**СОДЕРЖАНИЕ**

Введение

1. Основные понятия. Канонический метод структурного синтеза автоматов. Теорема Глушкова о структурной полноте

2. Основные этапы канонического метода структурного синтеза

2.1 Кодирование алфавитов автомата

2.2 Выбор элементов памяти автомата

2.3 Выбор функционально-полной системы логических элементов

2.4 Построение уравнений булевых функций возбуждения и выходов автомата

2.5 Построение функциональной схемы автомата

3. Пример канонического метода структурного синтеза

4. Элементы памяти

4.1 Элементы памяти с одним информационным входом

4.2 Элементы памяти с двумя информационными входами

4.3 Матрица переходов элемента памяти

5. Кодирование состояний и выходов автомата и сложность

комбинационных схем

6. Обеспечение устойчивости функционирования цифровых автоматов. Гонки в автоматах

6.1 Методы устранения гонок в автоматах

Выводы

Литература

**Введение**

Тема курсовой работы «Структурные автоматы» по дисциплине «Прикладная теория управления автоматами».

Цель работы – рассмотреть:

## - основные понятия структурных автоматов;

## - канонический метод структурного синтеза автоматов;

## - теорему Глушкова о структурной полноте;

## - основные этапы канонического метода структурного синтеза;

## - примеры канонического метода структурного синтеза;

## - кодирование состояний и выходов автомата и сложность комбинационных схем;

## - обеспечение устойчивости функционирования цифровых автоматов;

## - гонки в автоматах;

## - методы устранения гонок в автоматах и др.

## **Основные понятия. Канонический метод структурного синтеза автоматов. Теорема Глушкова о структурной полноте**

Вслед за этапом абстрактного синтеза автоматов, заканчивающимся минимизацией числа состояний, следует этап структурного синтеза, цепью которого является построение схемы, реализующей автомат из логических элементов заданного типа.

Если абстрактный автомат был лишь математической моделью дискретной системы, то в структурном автомате учитывается структура входных и выходных сигналов автомата, а также его внутреннее устройство на уровне структурных схем. Структурным синтезом занимается структурная теория автоматов, основной задачей которой является нахождение общих приемов построения структурных схем автоматов на основе композиции элементарных автоматов, принадлежащих к заранее заданному конечному числу типов.

Под композицией элементарных автоматов в общем случае понимается следующее.

Пусть заданы элементарные автоматы S1,...,Sk. Произведем объединение элементарных автоматов в систему совместно работающих автоматов. Введем в рассмотрение некоторое конечное множество узлов, называемых внешними входными и внешними выходными узлами. Эти узлы отличаются от узлов рассматриваемых элементарных автоматов, которые носят название внутренних. Композиция автомата состоит в том, что в полученной системе элементарных автоматов S1,...,Sk и внешних узлов производится отождествление некоторых узлов (как внешних, так и внутренних). С точки зрения совместной работы системы автоматов смысл операции отождествления узлов состоит в том, что элементарный сигнал, попадающий на один из узлов, входящих в множество отождествляемых между собой узлов, попадает тем самым на все узлы этого множества, После проведенных отождествлений узлов система автоматов превращается в так называемую схему (сеть) автоматов. Будем считать, что автоматы, входящие в схему автоматов, работают совместно, если в каждый момент автоматного времени на все внешние входные узлы подается набор входных сигналов (структурный входной сигнал схемы) и со всех внешних выходных узлов снимается набор выходных сигналов (структурный выходной сигнал).

В структурной теории как входные так и выходные каналы считаются состоящими из элементарных входных (выходных) каналов. По всем элементарным входным (выходным) каналам могут передаваться только элементарные сигналы.

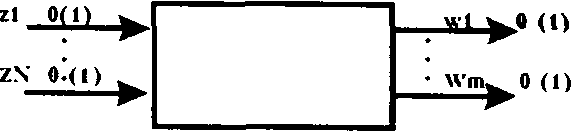


Рисунок 1- Структурный автомат

Набор возможных значений сигналов, подаваемых на один внешний входной (выходной) узел, называется структурным входным (выходным) алфавитом автомата. Алфавит должен быть конечным.

Входной и выходной сигналы задаются конечными упорядоченными наборами элементарных сигналов, называемыми векторами, а составляющие их элементарные сигналы - компоненты векторов. Число компонентов вектора - это размерность алфавита.

Например, X={x1, x2, x3, x4, x5} - входной алфавит абстрактного автомата.

Структурный входной алфавит, размерность которого равна трем:

X1=000, x2=001, x3=010, x4=011, x5= 100.

Векторное представление входных и выходных сигналов называется структурным входным выходным сигналом, соответственно.

Предполагается, что все входящие в композицию автоматы имеют один и тот же структурный алфавит и работают в одном и том же автоматном времени. В настоящее время наиболее распространенным структурным алфавитом является двоичный, что объясняется простотой его представления в современных элементах и приборах. Кроме того, для двоичного алфавита наиболее разработан аппарат булевых функций, позволяющий производить многие операции над схемой формально. Поэтому в дальнейшем при решении задач структурного синтеза автоматов будет использоваться в основном двоичный, структурный алфавит.

Предположим, что в каждый момент автоматного времени структурный выходной сигнал схемы однозначно определяется поступившей к этому времени конечной последовательностью структурных входных сигналов, начальными состояниями входящих в схему автоматов и сделанными при построении схемы отождествлениями узлов. В этом случае построенную схему будем рассматривать как некоторый автомат S, а саму схему назовем структурной схемой этого автомата.

Построенный таким образом автомат S есть результат композиции элементарных автоматов S1,...,Sk. В отличие от абстрактного C-автомата, имеющего один входной и два выходных канала, на которые поступают сигналы во входном и выходных алфавитах автомата, структурный автомат имеет входные и выходные каналы, на которых появляются сигналы в структурном алфавите автомата. В случае двоичного алфавита каждый входной и выходные сигналы абстрактного автомата могут быть закодированы векторами различной длины соответственно, компоненты которых принимают два значения - нуль и единицу.

На этапе структурного синтеза предварительно выбираются элементарные автоматы, из которых затем путем их композиции строится структурная схема полученного на этапе абстрактного синтеза автомата Мили, Мура или C-автомата. Если решение задачи структурного синтеза существует, говорят, что заданная система автоматов структурно полна.

В настоящее время нет сколько-нибудь эффективных методов (существенно более простых, чем метод перебора всех вариантов) решения основной задачи структурного синтеза при любом наборе структурно полных систем элементарных автоматов. Поэтому обычно применяется так называемый канонический метод структурного синтеза, при котором используются элементарные автоматы некоторого специального вида: автоматы с памятью, имеющие более одного состояния, и автоматы без памяти - с одним состоянием. Автоматы первого класса носят название элементов памяти, а автоматы второго класса - комбинационных или логических элементов.

Теоретическим обоснованием канонического метода структурного синтеза автоматов является доказанная в теорема о структурной полноте (теорема Глушкова):

Всякая система элементарных автоматов, которая содержит автомат Мура с нетривиальной памятью, обладающий полной системой переходов и полной системой выходов, и какую-либо функционально полную систему логических элементов, является структурно полной.

