cdot Q_1} = \overline{I(\alpha, Q_1)}; \\
 Z_2 &= \overline{C Q_1 \nu C \alpha Q_2} = \overline{C Q_1 \nu C \alpha Q_2} = \overline{I(C, Q_1), I(C, I(\alpha, Q_2))}; \\
 T_2 &= \overline{\alpha Q_2 \nu \alpha Q_2} = \overline{\alpha Q_2 \cdot \alpha Q_2} = \overline{I(\alpha, Q_2), I(\alpha, Q_2)}.
 \end{aligned}$$

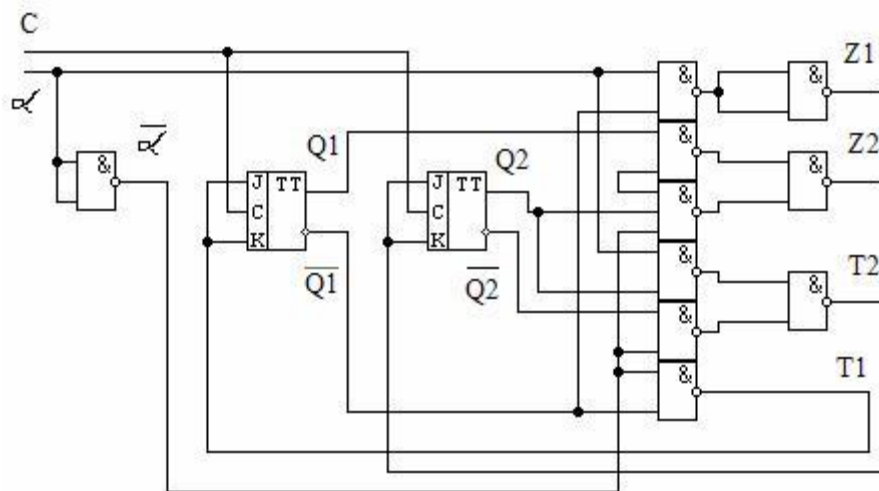


Рис.14

9. На основе полученных выражений строится схема автомата (рис.14). Здесь в качестве синхронных Г - триггеров использованы JK - триггеры.

### Содержание отчета

1. Граф и таблицы переходов и выходов абстрактного автомата.
2. Таблицы кодирования состояний, входных и выходных сигналов абстрактного автомата.
3. Кодированная таблица переходов, таблицы кодированных выходов и таблица функций возбуждения триггеров.
4. Получение минимальных выражений для функций кодированных выходов и функций возбуждения триггеров. Перевод выражений в заданный базис.
5. Схема автомата на эмуляторе, работающая в заданном базисе.



# Лабораторная работа №3

## *РЕГИСТРЫ. СЧЕТЧИКИ*

Цель работы: изучение схем построения регистров и счетчиков на универсальных триггерах и логических элементах.

Регистры Регистр - запоминающее устройство, предназначенное для хранения одного  $n$ -разрядного двоичного числа, представляет собой совокупность  $n$  триггеров, каждый из которых служит для запоминания одного двоичного разряда, и вспомогательных схем, обеспечивающих, в частности, выполнение операций по вводу и выводу (съему) информации из регистра. В зависимости от способа ввода и съема информации различают параллельные и последовательные регистры.

В параллельных регистрах все разряды числа вводятся, а также выдаются одновременно. Регистры могут быть построены на триггерах любого типа: синхронных и асинхронных, одноступенчатых и двухступенчатых. На рис.3.1 представлена схема параллельного регистра на JK-триггерах (используемых в лабораторном макете) и логических элементах И-НЕ.

В схеме рис.3.1 запись числа в регистр осуществляется двухтактным способом с использованием асинхронных установочных входов R и S. В первом такте производится установка всех триггеров в нулевое состояние сигналом "Уст.0". В следующем такте по сигналу "Прием числа" прямой код числа, поступающего по кодовым шинам числа (КШЧ), подается на входы S триггеров. Выдача числа из регистра может производиться в прямом коде (по сигналу на линии I), в обратном коде (по сигнала-

лу на линии II) и в парафазном коде (путем подачи сигналов одновременно по линиям I и II).

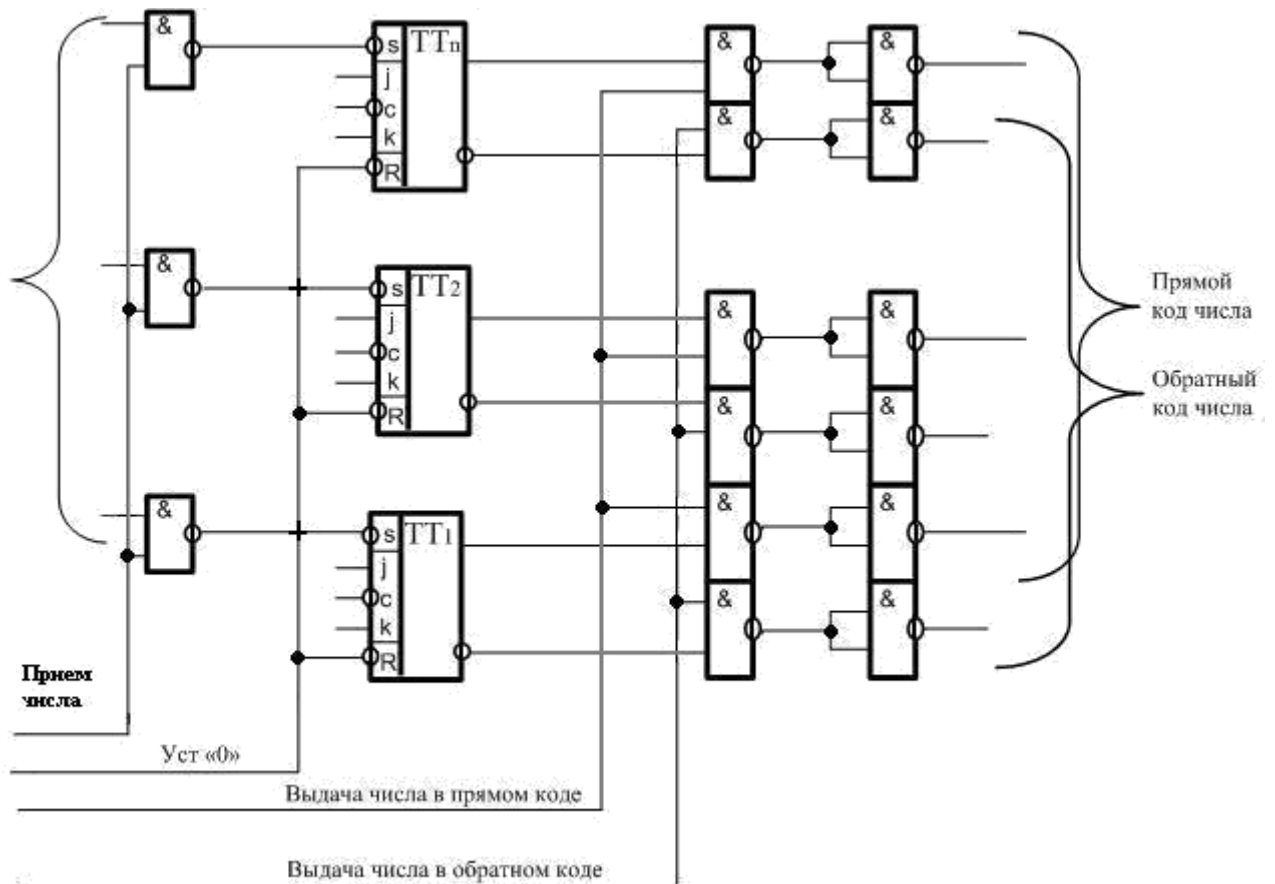


Рис.3.1

Последовательные регистры используются для ввода или съема двоичной информации, представляемой последовательным способом, т.е., в виде временной последовательности сигналов, которые соответствуют значениям цифр в двоичных разрядах и проходят по одному каналу в дискретные моменты времени, задаваемые синхронизирующими сигналами.

Схема трехразрядного последовательного регистра представлена на рис.3.2а. Регистр построен на двухступенчатых триггерах, соединенных друг с другом последовательно. Основу схем составляет сдвигающий регистр, в котором по оконча-

нии сигнала "Сдвиг" в каждый триггер заносится состояние предыдущего триггера, т.е. двоичный код, записанный в регистре, сдвигается вправо на один разряд. При этом в старший триггер записывается значение сигнала на линии  $\alpha$ , а выдвигаемый вправо младший разряд кода теряется.

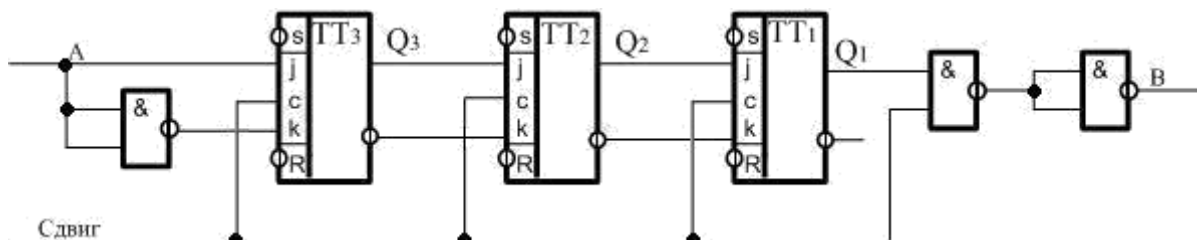


Рис.3.2а

В процессе ввода информации в  $n$ -разрядный регистр разряды вводимого числа (начиная с младшего) подаются последовательно на вход  $\alpha$ , а в качестве сдвигающих сигналов используются синхронизирующие сигналы. По окончании  $n$ -го сдвигающего сигнала число окажется записанным в триггерах регистра. Вывод информации из регистра последовательным способом осуществляется путем подачи сдвигающих сигналов. При этом на выходе  $\beta$  формируется последовательный код числа, ранее записанного в регистре.

### Счетчики

В схемах ЦВМ счетчики используются для подсчета количества сигналов, формирования последовательных чисел и деления частоты. В потенциальной системе элементов счетчики строятся на асинхронных или синхронных двухступенчатых Т-триггерах, количество которых определяется максимальной разрядностью хранимых чисел. Счетчик на  $n$  триггерах имеет  $2^n$  различных состояний, которым соответствуют двоичные числа от

0 до  $2^n - 1$ .

