**ЛИСТ ЗАДАНИЯ**

Изначальные данные:

1. Задан конечный цифровой автомат аналитическим методом следующими параметрами:
2. Множество букв входного алфавита: A= {0, 1};
3. Множество состояний автомата: Q= {q0, q1, q2, q3, q4, q5, q6, q7 };
4. Множество букв входного алфавита: V= {0,1};
5. Таблица переходов:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| δ | q0 | q1 | q2 | q3 | q4 | q5 | q6 | q7 |
| 0 | q1 | q0 | q3 | q6 | q5 | q7 | q2 | q4 |
| 1 | q6 | q1 | q7 | q4 | q0 | q2 | q5 | q3 |

1. Таблица выходов:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| λ | q0 | q1 | q2 | q3 | q4 | q5 | q6 | q7 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |

1. Граф-схема автомата:
2. Построить заданный автомат в базисе { и-не }.

Дата выдачи задания: 22 февраля 2010 г.

Дата согласования: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководители проекта:

Секирин А.И. \_\_\_\_\_\_\_\_

Поляков А.И. \_\_\_\_\_\_\_\_

Выполнил студент группы КИ-09д

Черновалов Д.И. \_\_\_\_\_\_\_\_

**РЕФЕРАТ**

стр. – 31 , рис. – 14 , табл. – 12

ЦИФРОВОЙ АВТОМАТ, ГРАФ-СХЕМА, КАРТЫ КАРНО, БАЗИС, ФУНКЦИОНАЛЬНАЯ СХЕМА, ELECTRONIC WORKBENCH, ЛОГИЧЕСКИЙ КОНВЕРТЕР, ЛОГИЧЕСКИЙ АНАЛИЗАТОР.

Целью данного проекта является разработка КЦА по заданным таблицам переходов и выходов автомата, а так же анализ работы функциональной схемы КЦА в программе EWB.

**СОДЕРЖАНИЕ**

Условные сокращения

Введение

1. Постановка задачи
2. Теоретические основы построения конечных автоматов
   1. Цифровой автомат
   2. Последовательность построения КЦА
   3. Формы представления функций алгебры логики
      1. Табличная форма представления
      2. СДНФ
      3. СКНФ
   4. Карты Карно
   5. Схемная реализация конечных автоматов
   6. Применение ЦА
3. Разработка и построение функциональной схемы КЦА
   1. Кодирование состояний автомата
   2. Минимизация функции выходов автомата
   3. Минимизация функции переходов автомата
   4. Перевод функций в заданный базис
   5. Теоретическая функциональная схема КЦА
4. Исследование работы КЦА в EWB
   1. Основные сведенья о программе EWB
   2. Интерфейс EWB
      1. Логические компоненты
         1. Логический элемент 2-И-НЕ
      2. Логический анализатор
      3. Логический конвертер

4.3 Результат работы функциональной схемы КЦА

4.3.1 Без применения устройства, осуществляющего задержку между переходами автомата

4.3.2 C применением устройства, осуществляющего задержку между переходами автомата

Заключение

Литература

Приложение А. Теоретическая функциональная схема КЦА

Приложение Б.1 Функциональная схема конечного автомата в EWB без применения устройства, осуществляющего задержку между переходами автомата

Приложение Б.2 Функциональная схема конечного автомата в EWB с применением устройства, осуществляющего задержку между переходами автомата

**УСЛОВНЫЕ СОКРАЩЕНИЯ**

|  |  |
| --- | --- |
| СДНФ | Совершенная Дизъюнктивная Нормальная Форма |
| СКНФ | Совершенная Конъюнктивная Нормальная Форма |
| ЭВМ | Электронно-вычислительная Машина |
| КЦА | Конечный Цифровой Автомат |
| EWB | Electronics Workbench |
| ЦА | Цифровой Автомат |
| ГРАФ | Графсхема |

**ВВЕДЕНИЕ**

В наступившем двадцать первом веке человечество не мыслит себя без цифровых технологий. Цифровое телевидение, цифровая фото и видео съёмка, цифровая сотовая и стационарная связь и т. д.; человек пользуется этими благами цивилизации, не задумываясь о том, как и что там работает. А ведь там происходят очень сложные действия над информацией, представленной в двоичной системе счисления.