Существует общий конструктивный прием (канонический метод структурного синтеза), позволяющий в рассматриваемом случае свести задачу структурного синтеза произвольных автоматов к задаче синтеза комбинационных схем.

Результатом канонического метода структурного синтеза является система логических уравнений, выражающая зависимость выходных сигналов автомата (функции выходов автомата) и сигналов, подаваемых на входы запоминающих элементов, от сигналов, приходящих на вход всего автомата в целом, и сигналов, снимаемых с выхода элементов памяти (функции возбуждения элементов памяти автомата). Эти уравнения называютсяканоническими.

Для правильной работы схем, очевидно, нельзя разрешать, чтобы сигналы на входе запоминающих элементов непосредственно участвовали в образовании выходных сигналов, которые по цепям обратной связи подавались бы в тот же самый момент времени на эти входы. В связи с этим запоминающими элементами должны быть не автоматы Мили, а автоматы Мура (см. уравнения функционирования этих автоматов).

Таким образом, структурно полная система элементарных автоматов должна содержать хотя бы один автомат Мура. В то же время для синтеза любых автоматов с минимальным числом элементов памяти необходимо в качестве таких элементов выбирать автоматы Мура, имеющие полную систему переходов и полную систему выходов - так называемые полные автоматы.

Рассмотрим полноту автоматов памяти на примере автомата Мура. (табл. 1.) Полнота системы переходов означает, что для любой пары состояний (am,…,аs) автомата найдется входной сигнал, переводящий первый элемент этой пары am во второй - as т. е. в таком автомате в каждом столбце таблицы переходов должны встречаться все состояния автомата. Полнота системы выходов автомата Мура состоит в том, что каждому состоянию автомата поставлен в соответствие свой особый выходной сигнал, отличный от выходных сигналов других состояний.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 1. | | | |
| У | y1 | y2 | y3 |
| A\X | x1 | x2 | x3 |
| a1 | a1 | a2 | a3 |
| a2 | a3 | a1 | a2 |
| a3 | a2 | a3 | a1 |

Очевидно, что в таком автомате число выходных сигналов равно числу состояний автомата. Вместе с предыдущим утверждением это приводит к тому, что в автомате Мура с полной системой выходов можно отождествить состояния автомата с его выходными сигналами. В связи с этим в автоматах памяти мы будем использовать одни и те же обозначения и для состояний, и для выходных сигналов, т. е. отмеченная таблица переходов в автоматах Мура с полной системой выходов превращается просто в таблицу переходов. Автомат Мура в табл. 1. удовлетворяет условиям полноты системы переходов и выходов.

Наличие функционально полной системы логических элементов позволяет реализовать булеву функцию любой степени сложности.

После выбора элементов памяти и кодирования состояний синтез структурного автомата сводится к синтезу комбинационной схемы, реализующей канонические уравнения.

Автомат памяти также можно рассматривать на абстрактном и структурном уровнях. Абстрактный автомат памяти имеет один входной и один выходной каналы. При переходе от абстрактного автомата к структурному автомату входные и выходные сигналы должны быть закодированы соответствующими двоичными наборами.

## **Основные этапы канонического метода структурного синтеза**

В каноническом методе структурного синтеза можно выделить несколько основных этапов:

1. Кодирование алфавитов автомата.

2. Выбор элементов памяти.

3. Выбор функционально полной системы логических элементов.

4. Запись и минимизация канонических уравнений.

5. Построение функциональной схемы автомата.

Исходными данными для начала работы данного метода являются абстрактный цифровой автомат с памятью, заданный таблицей переходов и выходов. Рассмотрим подробнее каждый из этапов канонического метода.

### **2.1 Кодирование алфавитов автомата**

На структурном уровне каждая буква входного алфавита x Х, каждая буква выходного алфавита yY и каждая буква алфавита состояний аА кодируется двоичными векторами (двоичными наборами), число компонент которых определяется следующим образом:



Kвх >= int(log2|X|); Kвых >= int(log2|Y|); Kсост>=int(log2|A|);

где int - ближайшее большее целое число.

|Х|, |У|, |А| - мощность алфавита входного, выходного и состояний, соответственно. Мощностью алфавита называется количество различных символов входящих в этот алфавит.

Процесс замены буква алфавита (X, У, А) абстрактного автомата двоичными векторами носит название кодирования и описывается таблицами кодирования. Таблица кодирования имеем следующий вид: в левой части перечисляются все символы абстрактного алфавита, а в правой - соответствующие им двоичные векторы.

Рассмотрим пример.

Абстрактный автомат Мили задан таблицей переходов и выходов (табл. 2.). Выполнить кодирование алфавитов автомата.

Таблица2

|  |  |  |
| --- | --- | --- |
| А\ Х | x1 | x2 |
| a1 | a2/y1 | a3/y3 |
| a2 | a1/y2 | a2/y1 |
| a3 | a2/y2 | a1/y1 |

Выпишем алфавиты автомата и определим длины векторов кодов алфавитов:

X={x1,x2}; Kвх >= int(log2|2|)=1,

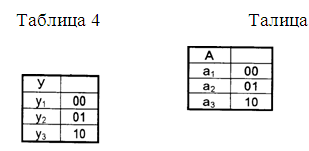
Y={y1,y2,y3}; Kвых >= int(log2|3|)=2,

A={a1,a2,a3}; Kсост >= int(log2|3|)=2.

Заполним таблицы кодирования:

Таблица 3

|  |  |
| --- | --- |
| x1 | 0 |
| x2 | 1 |



Результирующая таблица - структурная таблица переходов и выходов автомата (табл. 6.) Получением структурной таблицы переходов -выходов автомата заканчивается этап кодирования.

Таблица 6.

|  |  |  |
| --- | --- | --- |
| А\ X | 0 | 1 |
| 00 | 01/00 | 10/10 |
| 01 | 00/01 | 01/00 |
| 10 | 01/01 | 00/00 |

В общем случае, каждой кодируемой букве может быть присвоен произвольный двоичный вектор, но обязательно две различные буквы одного алфавита должны кодироваться различными векторами. Коды, соответствующие символам различных алфавитов могут совпадать, в рассмотренном примере - коды выходных сигналов и состояний.

На данном этапе целесообразно отметить, что способ кодирования состояний в значительной степени определяют сложность реализации комбинационных схем. Существуют специальные способы кодирования, обеспечивающие получение экономичных схем. Они будут рассмотрены ниже.

### 

### **2.2 Выбор элементов памяти автомата**

Замена таблицы переходов автомата на структурную таблицу переходов приводит к тому, что функция переходов (аi,xj) автомата становится векторной. Иными словами, аргументами такой функции являются все возможные пары двоичных векторов (ai,xj), а сама функция принимает значение из множества A двоичных векторов состояний автомата. В соответствии со структурной таблицей переходов автомата его векторная функция переходов каждой паре двоичных векторов ставит в соответствие определенный двоичный вектор ak, что на абстрактном уровне определяется соотношением аk = (аi,хj)*.* Из этого следует, что структурный автомат должен запоминать двоичный вектор каждого очередного состояния автомата, для чего служат элементы памяти (запоминающие элементы, триггеры).