Таблица 3.1

		Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>
№ Вх сигн	Исх сост	0	0	0
	1		0	0
2		0	1	0
3		0	1	1
4		1	0	0
5		1	0	1
6		1	1	0
7		1	1	1
8		0	0	0

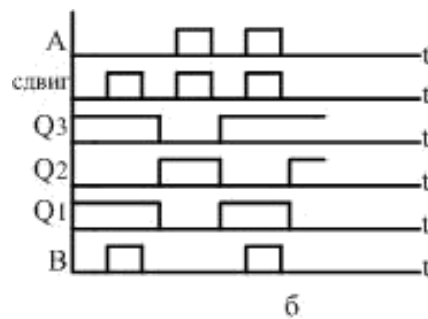


Рис.3.3б

Пусть в состав счетчика входят три триггера, состояния которых обозначаются через  $Q_3$ ,  $Q_2$ ,  $Q_1$ . При поступлении последовательности сигналов на вход счетчика состояния триггеров будут меняться согласно табл.3.1. Из таблицы следует, что в процессе счета младший триггер должен переключаться по каждому входному сигналу счета, а каждый последующий триггер должен менять состояние на противоположное, если предыдущий триггер переключается из 1 в 0. Первое условие реализуется подачей сигналов счета ( $\alpha_{сч}$ ) на вход Т младшего триггера счетчика. Так как двухступенчатые триггеры переключаются по спаду единичного входного сигнала, то для выполнения второго ус-

ловия нужно вход каждого последующего триггера связать цепью переноса с прямым выходом предыдущего триггера (рис.3.3.а ). На этом выходе будет отрицательный перепад напряжения при переходе триггера из 1 в 0.

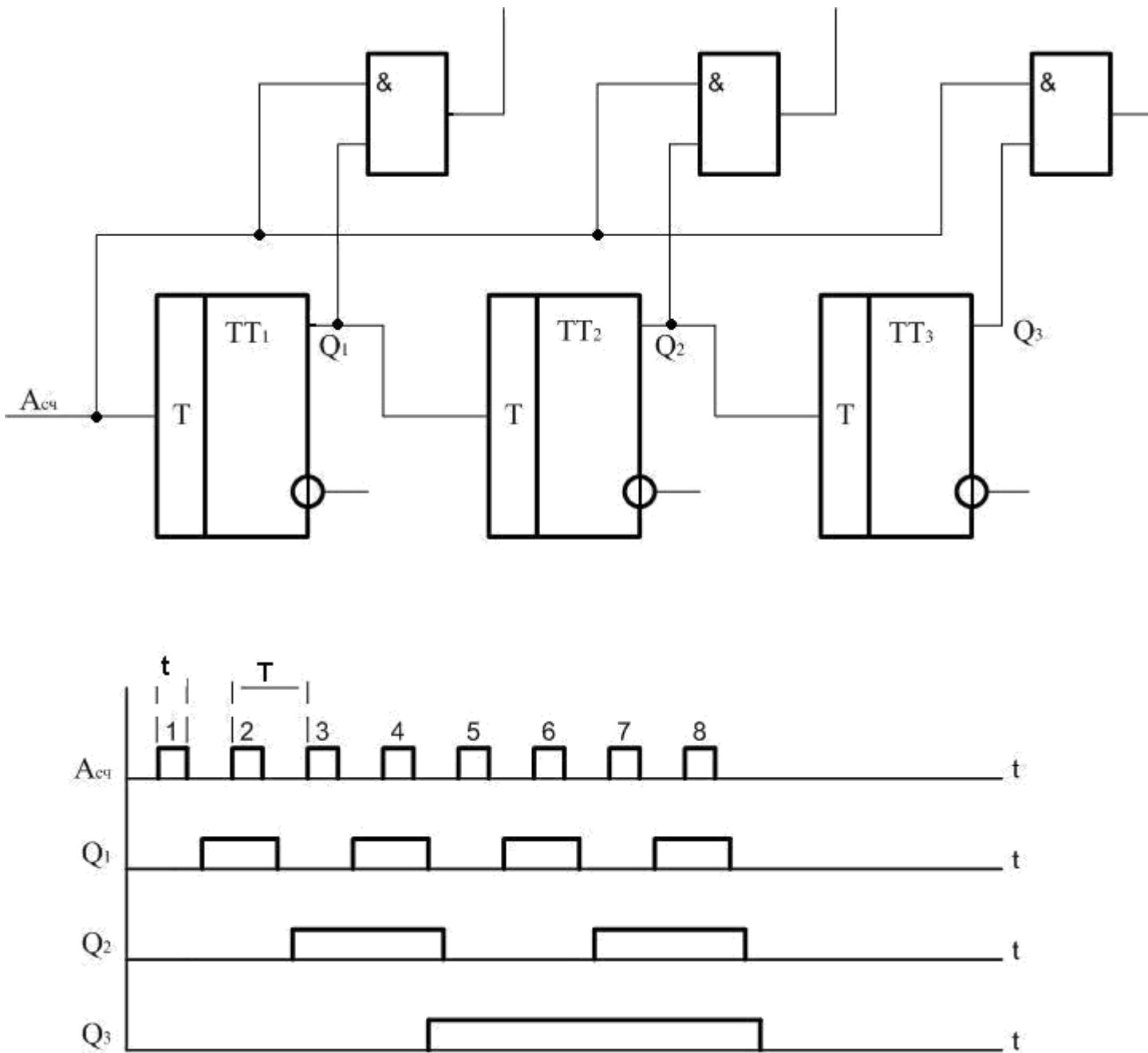


Рис.3.3

Работа трехразрядного суммирующего счетчика, представленного на рис.3, а, поясняется временной диаграммой на

рис.3/3,б. Согласно диаграмме триггеры в процессе счета переключаются последовательно друг за другом. Поэтому счетчик носит название счетчика с последовательным переносом. Максимальная частота работы счетчика с последовательным переносом определяется максимально допустимой частотой переключения его младшего разряда и составляет  $F_{сч}=1/T_{сч}=1/(t_{сч}+t_n)$ . Если числа, формируемые счетчиком, необходимо выдавать в параллельном коде (путем одновременного опроса всех триггеров), то частота  $F_{сч}$  оказывается значительно меньшей, так как очередной сигнал опроса может быть подан лишь по окончании переходных процессов во всех триггерах счетчика. В схеме рис.3.3.а в качестве сигналов опроса используются сигналы счета, и минимальный период  $F_{сч}=(t_{сч}+nt_n)$ .

Таблица 3.2

		Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>
№ Вх. сигнала	Исх. сост.	1	1	1
	1	1	1	0
	2	1	0	1
	3	1	0	0
	4	0	1	1
	5	0	1	0
	6	0	0	1
	7	0	0	0
	8	1	1	1

Наряду с суммирующими счетчиками применяют вычитающие счетчики, осуществляющие «обратный» счет. Из таблицы последовательных состояний вычитающего счетчика (табл.3.2) следует, что в процессе обратного счета младший триггер переключается по каждому входному сигналу счета, а каждый последующий триггер меняет состояние на противоположное, если предыдущий триггер переходит из 0 в 1. Поэтому для полу-

чения вычитающего счетчика нужно в схеме суммирующего счетчика связать вход каждого последующего триггера не с прямым, а с обратным выходом предыдущего триггера (рис.3.4).

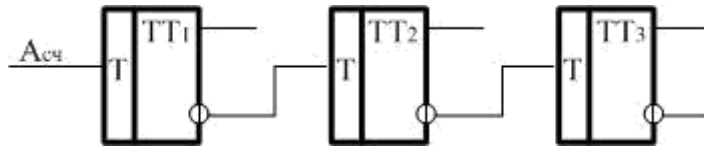


Рис.3.4

Счетчик, который может работать как в режиме суммирования, так и вычитания, называется реверсивным. На рис.5 представлена схема реверсивного счетчика со стороны младших разрядов. Здесь в состав схемы входят управляющие линии I и II, задающие режим работы. Если единичный потенциал присутствует на линии I, то триггеры связаны, как в суммирующем счетчике. Если единичный потенциал подан на линию II, то схема реализует обратный счет.

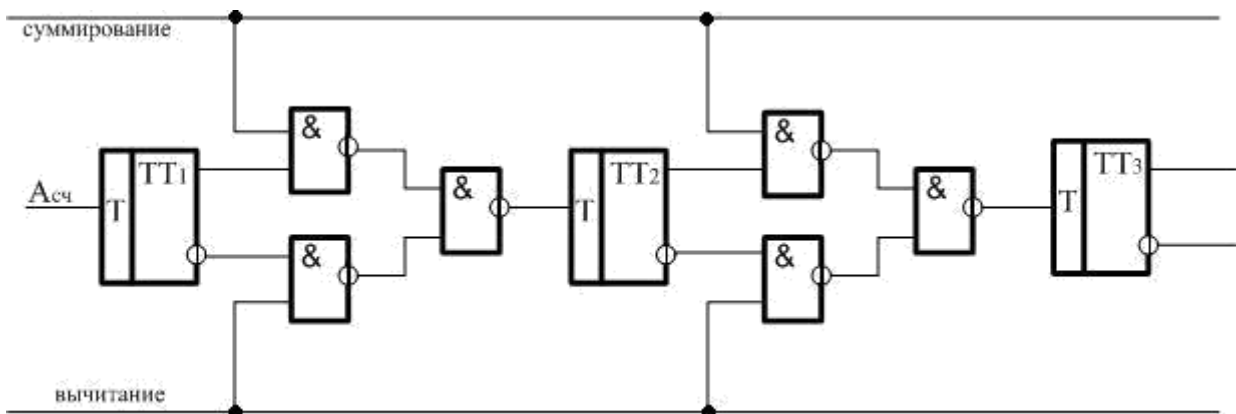


Рис.5

Кроме рассмотренных, широко применяются более быстродействующие схемы счетчиков со сквозным, групповым и одновременным переносом, характеризующиеся усложненными цепями связи между триггерами.

## ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Составить принципиальные схемы соединений на эмуляторе для исследования заданных преподавателем видов регистров и счетчиков.

2. Собрать на эмуляторе схемы заданных видов регистров и счетчиков. Для каждой из собранных схем:

- провести проверку правильности работы в режиме одиночного запуска, используя кнопку "Такт", и в динамическом режиме, используя сигналы от ГТИ;

- определить допустимые пределы изменения длительности синхронизирующих сигналов (или сигналов счета);

3. Объяснить полученные результаты.