Устройства, которые обеспечивают цифровую обработку информации, называются цифровыми. Эти устройства выполнены на цифровых интегральных микросхемах, что позволяет значительно уменьшить габариты устройств, массу, потребляемую мощность, стоимость и намного улучшить надёжность. Применение технологий больших и сверхбольших интегральных микросхем позволило перенести компьютеры из лабораторий и специальных учреждений в каждый дом. Помимо улучшения качества и надёжности аппаратуры интегральные микросхемы также влияют на производительность и качество труда человека, посредством того, что работа становится автоматизированной и, следовательно, увеличивается в разы вычислительная способность и уменьшается риск ошибки.

С появлением цифровых устройств в технике связи стало возможным предоставление огромного числа услуг, улучшилось качество связи и увеличился объём передаваемой информации.

Вершиной цифровой техники является микропроцессор. Мало кто знает, что в современных японских автомобилях автоматической подкачкой колёс управляет микропроцессор.

Микропроцессорные устройства широко используются в аппаратуре, которая обеспечивает обмен, обработку и распределение информации.

1. **ПОСТАНОВКА ЗАДАЧИ**

По имеющимся данным нужно разработать и построить функциональную схему КЦА и проверить правильность её работы. Для этого необходимо:

* составить таблицы истинности для выходов и переходов автомата;
* минимизировать функции выходов и переходов по СДНФ;
* перевести полученные минимальные функции в заданный базис;
* построить теоретическую схему каскадным принципом с использованием элементов 2-И-НЕ и с устройством, осуществляющим задержку между переходами автомата;
* построить теоретическую схему в программе EWB и проверить правильность её работы по временным диаграммам логического анализатора.

Построенную теоретическую схему, а так же схему из EWB необходимо предоставить в виде приложений, входящих в пояснительную записку. Вместе с пояснительной запиской нужно предоставить сменный носитель, на котором будет находиться файл с функциональной схемой КЦА, в формате EWB, и пояснительная записка.

1. **ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПОСТРОЕНИЯ КОНЕЧНЫХ АВТОМАТОВ**
   1. **Цифровой автомат**

Цифровым или конечным автоматом называется система которая может быть полностью задана следующими параметрами: S=<A,Q,V,



– множество букв входного алфавита;



– множество состояний автомата;



– множество букв выходного алфавита;



– функция переходов;



– функция выходов.



Автомат также может быть полностью задан графически (графсхемой).

Графсхема – топологический объект вершины которого соответствуют состоянию автомата, а дуги или ребра соответствуют переходам автомата

**Рис. 2.1** – Пример графсхемы №1

Различают направленные или ориентировочные и не ориентировочные графы

* не ориентировочные графы - содержат информацию о наличии переходов и не отображают их направление
* ориентировочные графы – учитывают направление переходов

Над дугами графа указывается информация о входном воздействии и состоянии выхода

q1

q2

q3

**Рис. 2.2** – Пример графсхемы №2

**2.2 Последовательность построения конечных автоматов**

Для построения КЦА необходимо:

* построить граф-схему автомата или таблицы выходов и переходов ЦА;
* Закодировать каждое состояние автомата;
* Построить таблицу истинности, содержащую вход, состояние и выход автомата. Минимизировать функцию выходов автомата;
* Построить таблицу истинности, содержащую вход, состояние и переходы автомата. Минимизировать функцию переходов автомата;
* Перевести полученные минимизированные функции в нужный базис.
  1. **Формы представления функций алгебры логики**

**2.3.1 Табличная форма представления**

Если мы имеем k переменных, то из них можно составить 2k комбинаций, а так как для каждой комбинации может задана своя функция, то число возможных функций .