При каноническом методе структурного синтеза автоматов в качестве элементов памяти используются элементарные автоматы Мура с двумя состояниями, обладающие полной системой переходов и выходов.

Полнота системы переходов автомата в общем случае означает, что для любой пары состояний автомата существует входной сигнал, переводящий элементарный автомат из одного состояния в другое. Таблица переходов элементарного автомата с полной системой переходов должна содержать в каждой своей строке все возможные состояния.

Полнота системы выходов означает, что различным состояниям автомата соответствуют различные выходные сигналы. Обычно нулевому состоянию элементарного автомата соответствует нулевой выходной сигнал, единичному - единичный.

Очевидно, что число элементов памяти структурного автомата равно числу компонент вектора его состояний.

### 

### **2.3 Выбор функционально-полной системы логических элементов**

Функционирование структурного автомата во времени предполагает управление переключением каждого элементарного автомата его памяти в соответствии со структурной таблицей переходов синтезируемого автомата. Последнее осуществляется с помощью специальной комбинационной схемы, подключаемой к информационным входам элементарного автомата памяти и реализующей булевы функции, управляющие его переключением.

Такие булевы функции называются функциями возбуждения элемента памяти и, в общем случае, различных функций возбуждения столько, сколько различных информационных входов имеется у элементарных автоматов памяти в синтезируемом структурном автомате.

Функция возбуждения любого элемента памяти является произвольной булевой функцией и для ее реализации комбинационной схемой необходимо использовать какую-либо функционально-полную систему логических элементов. Теоретическим фундаментом канонического метода структурного синтеза автоматов с памятью является теорема о структурной полноте, из которой следует, что для построения структурного автомата необходимо кроме элементов памяти иметь комбинационную схему, реализующую булевы функции возбуждения элементов памяти автомата, а для выработки выходных сигналов структурного автомата - специальную комбинационную схему формирования выходных сигналов автомата.

Функция возбуждения структурного автомата является векторной. Ее аргументами являются пары двоичных векторов (аi,хj).- а значением функции - двоичный вектор, каждая i-я компонента которого есть значение булевой функции возбуждения i-го элемента памяти автомата, определяющая тот двоичный сигнал, который необходимо подать на вход i-го элемента памяти для обеспечения его переключения в соответствии с требованиями структурной таблицы переходов.

Если векторная функция переходов задает переход из одного вектора состояния структурного автомата в другой вектор состояния под воздействием двоичного вектора входного сигнала, то векторная функция возбуждения автомата задает двоичный вектор, который нужно подать на входы элементов памяти автомата, чтобы обеспечить требуемый переход (в соответствии с векторной функцией переходов автомата).

Последнее означает, что переменными, от которых зависит векторная функция возбуждения, являются те же переменные, что и для векторной функции переходов автомата, т. е. выходы всех элементов памяти автомата и входы автомата. Поэтому структурный автомат Мура, в общем случае, может быть представлен структурной схемой (рис. 2), а структурный автомат Мили - структурной схемой (рис. 3).

Буквами б1,... б ц на рисунках обозначены выходы элементов памяти. Буквами ц1,…, ц j обозначены булевы функции возбуждения элементов памяти. Для простоты положим, что каждый элемент памяти структурного автомата имеет один информационный вход. Буквами w1,...,wj обозначены выходные каналы автомата, где j - число выходных каналов автомата. Буквами z1,…,zm – входные каналы.

### **2.4 Построение уравнений булевых функций возбуждения и выходов автомата**

Кодирование и выбор системы элементов однозначно определяют комбинационную часть автомата: вначале строится таблица истинности функций возбуждения элементов памяти автомата, получившая название таблицы функций возбуждения; канонические уравнения функций возбуждения выписываются исходя из построенной таблицы. Полученная аналитическая запись булевой функции возбуждения (для каждого элемента памяти автомата) может быть минимизирована любым из известных методов. Исходными данными для построения таблицы функций возбуждения являются структурная таблица переходов автомата и таблица переходов элемента памяти. Обрамление таблицы функций возбуждения, т.е. идентификация ее строк и столбцов полностью совпадает с обрамлением структурной таблицы переходов синтезируемого автомата. Клетки, расположенные внутри таблицы функций возбуждения, заполняются специальным образом.

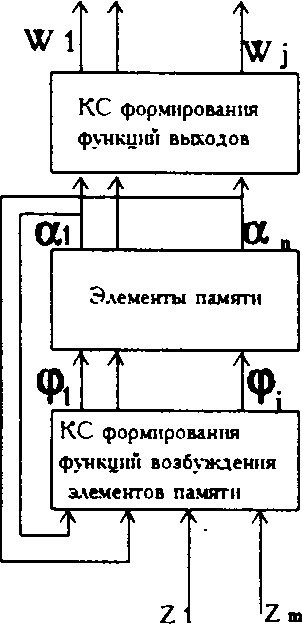


Рисунок 2- Структурная схема автомата Мура



Рисунок 3- Структурная схема автомата Мили

Получение канонических уравнений булевых функций выходов структурного автомата проще и может быть сделано непосредственно по структурной таблице выходов автомата. Структурная таблица выходов автомата есть таблица истинности булевых функций выходов автомата. Иными словами, уравнения булевых функций выходов автомата не зависят от типа используемых элементов памяти, однако зависят от их количества.

Если синтезируемый автомат является автоматом Мура, то задача построения уравнений булевых функций возбуждения решается так же. Уравнения булевых функций выходов автомата Мили строятся несколько иначе. Последнее связано с различиями в способах построения структурных таблиц выходов автомата Мили и Мура. После проведения этапа кодирования состояний автомата и кодирования выходных сигналов получаем структурную таблицу выходов автомата, которая является таблицей истинности булевых функций выходов автомата.

### **2.5 Построение функциональной схемы автомата**

На основании полученных выражений для булевых функций возбуждения элементов памяти автомата и булевых функций выходов автомата строятся комбинационная схема функций возбуждения и комбинационная схема формирования выходных сигналов автомата. Элементы памяти подключаются к построенным комбинационным схемам. Функциональная схема автомата Мура отлична только комбинационной схемой формирования выходных сигналов, которая строится на основании уравнений для булевых функций выходов. Отметим, что реализация комбинационных схем может быть выполнена в любом функционально-полном базисе.

## **Пример канонического метода структурного синтеза**

Пусть задан абстрактный автомат Мили таблицей переходов и выходов (табл. 7.). Используя логические элементы И, ИЛИ, НЕ и элементарный автомат Мура (элемент памяти), заданный табл. 8.