### ***Примеры построения схем, собираемых на эмуляторе***

#### Счетчики с одновременным, сквозным и групповым переносом.

При построении схем счетчиков можно использовать методику структурного синтеза конечных автоматов [3]. Построим, например, кодированную таблицу переходов и таблицу функций возбуждения триггеров трехразрядного суммирующего счетчика (табл.3). В табл.3 строки, соответствующие отсутствию входного сигнала счета ( $\alpha_{сч}=0$ ), можно не приводить. При  $\alpha_{сч}=0$  состояние счетчика не меняется и, следовательно, сигналы возбуждения триггеров  $T_1$ ,  $T_2$ ,  $T_3$ , должны быть равны 0. Из табл.3 с помощью, например, диаграмм Вейча можно получить следующие выражения для входных сигналов триггеров:  $T_1=\alpha_{сч}$ ;  $T_2=\alpha_{сч}Q_1$ ;  $T_3=\alpha_{сч}Q_1 \cdot Q_2$ .



Таблица 3.3

	t			t+1			t		
Асч	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	T <sub>3</sub>	T <sub>2</sub>	T <sub>1</sub>
1	0	0	0	0	0	1	0	0	1
1	0	0	1	0	1	0	0	1	1
1	0	1	0	0	1	1	0	0	1
1	0	1	1	1	0	0	1	1	1
1	1	0	0	1	0	1	0	0	1
1	1	0	1	1	1	0	0	1	1
1	1	1	0	1	1	1	0	0	1
1	1	1	1	0	0	0	1	1	1

В общем случае для n-разрядного счетчика можно записать:

$$T_1 = \alpha_{сч}; \quad T_i = \alpha_{сч} Q_1 \cdot Q_2 \cdot \dots \cdot Q_{i-1}, \quad i = 2 \div n \quad (3.1)$$

или в преобразованном виде  $T_1 = \alpha_{сч}; \quad T_i = T_{i-1} Q_{i-1}; \quad i = 2 \div n \quad (3.2)$

Так как в выражения для входов всех триггеров счетчика входит сомножителем  $\alpha_{сч}$ , то уравнения (1) и (2) удобно реализовывать на синхронных Т-триггерах, подавая сигналы счета непосредственно на входы синхронизации всех триггеров. При этом вместо уравнений (1) и (2) следует использовать соответственно уравнения:

$$T_1 = 1; \quad T_i = Q_1 \cdot Q_2 \cdot \dots \cdot Q_{i-1}, \quad i = 2 \div n \quad (3.3)$$

$$T_1=1; \quad T_i=T_{i-1}Q_{i-1}; \quad i=2 \div n. \quad (3.4)$$

Счетчик, построенный по уравнениям (3.4), носит название счетчика со сквозным переносом. Схема  $n$ -разрядного счетчика со сквозным переносом представлена на рис.3.7. Максимальная частота работы такого счетчика равна  $F_{сч}=1/[t_{сч}+t_n+(n-1)\tau_0]$  где  $\tau_0$ - время задержки одного логического элемента.

Счетчик, построенный по уравнениям (3.3), называется счетчиком с одновременным (параллельным) переносом. Схема такого счетчика при  $n=4$  представлена на рис.3.8. Здесь  $F_{сч}=1/(t_{сч}+t_n+\tau_0)$ . Счетчик с одновременным переносом характеризуется наивысшим быстродействием, однако разрядность его ограничивается возможностями элементов схемы: нагрузочной способностью триггеров и коэффициентом объединения по входу И. В тех случаях, когда указанные ограничения не позволяют строить схему счетчика с заданным числом разрядов  $n$ , разряды счетчика разбивают на группы по  $m$  разрядов ( $m \geq 2$ ). В пределах каждой группы разряды организуются, как в счетчике с одновременным переносом, а группы связываются между собой цепями сквозного переноса.

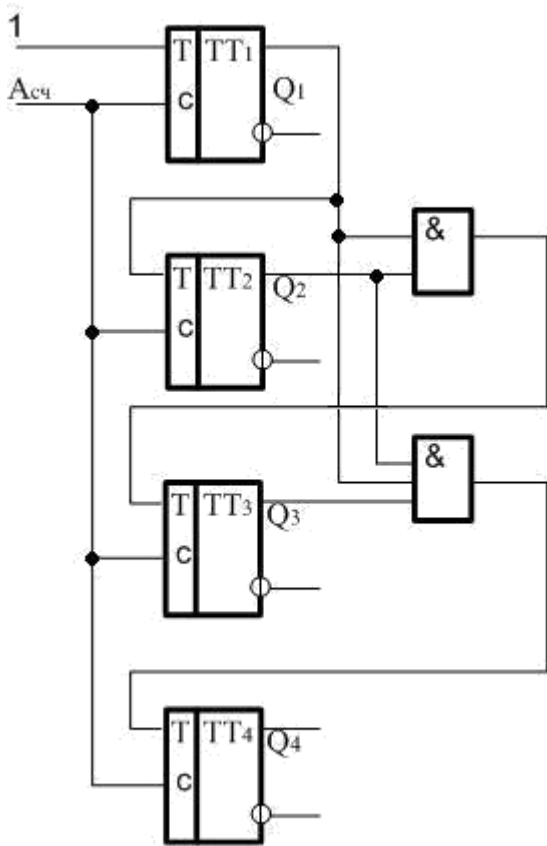
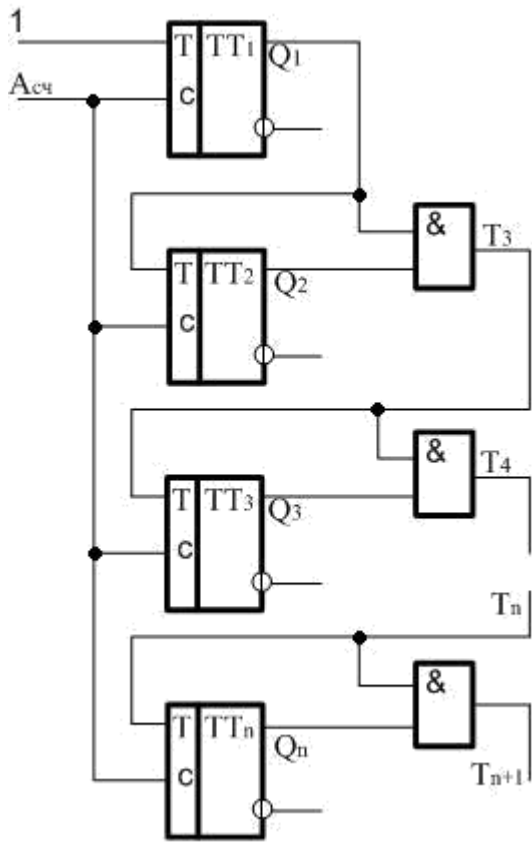


Рис.3.7

Рис.3.8

Обозначим через  $P_j$  - сигнал переноса на входе  $j$ -й группы разрядов счетчика.

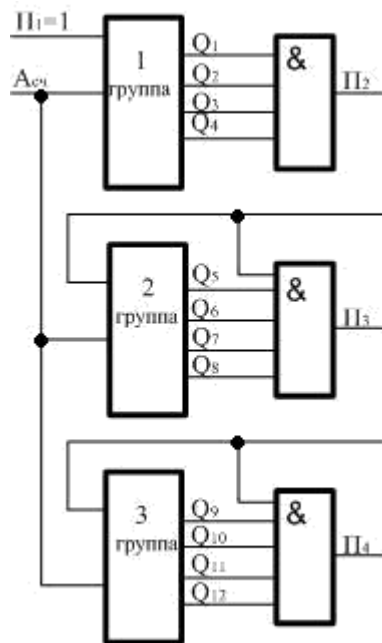


Рис.3.9

$$P_1=T_1=1; \quad P_{j+1}=P_j Q_{jm} Q_{j,m-1} \dots Q_{j,m-m+1} \quad j=1 \div ]n/m[$$

На рис.3.9 приведена структурная схема счетчика с групповым переносом при  $m=4$ . Здесь первая группа разрядов организована по схеме рис.3.8. В каждой последующей группе входной сигнал переноса подается через соответствующие схемы И на входы Т всех триггеров данной группы. Например, во второй группе:

$$T_5=P_2; \quad T_6=P_2 Q_5; \quad T_7=P_2 Q_5 Q_6; \quad T_8=P_2 Q_5 Q_6 Q_7.$$

В счетчике с групповым переносом  $F_{сч}=1/[t_{сч}+t_n+n/m[\tau_0]$ .

Счетчики с коэффициентом пересчета, не равным целой степени двух Коэффициент пересчета  $K_n$  определяет число со-

стояний счетчика или, иначе, максимальное число входных сигналов, которое может быть сосчитано счетчиком. Например, при  $K_n = 5$  счетчик имеет пять состояний и каждый пятый входной сигнал счета переводит его в исходное состояние. Рассмотренные ранее  $n$ -разрядные счетчики имеют  $K_n = 2^n$

Счетчик с  $K_n \neq 2^n$ , где  $n$  - целое, можно построить путем введения в схему обратных связей, обеспечивающих переход суммирующего счетчика из состояния, соответствующего двоичному числу  $(K_n - 1)$ , в исходное нулевое состояние. При этом число разрядов счетчика  $n = \lceil \log_2 K_n \rceil$

Рассмотрим пример синтеза десятичного счетчика ( $K_n = 10$ ), предназначенного для деления частоты на 10. В состав счетчика входят четыре JK-триггера ( $Q_3, Q_2, Q_1, Q_0$ ). Входные сигналы счета поступают по линии  $\alpha_{сч}$ ; каждый десятый сигнал должен появляться на выходе  $\beta$ .

Таблица 4

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
t					t+1				t							
Асч	Q <sub>4</sub>	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>4</sub>	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	B	J <sub>4</sub>	K <sub>4</sub>	J <sub>3</sub>	K <sub>3</sub>	J <sub>2</sub>	K <sub>2</sub>	J <sub>1</sub>
1	0	0	0	0	0	0	0	1	0	0	B	0	B	0	B	1
1	0	0	0	1	0	0	1	0	0	0	B	0	B	1	B	B
1	0	0	1	0	0	0	1	1	0	0	B	0	B	B	0	1
1	0	0	1	1	0	1	0	0	0	0	B	1	B	B	1	B
1	0	1	0	0	0	1	0	1	0	0	B	B	0	0	B	1
1	0	1	0	1	0	1	1	0	0	0	B	B	0	1	B	B
1	0	1	1	0	0	1	1	1	0	0	B	B	0	B	0	1
1	0	1	1	1	1	0	0	0	0	1	B	B	1	B	1	B
1	1	0	0	0	1	0	0	1	0	B	0	0	B	0	B	1
1	1	0	0	1	0	0	0	0	1	B	1	0	B	0	B	B

В табл.3.4 совмещены кодированная таблица переходов (столбцы 1 - 9), таблица кодированных выходов (столбцы 1-5, 10) и таблица Функций возбуждения триггеров (столбцы 1 - 5, 11-18) счетчика с  $K_n = 10$ . Коэффициенты в в таблице обозначают безразличные значения соответствующих сигналов.