Таблица 2.1 Пример таблицы истинности для 3 переменных



Каждая комбинация таблицы истинности – набор (конституанта).

* + 1. **СДНФ**

СДНФ (Совершенная Дизъюнктивная Нормальная Форма) — это такая ДНФ, которая удовлетворяет трём условиям:

• в ней нет одинаковых элементарных конъюнкций

• в каждой конъюнкции нет одинаковых пропозициональных букв

• каждая элементарная конъюнкция содержит каждую пропозициональную букву из входящих в данную ДНФ пропозициональных букв, причем в одинаковом порядке.

В функцию выписываются все конституанты, в которых функция принимает значение 1. Если переменная входит в конституанту как 0, то записывается ёё отрицание, если как 1, то без отрицания. в наборе соединяются операцией конъюнкция (\* или &), между наборами ставится операция дизъюнкция (+ или v).



Для того, чтобы получить СДНФ функции, требуется составить её таблицу истинности

Таблица 2.2

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

В ячейках строки́ отмечаются лишь те комбинации, которые приводят логическое выражение в состояние единицы. Далее рассматриваются значения переменных при которых функция равна 1. Если значение переменной равно 0, то она записывается с инверсией. Если значение переменной равно 1, то без инверсии.Первый столбец содержит **1** в указанном поле. Отмечаются значения всех четырёх переменных, это:



= 0 = 0 = 0 = 0



Нулевые значения — тут все переменные представлены нулями — записываются в конечном выражении инверсией этой переменной. Первый член СДНФ рассматриваемой функции выглядит так: Переменные второго члена:



= 0 = 0 = 0 = 1



в этом случае будет представлен без инверсии: Таким образом анализируются все ячейки . Совершенная ДНФ этой функции будет дизъюнкцией всех полученных членов (элементарных конъюнкций).Совершенная ДНФ этой функции:



* + 1. **СКНФ**

СКНФ (Совершенная Конъюнктивная Нормальная Форма) — это такая КНФ, которая удовлетворяет трём условиям:

* в ней нет одинаковых элементарных дизъюнкций
* в каждой дизъюнкции нет одинаковых пропозициональных букв
* каждая элементарная дизъюнкция содержит каждую пропозициональную букву из входящих в данную КНФ пропозициональных букв

В функцию выписываются все конституанты, на которых функция принимает значение 0. Если переменная входит в конституанту как 1, то записывается ёё отрицание, если как 0, то без отрицания. В наборе *xi* соединяются операцией дизъюнкция (+ или v), а между наборами ставится операция конъюнкция (\* или &).

Для того, чтобы получить СКНФ функции, требуется составить её таблицу истинности.

Таблица 2.3

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

В ячейках строки́ отмечаются лишь те комбинации, которые приводят логическое выражение в состояние нуля. Четвертый столбец содержит 0 в указанном поле. Отмечаются значения всех четырёх переменных, это:



= 0



= 0



= 1



= 1



В дизъюнкцию записывается переменная без инверсии если она в наборе равна 0 и с инверсией если она равна 1. Первый член СКНФ рассматриваемой функции выглядит так: Остальные члены СКНФ составляются по аналогии.



* 1. **Карты Карно**

**Карта Карно́** — графический способ минимизации переключательных (булевых) функций, обеспечивающий относительную простоту работы с большими выражениями и устранение потенциальных гонок. Карты Карно рассматриваются как перестроенная соответствующим образом таблица истинности функции.

Карта Карно может быть составлена для любого количества переменных, однако удобно работать при количестве переменных не более пяти. По сути Карта Карно - это таблица истинности составленная в 2-х мерном виде. Благодаря использованию кода Грея в ней верхняя строка является соседней с нижней, а правый столбец соседний с левым, т.о. вся Карта Карно сворачивается в фигуру тор(бублик). На пересечении строки и столбца проставляется соответствующее значение из таблицы истинности. После того как Карта заполнена, можно приступать к минимизации.