Таблица 7.

|  |  |  |  |
| --- | --- | --- | --- |
| А\ Х | x1 | x2 | x3 |
| a1 | a2/y3 | a3/y4 | a2/y2 |
| a2 | - | a1/y3 | a3/y1 |
| a3 | a1/y2 | - | a3/y3 |
| А\ Х | x1 | x2 | x3 |
| a1 | a2/y3 | a3/y4 | a2/y2 |
| a2 | - | a1/y3 | a3/y1 |
| a3 | a1/y2 | - | a3/y3 |

Таблица 8

|  |  |  |
| --- | --- | --- |
|  | r1 | r2 |
| b1 | b1 | b2 |
| b2 | b2 | b1 |

Выполним кодирование алфавитов автомата (табл. 7.)

Выпишем алфавиты автомата и определим длины векторов кодов алфавитов:

Х = {x1, x2, x3}; Kвх >= int(log2|3|)= 2,

У = {y1, y2, y3, y4}; Kвых >= int(log2|3|)= 2,

А = {a1, a2, а3}; Kсост >= int(log2|3|)= 2.

Заполним таблицы кодирования (табл. 9 – 11):

Таблица 9.

|  |  |  |
| --- | --- | --- |
| Х | z1 | z2 |
| x1 | 0 | 0 |
| x2 | 0 | 1 |
| x3 | 1 | 0 |

Таблица 10

|  |  |  |
| --- | --- | --- |
| У | w1 | w2 |
| y1 | 1 | 0 |
| y2 | 0 | 0 |
| y3 | 1 | 1 |
| y4 | 0 | 1 |

Таблица 11

|  |  |  |
| --- | --- | --- |
| А | 1 | 2 |
| a1 | 0 | 0 |
| a2 | 0 | 1 |
| a3 | 1 | 1 |

Каждый разряд вектора кода обозначим символом с соответствующим номером. Входные - z, выходные - w, состояния – *а*.

Таблица 12

|  |  |  |  |
| --- | --- | --- | --- |
|  | z1z2 | | |
| 12 | 00 | 01 | 10 |
| 00 | 01/11 | 11/01 | 01/00 |
| 01 | - | 00/11 | 11/10 |
| 11 | 00/00 | - | 11/11 |

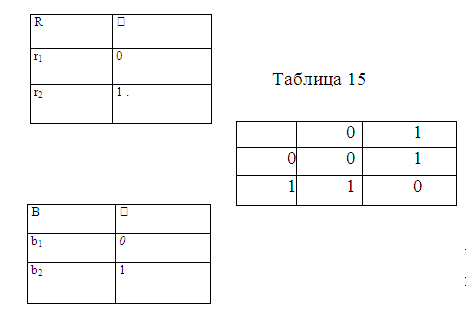
В результате получим таблицу переходов и выходов структурного автомата (табл. 12.). Выполним кодирование элементарного автомата Мура (табл. 8.):

- выпишем алфавиты автомата и определим длины векторов кодов алфавитов,

R={r1,r2}; Kвх >= int(log2|2|)=1,

В = {b1, b2}; Kсост >= int(log2|2|)= 1;

-заполним таблицы кодирования:



Структурная таблица переходов элементарного автомата Мура имеет вид (табл. 15.):

Так как абстрактный автомат имеет три состояния, каждое из которых кодируется двумя разрядами, то структурный автомат будет содержать два запоминающих элемента.

Теперь задача сводится к синтезу комбинационной схемы, реализующей канонические уравнения:

w12, z1, z2) - функции выходов автомата

ϕ1= fϕ) - функции возбуждения элементов памяти автомата.

Функции w1, w2 можно получить непосредственно из отмеченной таблицы переходов структурного автомата как дизъюнкцию конъюнкций, соответствующих тем наборам (1,2.z1,z2), на которых эти функции принимают значения 1. Но более удобно пользоваться так называемой таблицей формирования функций возбуждения и функций выходов автомата, в которой в табличной форме задана система булевых функций (табл. 16). Заполним эту таблицу, используя коды соответствующих алфавитов и таблицу переходов и выходов абстрактного автомата. Для заполнения колонок ϕ1, ϕ2 необходимо воспользоваться еще и таблицей элемента памяти (табл. 15).

Для заполнения функций возбуждения элементов памяти рассматривается переход из исходного состояния (12) в состояние перехода (12)*.* За первый разряд 1 отвечает первый элемент памяти (его функция ϕ1), за второй 2 *-* второй элемент памяти (его функция ϕ2).

В таблице проставляется значение входного сигнала, который обеспечивает соответствующий переход. Количество функций возбуждения элементов памяти автомата зависит от количества разрядов вектора кода состояния и от количества информационных входов самого запоминающего элемента. Рассмотрим, например, что будет со структурным автоматом, если он находится в состоянии 01, и на его вход поступил сигнал 10.

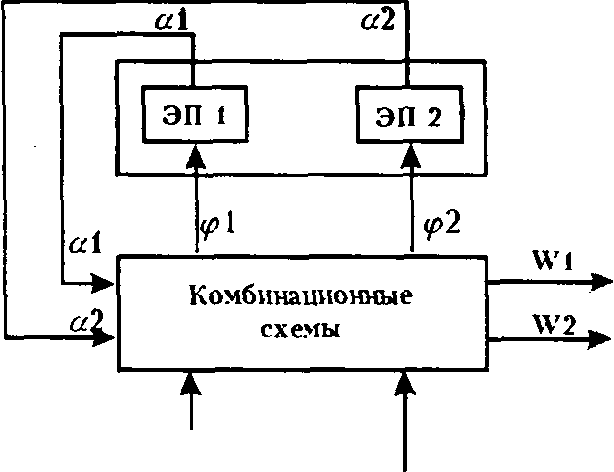
Как видно из таблицы переходов структурный автомат перейдет в состояние 11. Этот переход складывается из двух переходов элементов памяти: 1-й из 0 в 1, 2-й из 1 в 1. По таблице 15 определим входные сигналы элемента памяти, обеспечивающие эти переходы: это 0 и 1., и т.д. В клетку соответствующего перехода запишем вектор функции возбуждения, вызывающий данный переход.

Таблица 16.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Исходное состояние | | Входной сигнал | | Состояние перехода | | Функции возбуждения эл-в памяти | | Выходной сигнал | |
|  |  |  |  |  |  | ϕ | ϕ |  |  |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
|  |  | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
|  |  | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
|  |  | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
|  |  | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |



По таблице 16 запишем аналитические выражения канонических уравнений:



Z1Z2

Рисунок 4- Структурная схема автомата

Не занимаясь минимизацией канонический уравнений, построим схему электрическую функциональную (рис. 5).

## **Элементы памяти**

В качестве элементов памяти структурного автомата обычно используются триггеры. Как уже было сказано, с точки зрения прикладной теории цифровых автоматов, триггер - это элементарный автомат Мура, обладающий полной системой переходов и полной системой выходов.

Триггер характеризуется числом информационных входов, внутренних состояний, числом выходных сигналов и т.д. выходные сигналы триггера отождествляются с его внутренними состояниями, именно поэтому таблица переходов совпадает с таблицей выходов и триггер задается только одной из них (таблицей переходов). Как правило, триггер формирует как прямой сигнал, так и инверсный.

Рассмотрим некоторые из этапов канонического метода более подробно, с применением специальных методов.