Используя, например, диаграммы Вейча, можно получить следующие выражения для выхода  $\beta$  и входов триггеров:

$$\beta = \alpha_{сч} Q_3 Q_0; \quad J_0 = K_0 = \alpha_{сч}; \quad J_2 = K_2 = \alpha_{сч} Q_0 Q_3;$$

$$J_3 = K_3 = \alpha_{сч} Q_0 Q_1; \quad J_3 = \alpha_{сч} Q_2 Q_1 Q_0; \quad K_4 = \alpha_{сч} Q_0;$$

Схема синхронного десятичного счетчика, построенного в соответствии с этими выражениями, представлена на рис.10. Здесь сигналы  $\alpha_{сч}$  подаются непосредственно на входы синхронизации всех триггеров, и следовательно, выражения для входов триггеров преобразуется к виду

$$J_0=K_0=1; J_1=K_1=Q_3Q_1; J_3=K_3=Q_1Q_0; J_4=Q_2Q_1Q_0; K_3=Q_0.$$

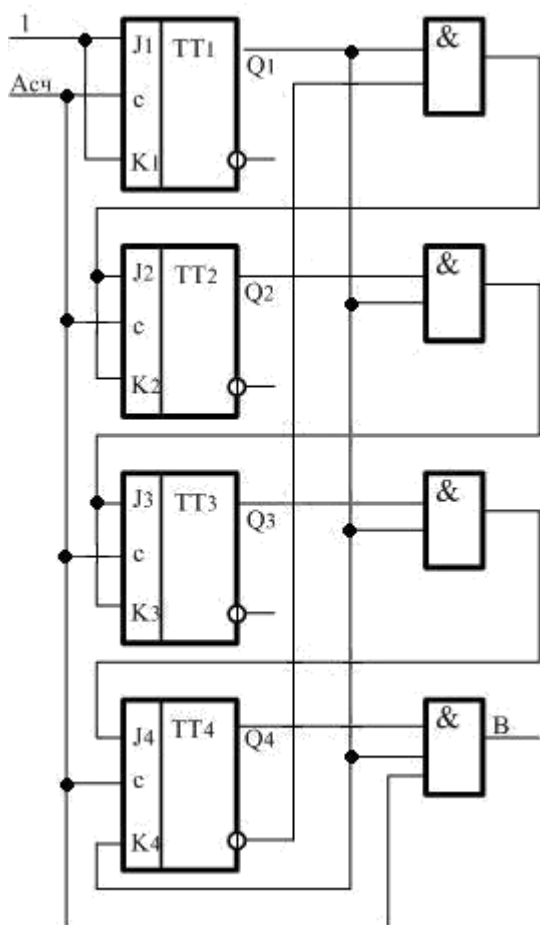


Рис.3.10

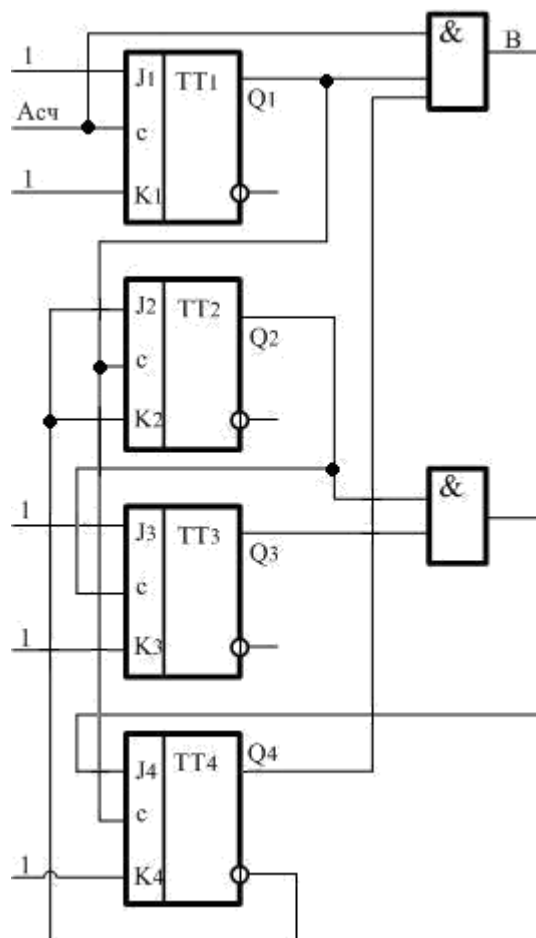


Рис.3.11

Схема асинхронного десятичного счетчика на JK-триггерах представлена на рис.3.11.

### Задания на лабораторную работу № 3

#### Счетчик с пересчетом

№	Коэффициент пересчета	Тип триггера
1.	9	JK – тр.
2.	9	RS - тр.
3.	9	T - тр.
4.	10	JK – тр.
5.	10	RS - тр.
6.	10	T - тр.

7.	11	JK – тр.
8.	11	RS - тр.
9.	11	T - тр.
10.	12	JK – тр.
11.	12	RS - тр.
12.	12	T - тр.
13.	13	JK – тр.
14.	13	RS - тр.
15.	13	T - тр.
16.	14	JK – тр.
17.	14	RS - тр.
18.	14	T - тр.
19.	15	JK – тр.
20.	15	RS - тр.
21.	15	T - тр.
22.	17	JK – тр.
23.	17	RS - тр.
24.	17	T - тр.
25.	18	JK – тр.
26.	18	RS - тр.
27.	18	T - тр.
28.	19	JK – тр.
29.	19	RS - тр.
30.	19	T - тр.
31.	20	JK – тр.
32.	20	RS - тр.
33.	20	T - тр.
34.	21	JK – тр.
35.	21	RS - тр.
36.	21	T - тр.

**Задания  
на лабораторную работу №3 по теории автоматов  
для специальности 230101**

1. Синтезировать вычитающий счетчик  $K_{CT} = 9$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
2. Синтезировать вычитающий счетчик  $K_{CT} = 11$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
3. Синтезировать вычитающий счетчик  $K_{CT} = 12$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.



4. Синтезировать вычитающий счетчик  $K_{CT} = 13$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
5. Синтезировать вычитающий счетчик  $K_{CT} = 14$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
6. Синтезировать вычитающий счетчик  $K_{CT} = 15$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
7. Синтезировать вычитающий счетчик  $K_{CT} = 9$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
8. Синтезировать вычитающий счетчик  $K_{CT} = 11$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
9. Синтезировать вычитающий счетчик  $K_{CT} = 12$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
10. Синтезировать вычитающий счетчик  $K_{CT} = 13$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
11. Синтезировать вычитающий счетчик  $K_{CT} = 14$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
12. Синтезировать вычитающий счетчик  $K_{CT} = 15$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
13. Синтезировать суммирующий счетчик  $K_{CT} = 9$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
14. Синтезировать суммирующий счетчик  $K_{CT} = 11$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
15. Синтезировать суммирующий счетчик  $K_{CT} = 12$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
16. Синтезировать суммирующий счетчик  $K_{CT} = 13$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
17. Синтезировать суммирующий счетчик  $K_{CT} = 14$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
18. Синтезировать суммирующий счетчик  $K_{CT} = 15$  на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.

19. Синтезировать суммирующий счетчик  $K_{СТ} = 9$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
20. Синтезировать суммирующий счетчик  $K_{СТ} = 11$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
21. Синтезировать суммирующий счетчик  $K_{СТ} = 12$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
22. Синтезировать суммирующий счетчик  $K_{СТ} = 13$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
23. Синтезировать суммирующий счетчик  $K_{СТ} = 14$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
24. Синтезировать суммирующий счетчик  $K_{СТ} = 15$  на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
25. Синтезировать реверсивный счетчик  $K_{СТ} = 5$  при сложении и  $K_{СТ} = 5$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах.
26. Синтезировать реверсивный счетчик  $K_{СТ} = 6$  при сложении и  $K_{СТ} = 6$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
27. Синтезировать реверсивный счетчик  $K_{СТ} = 7$  при сложении и  $K_{СТ} = 7$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
28. Синтезировать реверсивный счетчик  $K_{СТ} = 5$  при сложении и  $K_{СТ} = 6$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
29. Синтезировать реверсивный счетчик  $K_{СТ} = 6$  при сложении и  $K_{СТ} = 5$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах

30. Синтезировать реверсивный счетчик  $K_{CT} = 5$  при сложении и  $K_{CT} = 7$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
31. Синтезировать реверсивный счетчик  $K_{CT} = 7$  при сложении и  $K_{CT} = 5$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
32. Синтезировать реверсивный счетчик  $K_{CT} = 6$  при сложении и  $K_{CT} = 7$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
33. Синтезировать реверсивный счетчик  $K_{CT} = 7$  при сложении и  $K_{CT} = 6$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
34. Синтезировать реверсивный счетчик  $K_{CT} = 5$  при сложении и  $K_{CT} = 5$  при вычитании на Т-триггерах и элементах И-НЕ. Схему построить на JK- триггерах.
35. Синтезировать реверсивный счетчик  $K_{CT} = 6$  при сложении и  $K_{CT} = 6$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
36. Синтезировать реверсивный счетчик  $K_{CT} = 7$  при сложении и  $K_{CT} = 7$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
37. Синтезировать реверсивный счетчик  $K_{CT} = 5$  при сложении и  $K_{CT} = 6$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
38. Синтезировать реверсивный счетчик  $K_{CT} = 6$  при сложении и  $K_{CT} = 5$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
39. Синтезировать реверсивный счетчик  $K_{CT} = 5$  при сложении и  $K_{CT} = 7$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах
40. Синтезировать реверсивный счетчик  $K_{CT} = 7$  при сложении и  $K_{CT} = 5$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах

41. Синтезировать реверсивный счетчик  $K_{СТ} = 6$  при сложении и  $K_{СТ} = 7$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах

42. Синтезировать реверсивный счетчик  $K_{СТ} = 7$  при сложении и  $K_{СТ} = 6$  при вычитании на Т-триггерах и элементах ИЛИ-НЕ. Схему построить на JK- триггерах

**\*Два вида заданий представлены для удобства преподавателя.**

Кодовые кольца Особую группу составляют счетчики, работающие по принципу циклического сдвигающего регистра (кодовые кольца). Простую схему кодового кольца можно получить из схемы сдвигающего регистра на рис.3а, если выходы триггера  $ТТ_0$  связать со входами триггера  $ТТ_2$  ( $J_2 = Q_0$ ,  $K_2 = Q_0$ ). Схема такого кодового кольца и таблица состояний представлены на рис.12, а и б соответственно. Из таблицы состояний следует, что подобное кодовое кольцо можно рассматривать как счетчик с  $K_n = 2^n$ , где  $n$  - число разрядов кольца. Для схемы рис. 12,а  $K_n = 6$ , так как каждый шестой по счету входной сигнал переводит схему в исходное нулевое состояние.