Если необходимо получить минимальную ДНФ, то в Карте рассматриваем только те клетки которые содержат единицы, если нужна КНФ, то рассматриваем те клетки которые содержат нули. Сама минимизация производится по следующим правилам (на примере ДНФ):

* объединяем смежные клетки содержащие единицы в область, так чтобы одна область содержала 2*n* (*n* целое число = 0…) клеток(помним про то что крайние строки и столбцы являются соседними между собой), в области не должно находиться клеток содержащих нули;



* область должна располагаться симметрично оси(ей) (оси располагаются через каждые четыре клетки);
* не смежные области расположенные симметрично оси(ей) могут объединяться в одну;
* область должна быть как можно больше, а кол-во областей как можно меньше;
* области могут пересекаться;
* возможно несколько вариантов накрытия.

Далее берём первую область и смотрим какие переменные не меняются в пределах этой области, выписываем конъюнкцию этих переменных, если неменяющаяся переменная нулевая, проставляем над ней инверсию. Берём следующую область, выполняем то же самое что и для первой, и т. д. для всех областей. Конъюнкции областей объединяем дизъюнкцией. Например(для Карт на 2-ве переменные)

* 1. **Схемная реализация конечных автоматов**

Конечный автомат состоит из 2-х основных блоков

1. Блок реализует функцию переходов;
2. Блок реализует функцию выходов.



τ



**Рис. 2.3** – Функциональная схема конечного автомата

Для того чтобы автомат осуществил переход по истечению времени τ, в блоке τ должен быть предусмотрен управляющий вход. С учётом этого входа блок τ представлен на рис. 2.3.

D

C



**Рис. 2.4** – Задерживающий элемент с управляющим входом

**D-триггер** (**D** от англ. *delay* — задержка)— запоминает состояние входа и выдаёт его на выход. D-триггеры имеют, как минимум, два входа: информационный **D** и синхронизации **С**. Сохранение информации в D-триггерах происходит в момент прихода активного фронта на вход С. Так как информация на выходе остаётся неизменной до прихода очередного импульса синхронизации, D-триггер называют также триггером с запоминанием информации или триггером-защёлкой. Рассуждая чисто теоретически, D-триггер можно образовать из любых RS- или JK-триггеров, если на их входы одновременно подавать взаимно инверсные сигналы.

* 1. **Применение ЦА**

Термин "автомат", как правило, используется в двух аспектах. С одной стороны, автомат - это устройство, выполняющее некоторые функции без непосредственного участия человека. В этом смысле мы говорим, что ЭВМ – автомат, так как после загрузки программы и исходных данных ЭВМ решает заданную задачу без участия человека. С другой стороны, термин "автомат" как математическое понятие обозначает математическую модель реальных технических автоматов. В этом смысле автомат представляется как "черный ящик", имеющий конечное число входов и выходов и некоторое множество внутренних состояний Q = {q1(t), q2(t), ..., qn(t)}, в которые он под действием входных сигналов переходит скачкообразно, т. е. практически мгновенно, минуя промежуточное состояние. Конечно, это условие не выполняется в реальности, так как любой переходный процесс длится конечное время.

Автомат называется конечным, если множество его внутренних состояний и множество значений входных сигналов – конечные множества.

На практике часто используется понятие цифрового автомата, под которым понимают устройство, предназначенное для преобразования информации. С общей точки зрения, процесс получения информации есть ни что иное, как процесс снятия неопределенности в результате того, что из некоторой совокупности возможных в данной конкретной ситуации явлений выделяется явление, фактически имевшее место.

Таким образом, в понятии информации существенно не само происшедшее явление, а лишь его отношение к совокупности явлений, которые могли произойти.

Устройства, служащие для преобразования дискретной информации, называются дискретными автоматами.

В современных дискретных автоматах принято обычно отождествлять буквы используемого стандартного алфавита с цифрами той или иной системы счисления.

В состав цифровых автоматов обязательно входят запоминающие элементы(элементы памяти). Выходные сигналы в таких автоматах формируются в зависимости от входных сигналов и состояний, в которых находятся элементы памяти. Поэтому дискретные автоматы принято называть также цифровыми автоматами.