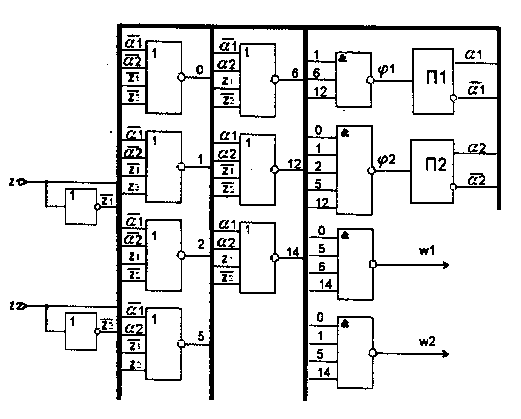


Рисунок 5- Функциональная схема автомата

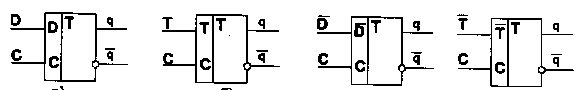
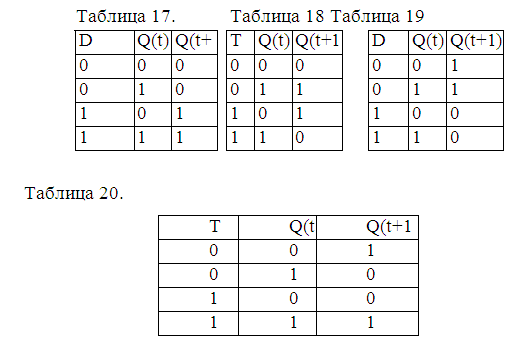
### 

### **4.1 Элементы памяти с одним информационным входом**

Существует только 4 типа запоминающих элементов с одним информационным входом, имеющих полную систему переходов и выходов: D-триггер, Т-триггер, -триггер, -триггер. Таблицы их переходов представлены табл. 17 - 20. соответственно, а условные графические изображения триггеров представлены на рис. 6. Входы D, T, называются информационными.



Таблицы переходов триггеров составляются только для информационных входов. Остальные входы являются вспомогательными. В частности, вход C - вход для подключения синхросерии (о чем будет сказано ниже). Каждый из триггеров имеет два выхода. Появление единичного сигнала на выходе, помеченном на рисунках символом q, означает, что триггер находится в единичном состоянии. Появление единичного сигнала на выходе говорит о нулевом состоянии.



а) б) в) г )

Рисунок 6- Условное графическое обозначение триггеров:

а)D-триггер; б) Т-триггер; в) D-триггер; г) T-триггер

В таблицах переходов две первые колонки одинаковые - в них перечислены все возможные комбинации входного сигнала и состояния элемента памяти. Для того, чтобы элементарный автомат имел полную систему переходов, колонку Q(t+1) следует заполнить таким образом, чтобы во второй и третьей колонках встречались все возможные типы переходов (00, 01, 10, 11). Для триггера типа D колонка Q(t+1) и D совпадают, т.к. выходной сигнал отождествляется с состоянием, то это означает, что данный элемент является элементом задержки на 1 такт. Его характеристическое уравнение имеет вид:





Триггер типа Т изменяет свое состояние только при подаче на его вход «1». Это триггер со счетным входом. Его характеристическое уравнение имеет вид:





### **4.2 Элементы памяти с двумя информационными входами**

Триггеры с двумя информационными входами имеют различное построение в зависимости от режимов использования имеющихся входов. Основными, наиболее распространенными двухвходовыми триггерами являются RS-триггер, JK-триггер, синхронизированный D-триггер. Рассмотрим подробнее каждый из них.

**RS-триггер**

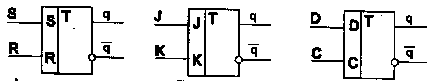
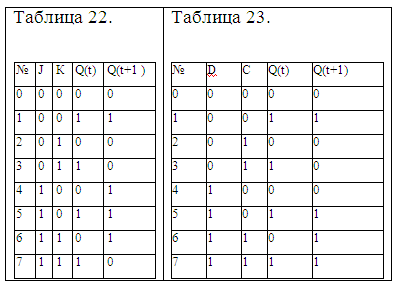
Название этого элемента происходит от английских слов «set-reset» - «установка-сброс». Он имеет два установочных входа: S -установка в 1, R - установка в ноль (сброс). Работа описывается таблицей переходов (табл. 21). На 6 и 7 наборах функция не определена, т.к. считается, что нет необходимости устанавливать данный триггер в положение «1» и «О» одновременно. Таким образом, входная комбинация 11 для RS-триггера является запрещенной и не должна возникать в реальных условиях работы.

Характеристическое уравнение после его преобразования и минимизации имеет вид:

**



Это соотношение показывает, что при нулевом сигнале на входе «установка в ноль» (R=0) RS-триггер является дизъюнктором накапливающего типа. Он осуществляет логическое сложение содержимого триггера Q(t) и сигнала S(t), после чего результат операции записывается вместо первого слагаемого. В частном случае, при обнуленном триггере, осуществляется запись в триггер той информации, которая поступила на вход S. Условное графическое обозначение RS-триггера представлено на рис. 7.а).



а) б) в)

Рисунок 7- Условное графическое обозначение триггеров:

а) RS-триггер; б) JK-триггер; в) синхронизированный D-триггер.

**J-K-триггер**

Он имеет два установочных входа: J - установка в 1, К - установка в ноль. Работа описывается таблицей переходов (табл. 22). Для него не существует запрещенных наборов входных сигналов.

Характеристическое уравнение после его преобразования и минимизации имеет вид:

**

Из этого соотношения следует, что JK-триггер является универсальным, имеющим два режима работы.

1) Режим записи и хранения информации, пришедшей по входу J, если триггер заранее был обнулен, поскольку работа обнуленного JK-триггера описывается уравнением RS-триггера. Данный режим называется RS-режимом.

2) Режим счета, который возникает при обеспечении одинаковых сигналов на обоих входах. Поскольку такой режим описывается уравнением, аналогичным уравнению Т-триггера, то его можно назвать Т-режимом. Условное графическое обозначение JK-триггера представлено на рис. 7.б.

**D-триггер**

Триггер имеет также 2 входа: информационный (D) и синхронизирующий (С). Функция на выходе триггера принимает значение информационного сигнала, если есть разрешающий сигнал по входу C (С=1). При отсутствии разрешающего сигнала (С=0), значение функции замораживается, т.е. остается равным содержимому триггера на предыдущем такте. Работа описывается таблицей переходов (табл. 23.).

Характеристическое уравнение триггера имеет вид:

**



Из чего следует, что D-триггер является переключателем накапливающего типа: он пропускает на выход либо сигнал, приходящий по условной шине D, либо сигнал, приходящий по условной шине Q(t), в зависимости от значения управляющего сигнала С. Условное графическое обозначение триггера представлено на рис. 7.в.

### **4.3 Матрица переходов элемента памяти**

Элемент памяти (триггер) может быть задан одним из нескольких способов: сокращенной таблицей переходов, полной таблицей переходов, характеристическим уравнением, матрицей переходов. Рассмотрим последний способ.