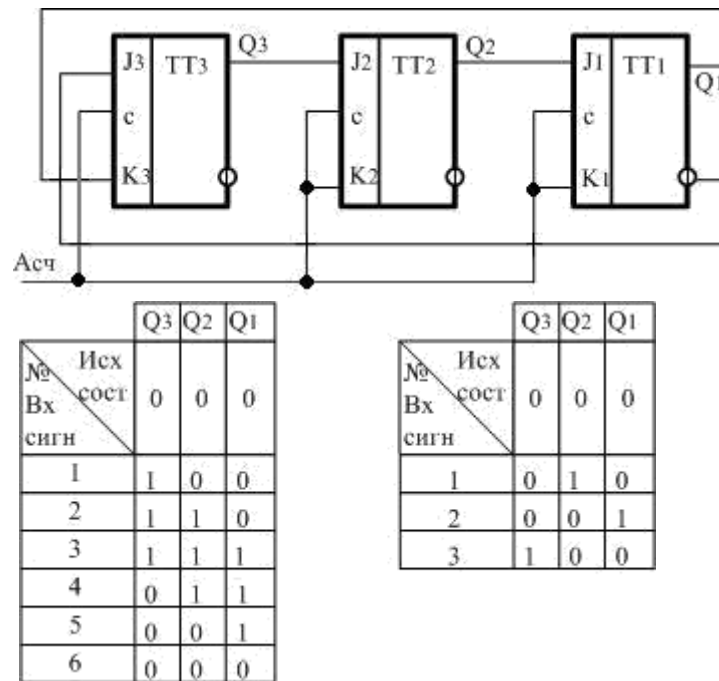


Рис.3.12

Другой вариант кольца можно получить, если в схеме рис.3.12,а переключить входы триггера  $ТТ_2$  ( $J_2 = Q_0$ ,  $K_2 = Q_0$ ) и в исходном состоянии занести единицу в один из триггеров (например, в  $ТТ_2$ ). В этом случае коэффициент пересчета схемы будет равен  $K_n = n = 3$  (рис.3.12,в). Различные варианты кодовых колец подробно рассматриваются в [2].

### Содержание отчета

1. Логические схемы исследуемых устройств,
2. Принципиальные схемы для исследуемых устройств в эмуляторе.
3. Временные диаграммы по всем пунктам программы.
4. Анализ полученных результатов и краткие выводы.

# Лабораторная работа №4

## *ДВОИЧНЫЕ СУММАТОРЫ*

Цель работы: изучение схем построения одnorазрядных и многоразрядных двоичных сумматоров на универсальных триггерах и логических элементах.

### Двоичные сумматоры

Двоичный сумматор является операционным узлом ЦВМ, предназначенным для арифметического суммирования двух  $m$ -разрядных двоичных чисел  $A$  и  $B$  с фиксированной запятой ( $A+B = S$ ).

Как правило, многоразрядные сумматоры ( $m>1$ ) строятся из одnorазрядных сумматоров (ОС), каждый из которых осуществляет двоичное суммирование одноименных разрядов слагаемых  $a_i$  и  $b_i$  и переноса из предыдущего разряда  $P_{i-1}$  ( $i=1 \div m$ ). На выходах ОС формируются значения данного разряда суммы  $S_i$  и переноса в следующий разряд  $P_i$ .

Одноразрядный комбинационный сумматор. Одноразрядный комбинационный сумматор (ОКС) представляет собой комбинационную схему с тремя входами ( $a_i, b_i, P_{i-1}$ ) и двумя выходами ( $S_i, P_i$ ). Процесс двоичного суммирования в ОКС описывается таблицей истинности для значений  $S_i, P_i$ , рассматриваемых как переключательные функции от  $a_i, b_i, P_{i-1}$  (рис.4.1.а).

С помощью, например, диаграмм Вейча (рис.1,б,в) можно получить следующие минимальные дизъюнктивные нормальные формы (БДНФ) для  $S_i$  и  $P_i$  :

$$S_i = \overline{\overline{a_i b_i}} P_{i-1} \vee \overline{\overline{a_i} \overline{b_i}} P_{i-1} \vee \overline{\overline{a_i} \overline{b_i}} P_{i-1} \vee \overline{\overline{a_i} \overline{b_i}} P_{i-1} = \overline{\overline{\overline{\overline{a_i b_i}} P_{i-1}}} \overline{\overline{\overline{\overline{a_i} \overline{b_i}} P_{i-1}}} \overline{\overline{\overline{\overline{a_i} \overline{b_i}} P_{i-1}}} \overline{\overline{\overline{\overline{a_i} \overline{b_i}} P_{i-1}}} \quad (4.1)$$

$$P_i = \overline{\overline{a_i b_i}} \vee \overline{\overline{a_i} \overline{b_i}} P_{i-1} \vee \overline{\overline{a_i} \overline{b_i}} P_{i-1} = \overline{\overline{a_i b_i}} \overline{\overline{a_i} \overline{b_i}} P_{i-1} \overline{\overline{a_i} \overline{b_i}} P_{i-1} \quad (4.2)$$

или

$$P_i = \overline{\overline{a_i b_i}} \vee (a_i \vee b_i) P_{i-1} = \overline{\overline{a_i b_i}} \overline{\overline{a_i} \overline{b_i}} P_{i-1} \quad (4.3)$$

a <sub>i</sub>	b <sub>i</sub>	P <sub>i-1</sub>	P <sub>i</sub>	S <sub>i</sub>
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

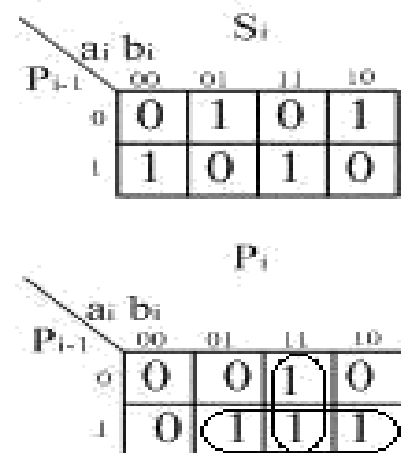


Рис.4.1

Схема ОКС, реализующая уравнения (4.1) и (4.2), представлена на рис.2. Введем следующие обозначения:  $\tau_s$  - время задержки образования суммы  $S_i$  в ОКС;  $\tau_p$  - время задержки распространения переноса через ОКС;  $\tau_0$  - среднее время задержки одного логического элемента;  $S_{вх}$  - суммарное число входов логических элементов в ОКС. Для ОКС на рис.4.2:  $\tau_s=3\tau_0$ ;  $\tau_p=2\tau_0$ ;  $S_{вх}=26$ . Здесь и далее предполагается, что разряды слагаемых  $a_i, b_i$  подаются в парафазном коде.

Рассматривая сумму  $S_i$  как функцию четырех переменных  $(a_i, b_i, P_{i-1}, P_i)$ , можно получить для  $S_i$  следующее более простое выражение:

$$S_i = a_i b_i P_{i-1} \vee a_i \ddot{P}_i \vee b_i \ddot{P}_i \vee P_{i-1} \ddot{P}_i = a_i b_i P_{i-1} \vee (a_i \vee b_i \vee P_{i-1}) \ddot{P}_i = \overline{a_i b_i P_{i-1}} \cdot \overline{\ddot{a}_i \ddot{b}_i \ddot{P}_{i-1} \ddot{P}_i} \quad (4.4)$$

На рис.4.3 представлена схема ОКС, построенная по уравнениям (4.4) и (4.3). Здесь  $\tau_s = 6\tau_0$ ;  $\tau_p = 2\tau_0$ ;  $S_{вх} = 19$ .



Рассмотрим еще два варианта ОКС. Выражения для суммы  $S_i$  (4.1) и переноса  $P_i$  (4.2) можно преобразовать к следующему виду:  $S_i = a_i \oplus b_i \oplus P_{i-1} = m_i \oplus P_{i-1}$ ; (4.5)

$$P_i = a_i b_i \vee (a_i \oplus b_i) P_{i-1} = a_i b_i \vee m_i P_{i-1} \quad (4.6)$$

где  $\oplus$  - знак логической операции сложения по модулю 2:

$$(\bar{x}y \vee x\bar{y} = x \oplus y); \quad m_i = a_i \oplus b_i = a_i \bar{b}_i \vee \bar{a}_i b_i .$$