Основным качеством, выделяющим дискретные автоматы из числа всех других преобразователей информации, является наличие дискретного множества внутренних состояний и свойства скачкообразного перехода автомата из одного состояния в другое. Скачкообразность перехода означает возможность трактовать этот переход как мгновенный, хотя для любого реально существующего автомата имеет место конечная длительность переходных процессов, так что требование скачкообразности перехода не удовлетворяется.

Второе допущение состоит в том, что после перехода автомата в произвольное состояние переход в следующее состояние оказывается возможным не ранее, чем через некоторый фиксированный для данного автомата промежуток времени τ > 0, так называемый интервал дискретности автомата. Это допущение дает возможность рассматривать функционирование цифрового автомата в дискретном времени. При построении автоматов с дискретным автоматным временем различают синхронные и асинхронные автоматы. В синхронных автоматах моменты времени, в которые оказывается возможным изменение состояния автомата, определяются специальным устройством – генератором синхронизирующих импульсов. Соседние моменты времени оказываются при этом обычно разделенными равными временными промежутками.

В асинхронных автоматах моменты переходов из одного состояния в другое заранее не определены и могут совершаться через неравные между собой промежутки времени.

Изменения состояний цифрового автомата вызываются входными сигналами, которые возникают вне автомата и передаются в автомат по конечному числу входных каналов. В отношении входных сигналов цифровых автоматов принимаются два допущения: во-первых, для любого цифрового автомата число различных входных сигналов обязательно конечно, а, во-вторых, входные сигналы рассматриваются как причина перехода автомата из одного состояния в другое и относятся к моментам времени, определяемым соответствующими им переходами.

Отметим, что при таком допущении входной сигнал рассматривается как мгновенный, хотя в действительности он имеет конечную длительность. Особо следует подчеркнуть, что реальный физический входной сигнал, вызывающий изменение состояния автомата в момент времени t, может кончиться до наступления этого момента, однако, тем не менее, он относится именно к текущему моменту времени t, а не к предыдущему (t –1).

Результатом работы цифрового автомата является выдача выходных сигналов, передаваемых из автомата во внешние цепи по конечному числу выходных каналов. В отношении выходных сигналов вводятся допущения, аналогичные допущениям для входных сигналов. Во-первых, число различных выходных сигналов для любого цифрового автомата всегда конечно. Во-вторых, каждому отличному от нуля моменту автоматного времени относится соответствующий ему входной сигнал. Реальный физический выходной сигнал y(t), отнесенный к моменту времени t, появляется всегда после соответствующего этому же моменту времени входного сигнала x(t). Что же касается момента времени t перехода автомата из состояния q(t–1) в состояние q(t), то сигнал y(t) может фактически появится либо раньше, либо позже этого момента.

В первом случае принимается, что выходной сигнал y(t) однозначно определяется входным сигналом x(t) и состоянием q(t–1) автомата в предыдущий момент времени, во втором случае сигнал y(t) однозначно определяется парой (x(t), q(t)). Будем считать, что для любого момента времени всегда имеет место лишь одна из этих возможностей (одновременно для всех переходов).

Цифровые автоматы, в которых выходной сигнал y(t) определяется парой (x(t),q(t – 1)), будем называть автоматами первого рода, а автоматы, в которых сигнал y(t) определяется парой (x(t), q(t)), – автоматами второго рода.

Цифровой автомат (первого или второго рода) называется правильным, если выходной сигнал y(t) определяется одним лишь его состоянием (q(t –1) или q(t)) и не зависит явно от входного сигнала x(t).

Автоматы первого рода обычно называют автоматами Мили, а автоматы второго рода – автоматами Мура.