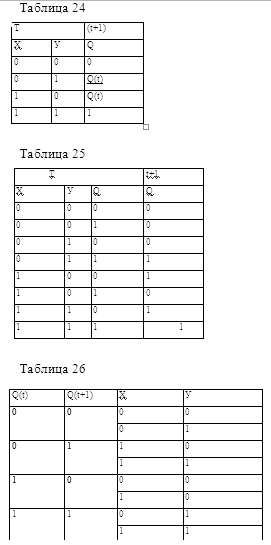
Для каждого из 4-х возможных переходов элементарного автомата (00, 01, 10, 11) всегда найдется значение входного сигнала, равное 0 или 1, которое вызывает данный переход. Если элементарный автомат имеет 2 или более входов, то на некоторые переходы значения входных сигналов, действующих на одном или другом входе, оказываются несущественными.

Количество строк матрицы всегда равно 4 (по количеству возможных переходов), количество столбцов равно числу входных сигналов. Элемент матрицы bisk представляет собой значение входного сигнала xk под действием которого элементарный автомат перейдет из состояния i в состояние s. При этом bisk всегда равняется 0 или 1, или неопределен, если он не влияет на данный переход.

Матрица переходов элементарного автомата составляется по таблице переходов.

Рассмотрим пример построения матрицы переходов триггера.

Пусть триггер, в общем случае, задан сокращенной таблицей переходов (табл. 24.). Построить полную таблицу переходов триггера и матрицу переходов.



Полная таблица переходов триггера, построенная по сокращенной, представлена в таблице 25. По полной таблице переходов запишем комбинации входных сигналов, соответствующих всем возможным переходам (табл. 26.) Таким образом:

1. Для перехода "0-0" Х=0, Y может быть равен 0 или 1

2. Для перехода "0-1" Х=1, Y может быть равен 0 или 1

3. Для перехода "1-0" Х может быть равен 0 или 1, а Y=0.

4. Для перехода "1-1" Х может быть равен 0 или 1, а Y=1.

Тогда матрица переходов триггера запишется в виде:

|  |  |  |
| --- | --- | --- |
| 0-0 | 0 | b1 |
| 0-1 | 1 | b2 |
| 1-0 | b3 | 0 |
| 1-1 | b4 | 1 |

где b1,b2,b3,b4 - произвольные сигналы (0 или 1).

Как правило, значение двух различных коэффициентов bi, и bs из одной строки матрицы являются зависимыми друг от друга и нахождение этой зависимости с ростом числа информационных входов усложняется. Установление точной взаимозависимости между входными переменными триггера является обязательным условием, обеспечивающим возможность максимального упрощения схем с памятью. В основе методики лежит таблица, в которой представлены возможные сочетания входных переменных и соответствующие им описания (табл. 27.).



С учетом доопределений входных переменных матрицы переходов некоторых стандартных триггеров имеют вид (таб. 28.)

Таблица 28.Матрицы переходов триггеров

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Триггер-D | | | Триггер-Т | | | Триггер-RS | | | | Триггеp-JK | | | |
| Q(t) | Q(t+1) | D | Q(t) | Q(t+1) | Т | Q(t) | Q(t+1) | R | S | Q(t) | Q(t+1) | К | J |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |

## **Кодирование состояний и выходов автомата и сложность комбинационных схем**

Анализ канонического метода структурного синтеза показал, что различные варианты кодирования состояний автомата приводят к различным выражениям для функций возбуждения и функций выходов.

В рассмотренном ранее примере коды состояний автомата принимали значения: a1=00, a2=01, а3=11. Если принять коды: a1=01, а2=01, а3=00, то таблица переходов структурного автомата примет вид (табл. 29).

Если в качестве запоминающего элемента применить D-триггер, то таблица формирования функций возбуждения будет совпадать с данной таблицей. Тогда функции примут вид:

Таблица 29

|  |  |  |  |
| --- | --- | --- | --- |
|  | ZlZ2 | | |
| α1α2 | 00 | 01 | 10 |
| 01 | 10 | 00 | 10 |
| 10 | - | 01 | 00 |
| 00 | 01 | - | 00 |

;



;



Если сравнить с предыдущим результатом, то нетрудно сделать вывод, что реализация этих выражений комбинационной схемой проще.

В данном случае при кодировании состояний был использован весовой метод, который может быть использован и для кодирования выходных сигналов.

*Алгоритм весового метода кодирования:*

1. Каждому выходному сигналу у, ставится в соответствие целое число Pi, равное числу появлений сигнала уi в таблице выходов автомата.

2. Числа Pi сортируются по убыванию.

3. Выходной сигнал yi с наибольшим весом (Pi max) кодируются кодом, содержащим все 0 (00...00).

4. Следующие L выходных сигналов (где L- число разрядов в двоичном векторе выходного сигнала) по списку убывания веса (см п. 2) кодируются кодами, содержащими одну 1 (00...01, 00...10,... ,10...00).

5. Для кодирования следующих по списку убывания выходных сигналов используются все коды, содержащие две единицы, затем три единицы и т.д., пока не будут закодированы все выходные сигналы.

В результате получаем такое кодирование, при котором чем чаще встречается выходной сигнал в таблице выходов, тем меньше единиц содержится в его коде.

Кодирование внутренних состояний двоичными символами оказывает существенное влияние на стоимость комбинационной части схемы автомата. Оптимальное кодирование даст минимальную стоимость, однако алгоритм эффективного кодирования неизвестен. Тем не менее, при кодировании внутренних состояний автомата используются различные эвристические алгоритмы, позволяющие, по крайней мере в некоторых случаях получить кодирование, близкое к оптимальному.

Д.А. Морозом предложен эвристический алгоритм кодирования внутренних состояний автоматов, основанный на минимизации суммарного числа изменений состояний элементов памяти автомата на всех переходах автомата.

При таком критерии уменьшается сложность схем, реализующих дизъюнкции на входах элементов памяти, т.е. минимизируется комбинационная схема автомата. Для оценки кодирования вводится коэффициент эффективности кодирования:

Kэф= W/P;

где Р - общее количество переходов автомата,

W - весовая функция : W=tij

где tij- расстояние Хэмминга между кодами состояний аi и аj, равное числу элементов памяти, изменяющих свое состояние на данном переходе.

Отметим, что при определении весовой функции суммирование производится по всем переходам автомата. Коэффициент эффективности позволяет оценить сложность комбинационной схемы автомата: чем меньше его значение, тем меньше сложность комбинационной схемы, и оптимальный вариант - Kэф=1.

Алгоритм состоит из следующих шагов:

1. Построить матрицу М, составленную из всех пар номеров (ar, br) переходов автомата.

2. Переставить строки в матрице таким образом, чтобы в каждой последующей строке содержался хотя бы один элемент из предыдущих строк.

3. Закодируем состояния из первой строки матрицы М следующим образом: Ka1=00...00, Kb1=00...01.

4. Вычеркнем из матрицы М первую строку с закодированными состояниями. Получим матрицу М'.