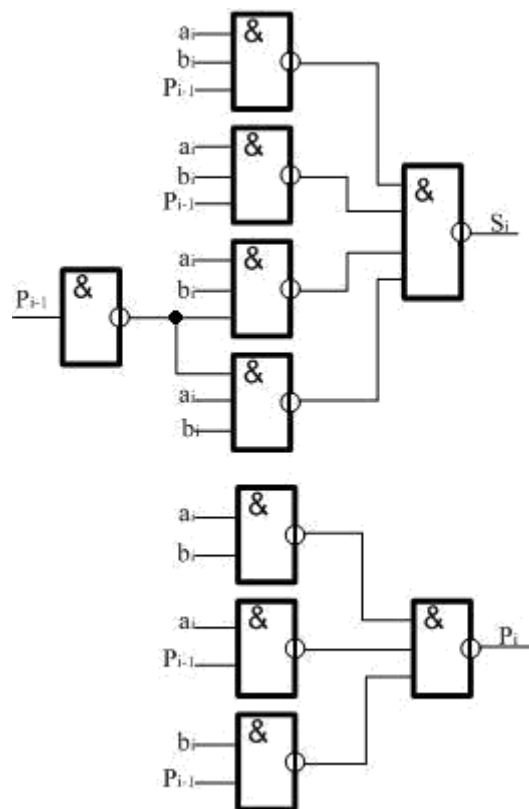


Рис.4.2

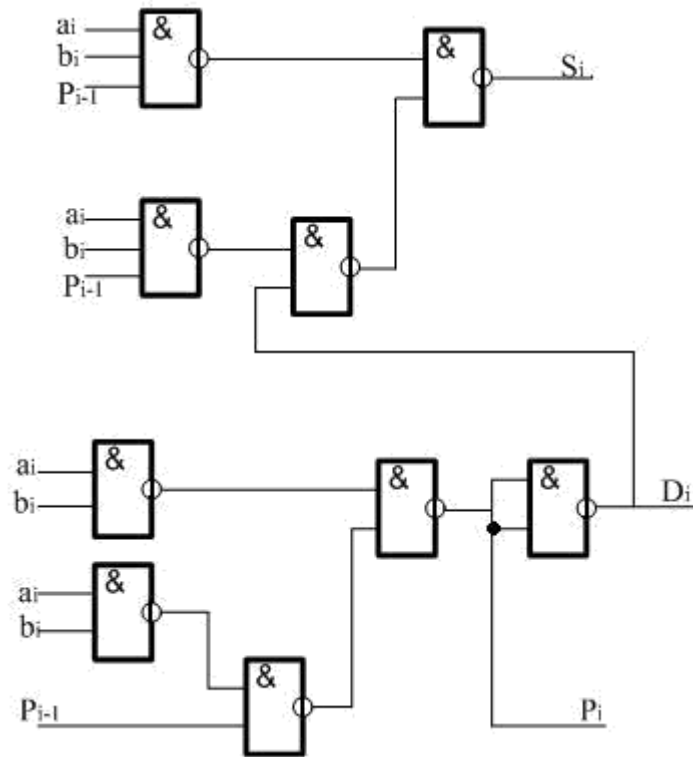


Рис.4.3

Схема ОКС, построенная на элементах И-НЕ в соответствии с уравнениями (4.5) и (4.6), представлена на рис.4.4. Здесь

$$m_i = a_i \oplus b_i = a_i \overline{\overline{b_i}} \vee \overline{\overline{a_i}} b_i = \overline{\overline{a_i \overline{b_i}}} \vee \overline{\overline{\overline{a_i} b_i}};$$

$$S_i = m_i \oplus P_{i-1} = m_i \overline{\overline{P_{i-1}}} \vee \overline{\overline{m_i}} P_{i-1} = \overline{\overline{m_i \overline{P_{i-1}}}} \vee \overline{\overline{\overline{m_i} P_{i-1}}} = \overline{\overline{m_i \overline{P_{i-1}}}} \vee \overline{\overline{\overline{m_i} P_{i-1}}};$$

$$P_i = a_i b_i \vee m_i P_{i-1} = \overline{\overline{a_i b_i}} \vee \overline{\overline{m_i P_{i-1}}}$$

Для ОКС на рис.4:  $\tau_s = 5\tau_0$ ;  $\tau_p = 2\tau_0$ ;  $S_{вх} = 18$ .

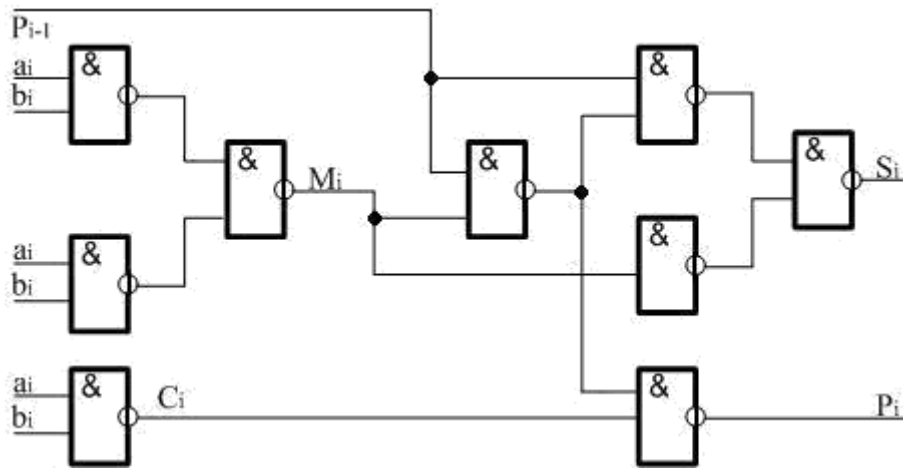


Рис.4.4

Для сокращения времени  $\tau_p$  можно применить схему, в которой перенос представляется в виде совокупности сигналов, передаваемых по нескольким линиям. Введем обозначения:  $C_i = a_i b_i$ ;  $T_i = a_i \vee b_i$ . Из диаграммы Вейча на рис.1,в получим МДНФ для отрицания переноса  $P_i$  :

$$P_i = \bar{a}_i \bar{b}_i \vee \bar{a}_i \bar{P}_{i-1} \vee \bar{b}_i \bar{P}_{i-1} \quad (4.7)$$

Из уравнения (2) , (5) и (7) следует:

$$S_i = \bar{C}_i T_i \bar{P}_{i-1} \vee (\bar{C}_i T_i P_{i-1}) P_{i-1};$$

$$P_i = T_i (\bar{C}_i T_i \bar{P}_{i-1});$$

$$\bar{P}_i = C_i (\bar{C}_i T_i P_{i-1}).$$

Обозначим  $U_i = \bar{C}_i T_i \bar{P}_{i-1}$ ;  $V_i = (\bar{C}_i T_i P_{i-1})$  тогда

$$S_i = U_i (\bar{V}_i P_{i-1}) \quad (4.8)$$

$$P_i = T_i U_i \quad (4.9)$$

$$\bar{P}_i = \bar{C}_i V_i \quad (4.10)$$

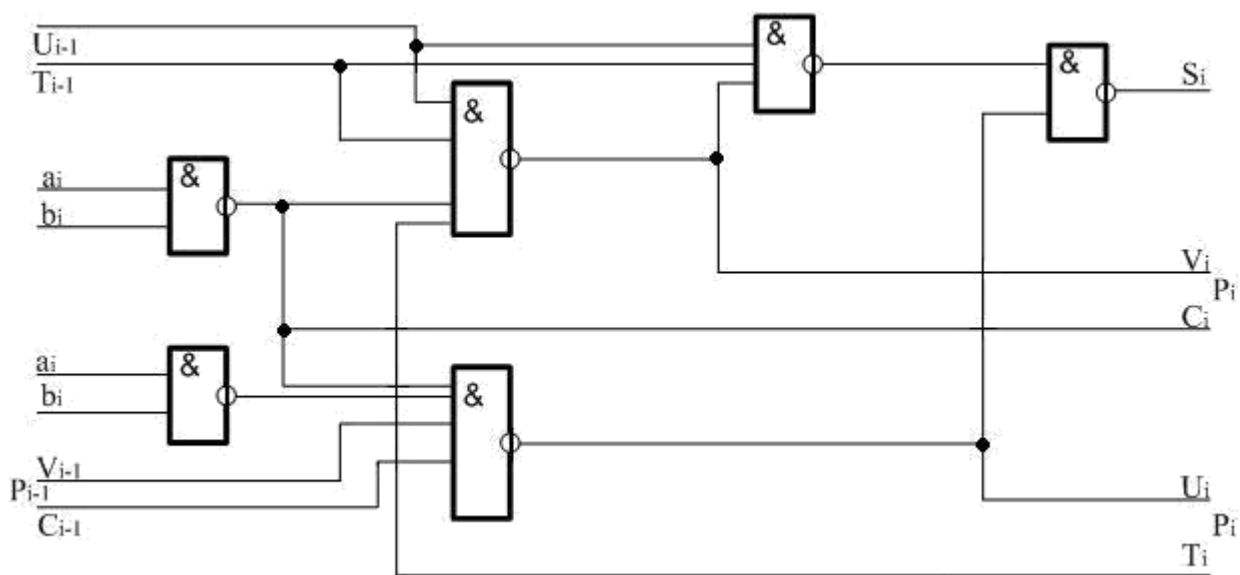


Рис.4.5

Схема ОКС, соответствующая уравнениям (4.8) - (4.10), приведена на рис.4.5. Здесь прямое и инверсное значение переносов представляются как конъюнкция пар сигналов, которые передаются отдельно. Формирование полных значений сигналов переноса осуществляется непосредственно на входе тех схем, на которые они поступают. В схеме рис.4.5  $\tau_p=2\tau_0$ ;  $\tau_s=4\tau_0$ ;  $S_{вх}=17$ . Недостатком схемы является большое количество связей для сигналов распространения переносов.

Одноразрядный накапливающий сумматор. Основу одноразрядного накапливающего сумматора (ОНС) составляет триггер со счетным входом (Т-триггер), реализующий операцию сложения по модулю 2. При построении ОНС выражения (4.1) и (4.2) для суммы  $S_i$  и переноса  $P_i$  преобразуются к виду

$$S_i = a_i \oplus b_i \oplus P_{i-1}; \tag{4.11}$$

$$P_i = \overline{(a_i \oplus b_i)} b_i \vee (a_i \oplus b_i) P_{i-1}; \tag{4.12}$$

Схема ОНС представлена на рис.6. Здесь первоначально триггер устанавливается в нулевое состояние сигналом «Уст.0», и в нем запоминается значение  $a_i$ . Далее процесс образования суммы  $S_i$  выполняется в два такта. В первом такте по управляющему сигналу  $y_1$  на счетный вход триггера подается значение  $b_i$ . При этом триггер реализует операцию сложения по модулю 2 ( $a_i \oplus b_i$ ). В этом же такте комбинационная часть ОНС формирует в соответствии с уравнениями (4.12) сигнал переноса в следующий разряд  $P_i$ . В следующем такте по сигналу  $y_2$  на счетный вход триггера подается значение  $P_{i-1}$ . При этом в соответствии с уравнением (4.11) в триггере образуется сумма  $S_i$ .

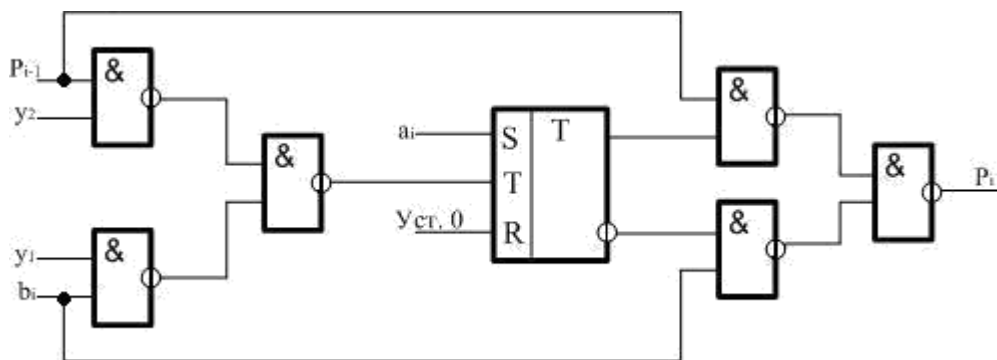


Рис.4.6

Многоразрядный накапливающий сумматор, построенный из ОНС, оказывается более простым, чем сумматор из ОКС, однако в нем требуется два такта на образование суммы, что увеличивает время выполнения операций сложения.