Общая теория автоматов при сделанных выше допущениях разбивается на две большие части, которым присвоены названия абстрактной теории автоматов и структурной теории автоматов. Различие между ними заключается в том, что в абстрактной теории не учитываются структура как самого автомата, так и структуры его входных и выходных сигналов. Входные и выходные сигналы рассматриваются при этом просто как буквы двух фиксированных для данного автомата алфавитов: входного и выходного. Не интересуясь способом построения автомата, абстрактная теория изучает лишь те переходы, которые претерпевает автомат под воздействием входных сигналов, и те выходные сигналы, которые он при этом выдает.

В противоположность абстрактной теории, структурная теория автоматов учитывает структуры автомата и его входных и выходных сигналов. В структурной теории изучаются способы построения автоматов из нескольких элементарных автоматов, способы кодирования входных и выходных сигналов элементарными сигналами,передаваемыми по реальным входным и выходным каналам.

Таким образом, структурная теория автоматов является продолжением и дальнейшим развитием абстрактной теории. В частности, задача синтеза идеализированного (без учета переходных процессов) цифрового автомата естественным образом подразделяется на этапы абстрактного и структурного синтеза.

Частным случаем дискретных автоматов являются автоматы, обладающие лишь одним внутренним состоянием. Такие автоматы называются комбинационными схемами или автоматами без памяти. Работа таких автоматов состоит в том,что они сопоставляют каждому входному сигналу x(t) выходной сигнал y(t).

Абстрактная теория автоматов без памяти совершенно тривиальна, а структурная теория таких автоматов много легче, чем теория произвольных автоматов с памятью. Основная идея излагаемой методики синтеза автоматов состоит в том, чтобы еще на уровне абстрактной теории преодолеть основные затруднения, вызванные наличием памяти, а на уровне структурной теории свести задачу синтеза автомата к задаче синтеза комбинационных схем.

**3. РАЗРАБОТКА И ПОСТРОЕНИЕ ФУНКЦИОНАЛЬНОЙ СХЕМЫ КЦА**

* 1. **Кодирование состояний автомата**

Каждое состояние автомата кодируется произвольным образом. Так как состояний 8, значит, чтобы закодировать все состояния, нужно использовать 3-ёх разрядный код, так как только 3-ёх разрядным кодом можно перебрать до 23 состояний.



* 1. **Минимизация функции выходов автомата**

С использованием закодированных состояний, строится таблица истинности для выходов автомата. Отсутствующие выходы для не существующих закодированных состояний обозначаются \*.

Таблица 3.1 Таблица истинности для выходов автомата



Таблица 3.2 Карты Карно для V по СДНФ



* 1. **Минимизация функций переходов автомата**

Используя кодированные состояния, строится таблица истинности для переходов автомата. Отсутствующие переходы для не существующих кодированных состояний обозначаются \*.

Таблица 3.3 Таблица истинности для переходов автомата



**Минимизация функций (d1,d2, d3) по Картам Карно**

Таблица 3.4 Карты Карно для d1 по СДНФ



.



Таблица 3.5 Карты Карно для d2 по СДНФ



.



Таблица 3.6 Карты Карно для d3 по СДНФ



.



* 1. **Перевод функций в заданный базис**

При помощи основных алгебраических действий и (закона де Моргана), минимизируем функции в базис {и-не}.



* 1. **Теоретическая функциональная схема КЦА**

Теоретическая функциональная схема КЦА (Приложение А) строится каскадным принципом с помощью минимальных функций V, d0, d1, d2, логического двухходового элемента {и-не} и устройства, осуществляющего задержку между переходами автомата.

**4. ИССЛЕДОВАНИЕ РАБОТЫ КЦА В EWB**

* 1. **Основные сведенья о программе EWB**

Приложение Electronics Workbench представляет собой средство программной разработки и имитации электрических цепей. Интерфейс пользователя состоит из полоски меню, панели инструментов и рабочей области. Полоса меню состоит из следующих компонент: меню работы с файлами (**File**), меню редактирования (**Edit**), меню работы с цепями (**Circut**), меню анализа схем (**Analysis**), меню работы с окнами (**Window**), меню работы с файлами справок (**Help**).