5. В начальной строке матрицы М' закодирован один элемент. Выберем из первой строки матрицы М' незакодированный элемент и обозначим его .

6. Построим матрицу M, выбрав из матрицы М' строки, содержащие у. Пусть В={1,2,...,f,...,F} - множество элементов из матрицы My, которые уже закодированы. Их коды обозначим через К1,К2,...,Кf,...,КF соответственно.



7. Для каждого Кgf (f=1, 2,...,F) найдем C1gf *-* множество кодов, отстоящих от Кgf на расстояние Хэмминга, равное 1 и еще не занятых для кодирования состояний автомата. Построим множество .



Если D1 =0, то строим новое множество , где - множество кодов, у которых кодовое расстояние с кодом Kf равно 2. Если и D2 =0, строим D3 и т.д., пока не найдем *.* Пусть .



8. Для каждого Кg находим wgf=|Kg- Kf|2 - расстояние Хэмминга между Кg и всеми используемыми кодами Kf(f=1, 2, ...,F).

9. Находим



10. Из выбираем К, у которого Wg=Wg min. Элемент у кодируем кодом К.



11. Из матрицы М' вычеркиваем строки, в которых оба элемента закодированы, в результате чего получаем новую матрицу, которую также обозначаем М’. Если в матрице М' не осталось ни одной строки, переходим к п. 12, иначе к п. 5.

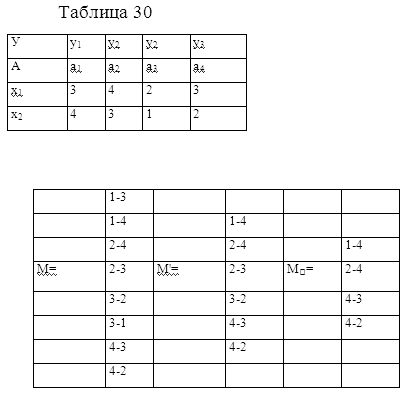
12. Вычисляем функцию где tij=|Kj-Kj|2.



13. Конец.

Оценкой качества кодирования по рассмотренному алгоритму может служить число K=W/P, где P - число переходов в автомате. Очевидно, что K>=1, причем, чем меньше значение K, тем лучше результат кодирования.

Рассмотрим пример кодирования автомата, заданного таблицей переходов и выходов.



1. Строим матрицу М:

Кодируем первую строку: K1=00;K2=01;

2. Вычеркиваем из матрицы М 1-ю строку (1-3) и 6-ю строку (3-1). Получаем матрицу М', в первой строке которой незакодирован элемент 4. Обозначим =4 и запишем матрицу M,. В матрице M закодированы 1 и З: В={1,3}={00,01} Находим W:



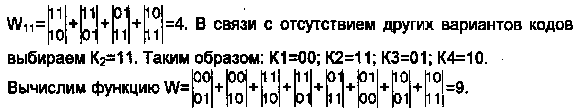
W10=|00| |10|=3; W11=|00| |11|=3;

|10|+|01| |11|+|01|

Выбираем K4=10.

Вычеркнем из М' строку (1-4). Получим новую матрицу М', в первой строке которой не закодирован элемент 2. Обозначим g=2 и запишем матрицу М2. В матрице М2 закодированы элементы 3, 4:

В



Вычислим K=W/P=9/8=1,125.

## **6. Обеспечение устойчивости функционирования цифровых автоматов. Гонки в автоматах**

Одна из главных задач, решаемых на этапе структурного синтеза синхронных цифровых автоматов с памятью, заключается в обеспечении устойчивости их функционирования. Понятие устойчивости связано с разработкой такой принципиальной электрической схемы автомата, которая обеспечивала бы его функционирование в соответствии с таблицей переходов и выходов автомата.

Неправильное функционирование автомата (неустойчивая его работа) связано с особенностями физической реализации логических элементов и элементов памяти его схемы, а также различными величинами задержек распространения сигнала в элементах и комбинационных схемах.

Рассмотрим процесс обеспечения устойчивости функционирования автомата более подробно. После поступления очередного входного сигнала и формирования сигналов возбуждения на входах элементов памяти автомат переходит в новое состояние. При этом происходит формирование новых сигналов возбуждения по цепям обратных связей (с выходов элементов памяти через логические элементы на входы элементов памяти), и автомат переходит в новое состояние и т. д.

Таким образом, автомат, в общем случае, не может остановиться в каком-то определенном состоянии и начинает функционировать в режиме генератора состояний. Для устранения такого эффекта используют синхросерию — последовательность специальных (обычно прямоугольных) сигналов, подаваемых на входы элементов памяти и разрешающих поступление очередных сигналов возбуждения на входы элементов памяти только с приходом очередного синхросигнала. При отсутствии синхросигнала сигнал возбуждения не поступает на вход элемента памяти, и элемент памяти цифрового автомата не переключается, т. е. остается в каком-то состоянии.

Практически подключение синхросерии осуществляется к специальным входам элемента памяти, обозначаемым символом C на рисунках и называемым синхровходами. Введение синхросерии, однако, не обеспечит устойчивого функционирования автомата, если не учитывать некоторые особенности. Если при переходе из одного состояния в другое должны изменять свои состояния сразу несколько элементов памяти, то между ними начинаются **состязания.** Тот элемент, который выиграет состязания, по цепям обратной связи может изменить сигналы на входах других элементов памяти до того, как они изменят свои состояния. Это может привести к переходу автомата в состояние, не предусмотренное графом.

Если в процессе перехода из состояния аm в состояние аs под действием входного сигнала xi автомат может оказаться в некотором промежуточном состоянии (аi, аk) в зависимости от того, какой из элементов памяти выиграл состязания, но, затем, при этом же входном сигнале перейдет в состояние аs, то такие состязания называются допустимыми, или не критическими. Но если произойдет переход в состояние аk (не предусмотренное графом переходов автомата) и правильность работы автомата нарушается, то такие состязания называются недопустимыми (критическими) илигонками.Пусть автомат должен выполнить переходы, изображенные на рис. 8.

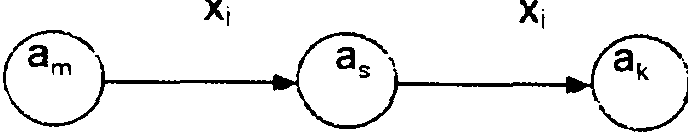


Рисунок 8

Возможны следующие ситуации:

а) если очередной синхросигнал на входы элементов памяти автомата поступает раньше, чем кончились переходные процессы в его комбинационной схеме возбуждения и элементах памяти после поступления входного сигнала хi, то возможно неправильное формирование сигнала возбуждения на входе одного или нескольких элементов памяти автомата, т. е. автомат может вместо перехода из состояния аm в состояние аs по входному сигналу xi осуществить ложный переход в некоторое состояние аt по тому же самому входному сигналу xi;

б) если длительность входного сигнала хi превышает длительность перехода автомата из состояния аm в состояние аs, то автомат может (при поступлении очередного синхросигнала) проскочить состояние аs и попасть сразу в состояние аk за счет двойного срабатывания автомата по входному сигналу хi. Иными словами, состояние аs может оказаться неустойчивым.