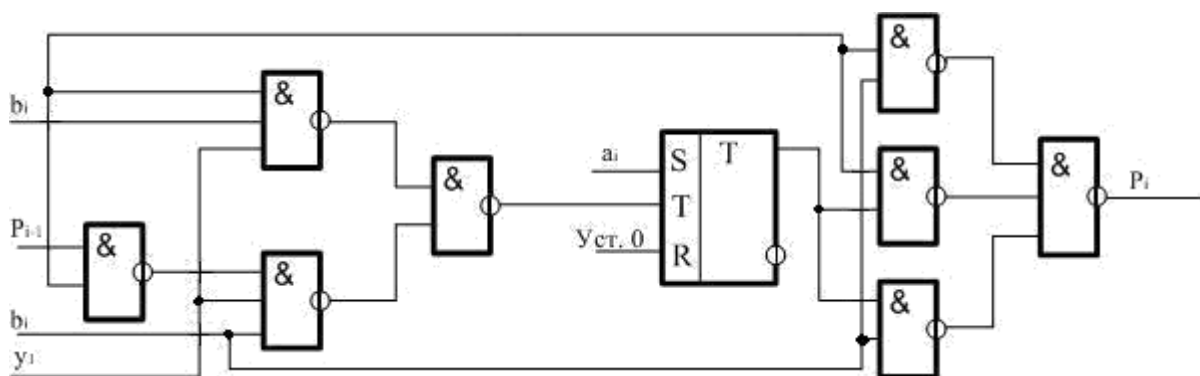


Рис.4.7

Можно уменьшить время образования суммы на один такт путем применения одноразрядных комбинационно-накапливающих сумматоров (ОКНС). В ОКНС сигнал переноса  $P_i$  формируется комбинационной схемой в соответствии с уравнением (4.2), а сумма  $S_i$  образуется в триггере, на счетный вход которого подается сумма по модулю 2 ( $b_i \oplus P_{i-1}$ ), формируемая комбинационной схемой. Схема ОКНС приведена на рис.4.7. Здесь по сигналу  $y_i$  на счетный вход триггера, хранящего значение  $a_i$ , подается сумма по модулю 2 ( $b_i \oplus P_{i-1}$ ) =  $b_i \overline{P_{i-1}} \vee \overline{b_i} P_{i-1}$ . При этом в триггере согласно уравнению (4.11) образуется сумма  $S_i$ .

Многоразрядные сумматоры. В зависимости от способа сложения разрядов чисел различаются последовательные и параллельные сумматоры. В последовательном сумматоре разряды чисел суммируются последовательно во времени, начиная с младших. Основу последовательного сумматора составляют один ОС и схема задержки переноса  $P_i$  на один такт. В состав параллельного сумматора входят  $t$  ОС по числу разрядов суммируемых чисел. ОС соединяют между собой цепями переносов. В схему параллельного сумматора разряды каждого слагаемого подаются одновременно.

## Примеры построения схем

Последовательный комбинационный сумматор На рис.4.9 представлена схема комбинационного сумматора при  $m=3$ . В состав схемы входят два сдвигающих регистра PrA и PrB, в которые заносятся слагаемые A и B; один одноразрядный комбинационный сумматор SM и схема задержки переноса  $P_i$  на один такт. Задержка реализуется с помощью JK-триггера (триггера переноса  $TT_p$ ), которым используется в режиме D-триггера [2].

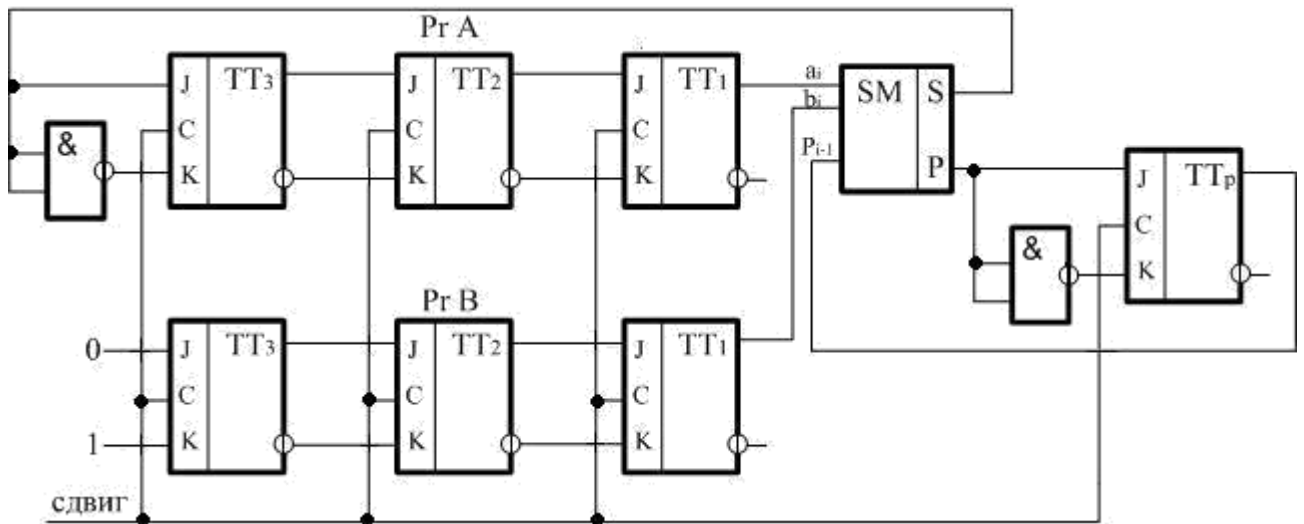


Рис.4.9

Суммирование в схеме рис.4.9 осуществляется за три такта ( $t = 3$ ). Пусть слагаемые занесены на соответствующие регистры, а триггер  $TT_p$  установлен в нулевое состояние. Тогда на входах ОКС будут действовать значения младших разрядов слагаемых  $a_i$  и  $b_i$  и сигнал  $P_0 = 0$ . При этом на выходах ОКС формируются значения  $S_i$ , которые подаются на входы триггера старшего разряда PrA и  $P_1$ , которое подается на входы триггера переноса  $TT_p$ . Если подать сигнал сдвига, то после его окончания числа на PrA и PrB сдвинутся вправо на один

разряд; значение  $S_i$  занесется в освободившийся старший разряд  $PrA$ ; значение  $P_i$  зафиксируется в триггере переноса. В результате на входах ОКС будут действовать сигналы  $a_i$ ,  $b_i$  и  $P_i$ . К моменту подачи второго сигнала сдвига на выходах ОКС будут действовать сигналы  $S_2$  и  $P_2$ . По окончании второго сигнала сдвига  $S_2$  занесется в старший разряд  $PrA$ ,  $P_2$  - в триггер  $TT_p$ . Окончательно, после подачи третьего сигнала сдвига в  $PrA$  окажется сумма  $S=S_3S_2S_1$ , а в триггере переноса будет зафиксирован перенос из старшего разряда  $P_4$ .

Время суммирования  $m$ -разрядных чисел в последовательном комбинационном сумматоре равно  $mT_{сдв} = m(\tau_{сдв} + t_n + \tau_s + \tau_0)$ , где  $T_{сдв}$  - период следования сигналов сдвига;  $\tau_{сдв}$  - длительность сигнала сдвига;  $t_n$  - время переключения триггера.

Параллельный комбинационный сумматор В параллельном комбинационном сумматоре используются  $n$  ОКС, соединенных последовательно цепями переносов. Разряды слагаемых  $A$  и  $B$  подаются на входы ОКО одновременно с двух параллельных регистров. Схема сумматора при  $n = 3$  представлена на рис.4.10. Здесь на входы трех ОКС поступают разряды слагаемых с двух регистров  $PrA$  и  $PrB$ . При этом на выходах ОКС формируются значения разрядов суммы  $S=S_3S_2S_1$ . Сигналы с выходов ОКС поступают на входы триггеров  $PrA$ . Предполагается, что разряды суммы окончательно фиксируются в  $PrA$ , который носит название регистра-накопителя сумматора. Для этого управляющий сигнал  $y_1$  подается на входы синхронизации всех триггеров  $PrA$ . Так как триггеры  $PrA$  двухступенчатые, то во время действия сигнала  $y_1$  выходные сигналы триггеров не меняются, и, следовательно, на входах ОКС поддержи-



ваются значения разрядов А. По окончании сигнала  $y_1$  в триггерах  $PrA$  зафиксируются разряды суммы.

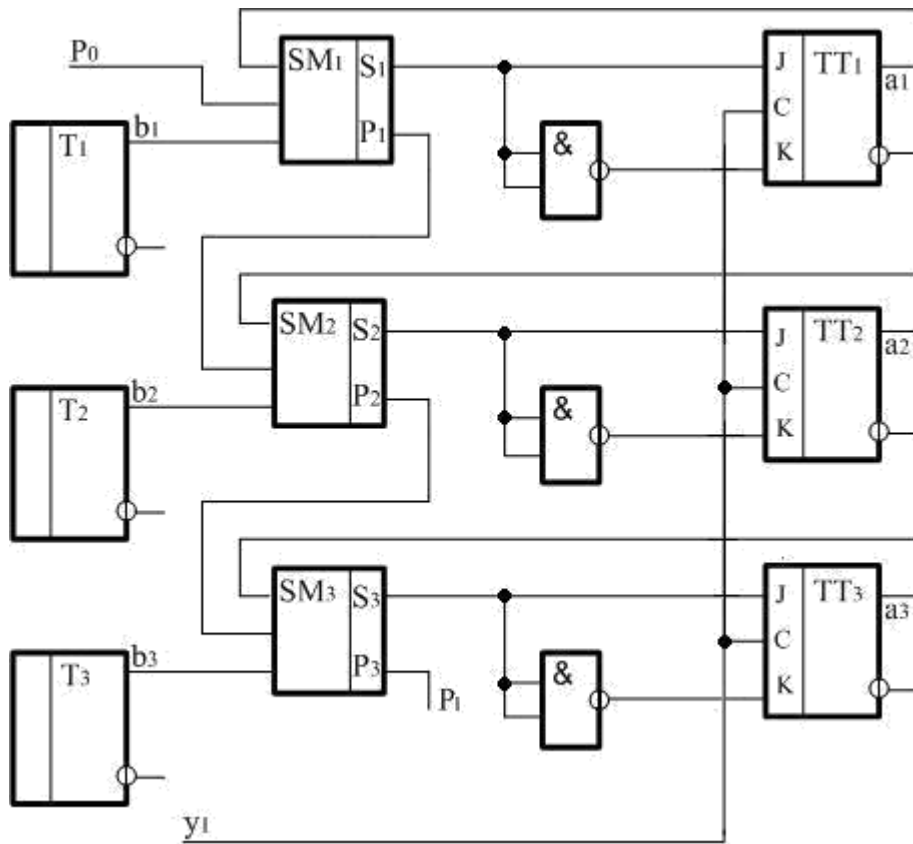


Рис.4.10

Определим максимально возможное значение времени с момента подачи на входы ОКС разрядов слагаемых до момента формирования на выходах ОКС всех разрядов суммы. Хотя в сумматор разряды А и В подаются одновременно, процесс формирования сигналов переноса является последовательным. В предельном случае сигнал переноса, возникший в младшем ОКС, будет распространяться вплоть до старшего ОКС; следовательно, искомое время равно  $(m-1)\tau_p + \tau_s$ .