Панель инструментов состоит из "быстрых кнопок", имеющих аналоги в меню, кнопок запуска и приостановки схем, набора радиоэлектронных аналоговых и цифровых деталей, индикаторов, элементов управления и инструментов.

* 1. **Интерфейс EWB**

Интерфейс **EWB** подобен реальному рабочему месту разработчика электронных схем. Все что вам необходимо для разработки и тестирования электронных схем собрано вместе и находится прямо перед вами или под рукой и легко доступно.

Самая большая центральная область - рабочее пространство (**workspace**) - место, где вы будете собирать и тестировать схемы.

Рядом с рабочим пространством находится линейка элементов ( **parts bin**).

А вот сверху вы найдете меню, иконки приборов и переключатель питания для активизации собранной схемы. Все действия, необходимые для работы в пакете **EWB**, такие как перемещение элементов схемы, соединение элементов между собой, перемещение по рабочей области экрана и линейке элементов осуществляются с помощью манипулятора типа **мышь**, что существенно облегчает работу с пакетом.

* + 1. **Логический элемент 2-И-НЕ**

Мнемоническое правило для И-НЕ с любым количеством входов звучит так: На выходе будет:

* "1" тогда и только тогда, когда ***хотя бы на одном*** входе действует "0",
* "0" тогда и только тогда, когда на ***всех*** входах действуют "1"



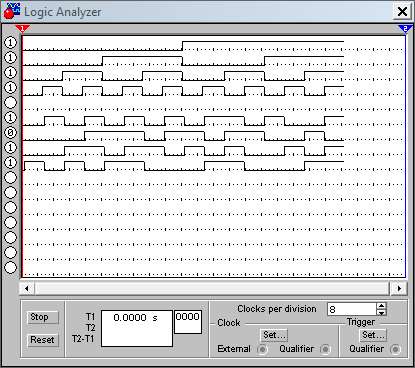
**Рис. 4.1** – Логический элемент 2-И-НЕ

**4.2.2 Логический анализатор**

Логический анализатор используется для наглядного представления и измерения параметров сигнала. Имеет 8 независимых входов для анализа сигналов, обозначенных цифрами от 1 до 8. Логический анализатор выдает информацию в виде временных диаграмм, аналогичных тем, которые были бы получены при подсоединении к реальной схеме осциллографа. Слева внизу на панели логического анализатора находятся клеммы для подключения к исследуемым точкам схемы. Справа от них отображается шестнадцатеричный код поданного на входы анализатора числа, причем вход "**1**" считается старшим разрядом числа, а вход "**8**" - младшим. Эта информация может оказаться полезной при синтезе счетчиков. Информация, поступающая на входы "**1**" .. "**8**", располагается на экране анализатора сверху вниз, т.е. вход "**1**" - самая верхняя строка на экране, вход "**8**" - самая нижняя. Также выдает их двоичное и шестнадцатеричное представление.



**Рис. 4.2** – Логический анализатор



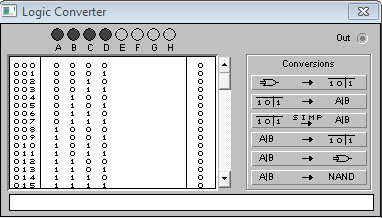
**Рис. 4.3** – Расширенное окно логического анализатора

* + 1. **Логический конвертер**

Внешний вид логического преобразователя показан на рис.4.8 На лицевой панели преобразователя показаны клеммы-индикаторы входов А, В,..., Н и одного выхода OUT, экран для отображения таблицы истинности исследуемой схемы, экран-строка для отображения ее булева выражения (в нижней части).