### **6.1 Методы устранения гонок в автоматах**

Для обеспечения устойчивого функционирования автомата нужно разнести во времени момент подачи информации на входы его элементов памяти и момент снятия информации с выходов элементов памяти. При таком разнесении формирование очередного сигнала возбуждения любого элемента памяти в момент появления синхросигнала осуществляется только по значениям состояний элементов памяти в предшествующий момент времени, а переходные процессы в элементах памяти не влияют на формирование сигнала возбуждения (выходы элементов памяти отключены).

Естественно, что период следования синхросигналов при этом должен выбираться исходя из учета окончания переходных процессов, связанных с задержками распространения входного для автомата сигнала по логическим элементам комбинационной схемы возбуждения. В результате устойчивость функционирования цифрового автомата может быть обеспечена, например, использованием двухэтажной памяти. В этом случае каждый элемент памяти дублируется, и перепись информации из нижнего элемента памяти в верхний осуществляется по отсутствию синхросигнала (рис. 9.).

Сигналы обратных связей, используемые для формирования функций возбуждения, и сигналы выходов автомата снимаются с выходов элемента памяти верхнего яруса. При такой организации памяти автомата отсутствует опасность формирования повторного сигнала возбуждения по одному и тому же синхросигналу и перехода автомата в новое состояние. Последнее связано с тем, что переход в рабочее состояние автомата завершается после окончания действия синхросигнала.

Однако использование двойной памяти автомата приводит к замедлению работы автомата. Если обычно период синхросигналов выбирается из расчета, что сигнал возбуждения элемента памяти успеет пройти по самой длинной цепочке логических элементов и переключить элемент памяти, то здесь период нужно удлинить по крайней мере на 3t где t*-* задержка распространения сигнала в логическом элементе (1t- на инвертор и 2t *-* на второй элемент памяти).

В случаях, когда из соображений быстродействия двухэтажную память использовать нельзя, прибегают к многофазной системе тактирования входных сигналов автомата. Так, для случая двухфазной синхронизации синхросериями CC1 и CC2 вместо одного входного сигналаxi**,** (рис. 8) используются два разных: хi • СС1 и xi • СС2 (рис. 10).

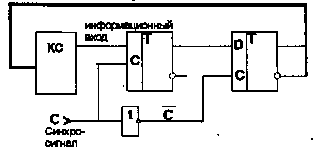


Рисунок 9



Рисунок 10

Таким способом устойчивость функционирования автомата обеспечивается автоматически. При двухфазной синхронизации необходимо, чтобы все дуги графа переходов автомата можно было бы разметить символами CC1 иCC2 так, что для любой вершины графа все выходящие из нее дуги отмечались бы символом одной синхросерии (например, СС1, а все дуги, заходящие в ту же вершину графа переходов автомата,— символом другой синхросерии (например, СС2). Если граф переходов автомата содержит контур нечетной длины, то такая разметка невозможна.

Однако ее можно сделать, преобразовав контур нечетной длины в контур четной длины, добавив дополнительную вершину или состояние автомата с пустым выходным сигналом. Задача преобразования произвольного графа с нечетными контурами к графу с четными контурами решается методами теории графов, в частности с использованием понятия цикпоматического числа графа и метода построения матрицы фундаментальных циклов графа.

Кроме описанных выше случаев, устойчивость функционирования цифрового автомата с памятью может быть частично обеспечена с помощью специальных мер, принятых относительно устранения в схеме автомата эффекта гонок. Это связано с тем, что элементы памяти имеют различные времена срабатывания. Различны также задержки сигналов возбуждения, поступающих на входы элементов памяти по цепочкам логических элементов различной длины.

Если при переходе автомата из одного состояния в другое, должны переключиться сразу несколько элементов памяти, то между ними начинаются гонки, (состязания), что может привести к неправильной работе автомата. В самом деле, при переходе автомата из состояния ai; в состояние aj на некоторый, хотя и очень короткий, промежуток времени может возникнуть промежуточное состояние автомата, отличное от ai и aj. Например, при переходе из ai (0110) в aj (1010) изменяют свое состояние два первых элемента памяти. Из-за состязаний может возникнуть состояние 1110 (или 0010), которое может привести к изменению состояния третьего или четвертого элемента памяти.

В последнем случае после окончания переходных процессов автомат уже не попадает в состояние aj. При использовании двухэтажной памяти гонки в автомате не возникают, так как изменение состояния автомата происходит в то время, когда синхросигнал отсутствует.

Следует заметить, что современная элементная база включает в себя элементы памяти с уже встроенной двухэтажной памятью (например, JK-триггер). Существует еще один способ устранения гонок в автоматах, связанный со специальным кодированием состояний автомата, которое называется противогоночным кодированием.

Частным случаем противогоночного кодирования является соседнее кодирование, при котором состояния автомата, связанные дугой на графе переходов, кодируются двоичными векторами, отличающимися друг от друга только в одном разряде. Для проведения соседнего кодирования в графе переходов автомата не должно быть контуров нечетной длины. Примеры графов переходов автоматов, состояния которых закодированы соседним образом, представлены на рис.

Отметим, что проблема состязаний и некоторые другие вопросы обеспечения устойчивости работы автоматов возникли лишь с появлением потенциальной элементной базы. Используемая ранее (в ЭВМ второго поколения) импульсно-потенциальная элементная база предусматривала применение статических триггеров со встроенной задержкой.

При этом величина задержки выбиралась большей длительности импульсного сигнала, поступающего на вход триггера. Тем самым переходные процессы формирования сигналов на выходах элементов памяти автомата начинались лишь после окончания входного импульсного сигнала и, следовательно, не оказывали воздействие на входы этих же элементов памяти по цепям обратной связи. Нечто подобное осуществляется теперь в потенциальной элементной базе введением не совпадающих во времени серий синхронизирующих сигналов, в особенности при организации двухэтажной памяти.

**Вывод**

В процессе выполнения курсовой работы мы ознакомились с:

## - основными понятиями структурных автоматов;

## - каноническим методом структурного синтеза автоматов;

## - теоремой Глушкова о структурной полноте;

## - основными этапами канонического метода структурного синтеза;

## - примерами канонического метода структурного синтеза;

## - кодированием состояний и выходов автомата и сложностью комбинационных схем;

## - обеспечением устойчивости функционирования цифровых автоматов;

## - гонкой в автоматах;

## - методами устранения гонок в автоматах.

**Литература**

1. Самофалов К.Г., Романкевич А.М., и др. Прикладная теория цифровых автоматов. - Киев. “Вища школа” 1987.

2. Соловьев Г.Н. Арифметические устройства ЭВМ. - М. “Энергия”. 1978.

3. Савельев А.Я. Прикладная теория цифровых автоматов - М. “Высшая школа”. 1987.

4. Каган Б.М. Электронные вычислительные машины и системы. - М. Энергоатомиздат. 1985.

5. Лысиков Б.Г. Арифметические и логические основы цифровых автоматов. - Минск. “Вышэйшая школа”. 1980.