Параллельный накапливающий сумматор Параллельный накапливающий сумматор строится из  $m$  ОНС, соединенных последовательно. На рис.4.11 представлена схема на-

капливающего сумматора при  $t=3$ . В состав схемы входят три ОНС, связанные цепями переносов. Триггеры, входящие в состав ОНС, образуют PrA, называемый регистром накопителем сумматора.

Суммирование чисел  $A$  и  $B$  осуществляется следующим образом. Предварительно триггеры PrA устанавливаются в нулевое состояние сигналом "Уст.0" и в PrA записывается первое слагаемое  $A$  (разряды  $A$  подаются на входы  $S$  триггеров). Второе слагаемое  $B$  предполагается записанным в PrB. Далее по управляющему сигналу  $y_1$  разряды  $B$  подаются на счетные входы триггеров PrA. В результате в каждом  $i$ -триггере ( $i=I+m$ ) образуется сумма по модулю 2 ( $a_i \oplus b_i$ ).

Сигналы с выходов триггеров PrA поступают в схемы формирования переносов, которые в каждом разряде организованы в соответствии с уравнением (4.12). По сигналу  $y_2$  сформированные сигналы переносов  $P_i$  подаются на счетные входы триггеров PrA, в которых при этом образуются разряды суммы (согласно уравнению (4.11)).

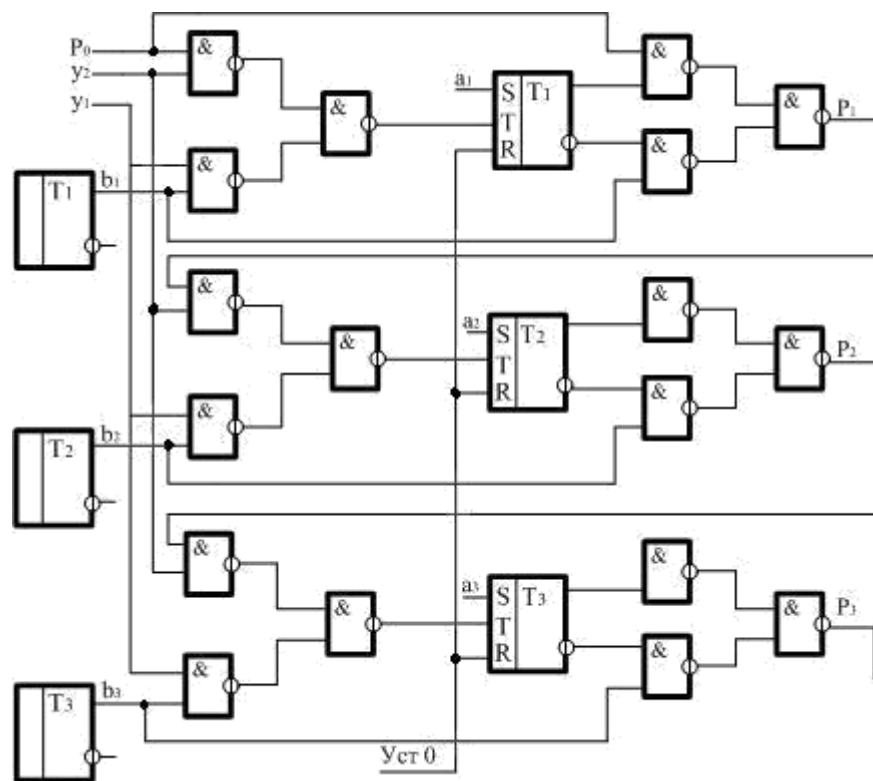


Рис.4.11

Как и в схеме комбинационного сумматора (рис.4.10), здесь процесс формирования переносов является последовательным; максимально возможное значение времени с момента переключения триггеров  $P_iA$  по сигналу  $y_1$  до момента подачи сигнала  $y_2$  равно  $m\tau_p = 2m\tau_0$ .

Параллельный комбинационный сумматор с одновременным переносом Параллельное суммирование используется в тех случаях, когда требования по быстродействию не позволяют применить последовательный сумматор. Рассмотренные параллельные сумматоры характеризуются последовательным формированием переносов между одноразрядными сумматорами (ОС) и носят название сумматоров с последовательным переносом. Применение их нецелесообразно, если задержка распространения переноса через ОС сравнима с длительностью такта ( $T_{сдв}$ ) в последовательном сумматоре. Время суммирования

оказывается при этом почти таким, как в последовательном сумматоре. Для уменьшения времени суммирования применяют параллельные сумматоры с одновременным переносом, принцип построения которых заключается в том, что значение каждого разряда суммы получается в результате одновременного анализа соответствующих и всех более младших разрядов слагаемых. Рассмотрим, как строится комбинационный сумматор с одновременным переносом.

В соответствии с уравнением (4.3) для переноса из ОКС можно записать, что

$$P_i = a_i b_i \vee (a_i \vee b_i) P_{i-1} = C_i \vee T_i P_{i-1} \quad (4.13)$$

Из уравнения (4.13) следует, что  $P_{i-1} = C_{i-1} \vee T_{i-1} P_{i-2}$ ;  $P_{i-2} = C_{i-2} \vee T_{i-2} P_{i-3}$  и т.д. Подставив выражение для  $P_{i-1}$  в уравнение (4.13), получим

$$P_i = C_i \vee T_i C_{i-1} \vee T_i T_{i-1} P_{i-2}$$

Продолжив этот процесс подстановки до появления в правой части уравнения переноса  $P_0$ , т.е. входного переноса младшего разряда сумматора, получим следующее выражение для  $P_i$  ( $i = I-m$ ):

$$P_i = C_i \vee T_i C_{i-1} \vee T_i T_{i-1} C_{i-2} \vee \dots \vee T_i T_{i-1} \dots T_2 C_1 \vee T_i T_{i-1} \dots T_2 T_1 P_0 \quad (4.14)$$

В соответствии с уравнением (4.14) строятся схемы формирования переносов в разряды сумматора с одновременным переносом (рис.4.12). Так как в правой части уравнения (4.14) участвуют лишь разряды слагаемых ( $C_i = a_i b_i$ ,  $T_i = a_i \vee b_i$ ) и перенос  $P_0$ , которые подаются в схему суммирования одновре-

менно, то, следовательно, переносы во все разряды сумматора будут формироваться одновременно. В схеме рис.4.12 в каждом ОКС выделены цепи формирования сигналов  $\overline{C}_i$  и  $T_i$ . Здесь задержка суммирования складывается из задержки в схеме формирования переноса и времени образования суммы  $S_i$  в ОКС.

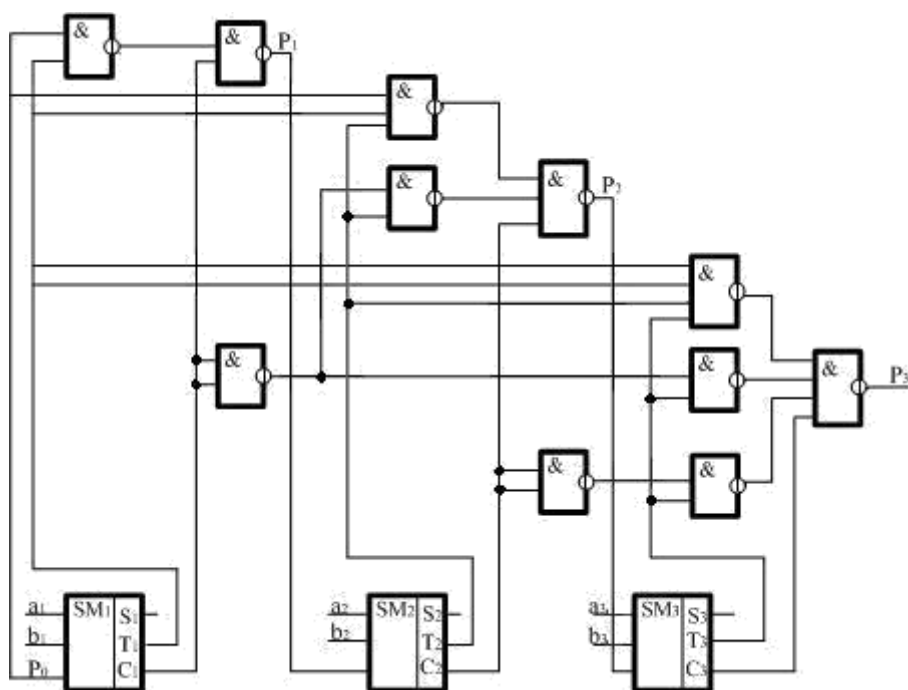


Рис.4.12

Сумматор с одновременным переносом является наиболее быстродействующим, однако разрядность его ограничена характеристиками логических элементов, а именно, коэффициентами объединения по входу J и разветвления по выходу F. В тех случаях, когда указанные ограничения не позволяют строить сумматор с заданным числом разрядов, разряды сумматора делятся на группы. При этом в пределах каждой группы переносы формируются одновременно, а группы связываются между собой цепями последовательного или одновременного переноса.

## Содержание отчета

1. Принципиальные схемы для заданных типов сумматоров, собранные в эмуляторе.

2. Результаты проверки правильности собранных схем сумматоров для различных случаев алгебраического суммирования двоичных чисел.