**Рис. 4.4** – Логический конвертер



**Рис. 4.5** – Расширенное окно логического конвертера

В правой части панели расположены кнопки управления процессом преобразования (CONVERSIONS) позволяющих осуществлять логические преобразования функции с числом переменных от 1 до 8:

Логический анализ га-входового устройства с одним выходом (входы исследуемого устройства подключаются к клеммам Н, а выход — к клемме OUT). В этом случае, используя кнопки управления, получим:

* булево выражение, реализуемое устройством
* таблицу истинности исследуемого устройства
* схему устройства на логических элементах без ограничения их типа
* минимизированное булево выражение;Синтез логического устройства по таблице истинности
* схему устройства только на логических элементах ИЛИ-НЕ;
* схему устройства только на логических элементах И-НЕ
  1. **Результат работы функциональной схемы КЦА**

**4.3.1 Без применения устройства, осуществляющего задержку между переходами автомата**

Подключаем к входам логического анализатора **a** , **x0 , x1 , x2 ,V, d1 , d2 , d3 .**

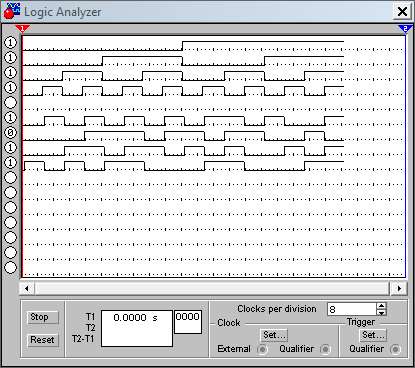
Пики диаграмм анализатора соответствуют логической единице, а впадины – логическому нулю.

Исходя из этого, сравнив функции переходов и выходов автомата с диаграммами логического анализатора, следует, что построенная функциональная схема КЦА работает верно.

Таблица 4.1 Таблица истинности для переходов и выходов автомата



a



**Рис. 4.6** – Результат работы функциональной схемы КЦА без применения устройства, осуществляющего задержку между переходами автоматаNиз приложения Б.1

**4.3.2 С применением устройства, осуществляющего задержку между переходами автомата**

Подключаем к входам логического анализатора **a, d1 , d2 , d3** , **V.**

Пики диаграмм анализатора соответствуют логической единице, а впадины – логическому нулю.

Исходя из этого, сравнив таблицы переходов и выходов автомата с диаграммами логического анализатора, следует, что построенная функциональная схема КЦА работает верно.



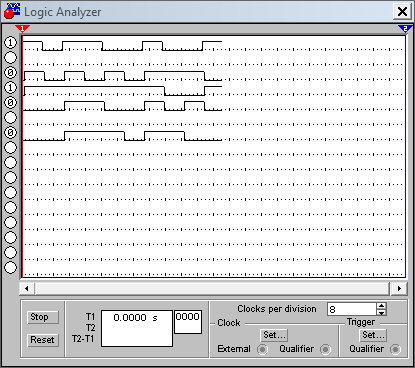
Таблица переходов: Таблица 4.2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| δ | q0 | q1 | q2 | q3 | q4 | q5 | q6 | q7 |
| 0 | q1 | q0 | q3 | q6 | q5 | q7 | q2 | q4 |
| 1 | q6 | q1 | q7 | q4 | q0 | q2 | q5 | q3 |

Таблица выходов: Таблица 4.3

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| λ | q0 | q1 | q2 | q3 | q4 | q5 | q6 | q7 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |

a



**Рис. 4.7** – Результат работы функциональной схемы КЦА с применением устройства, осуществляющего задержку между переходами автомата из приложения Б.2

**ЗАКЛЮЧЕНИЕ**

Курсовой проект был выполнен по дисциплине "Прикладная теория цифровых автоматов" по теме "Разработка конечного цифрового автомата".

По ходу работы была осуществлена кодировка 8 состояний автомата, для последующего построения таблиц истинности для переходов и выходов автомата. Произведена минимизация функций переходов и выходов автомата с помощью карт Карно. Построена теоретическая функциональная схема КЦА каскадным принципом, с использованием задерживающего устройства(D-триггера), осуществляющего задержку между переходами автомата, и логического элемента 2-И-НЕ. Построена функциональная схема КЦА в программе EWB, на основе теоретической схемы, с использованием логического конвертера, логического анализатора и логического элемента 2-И-НЕ.

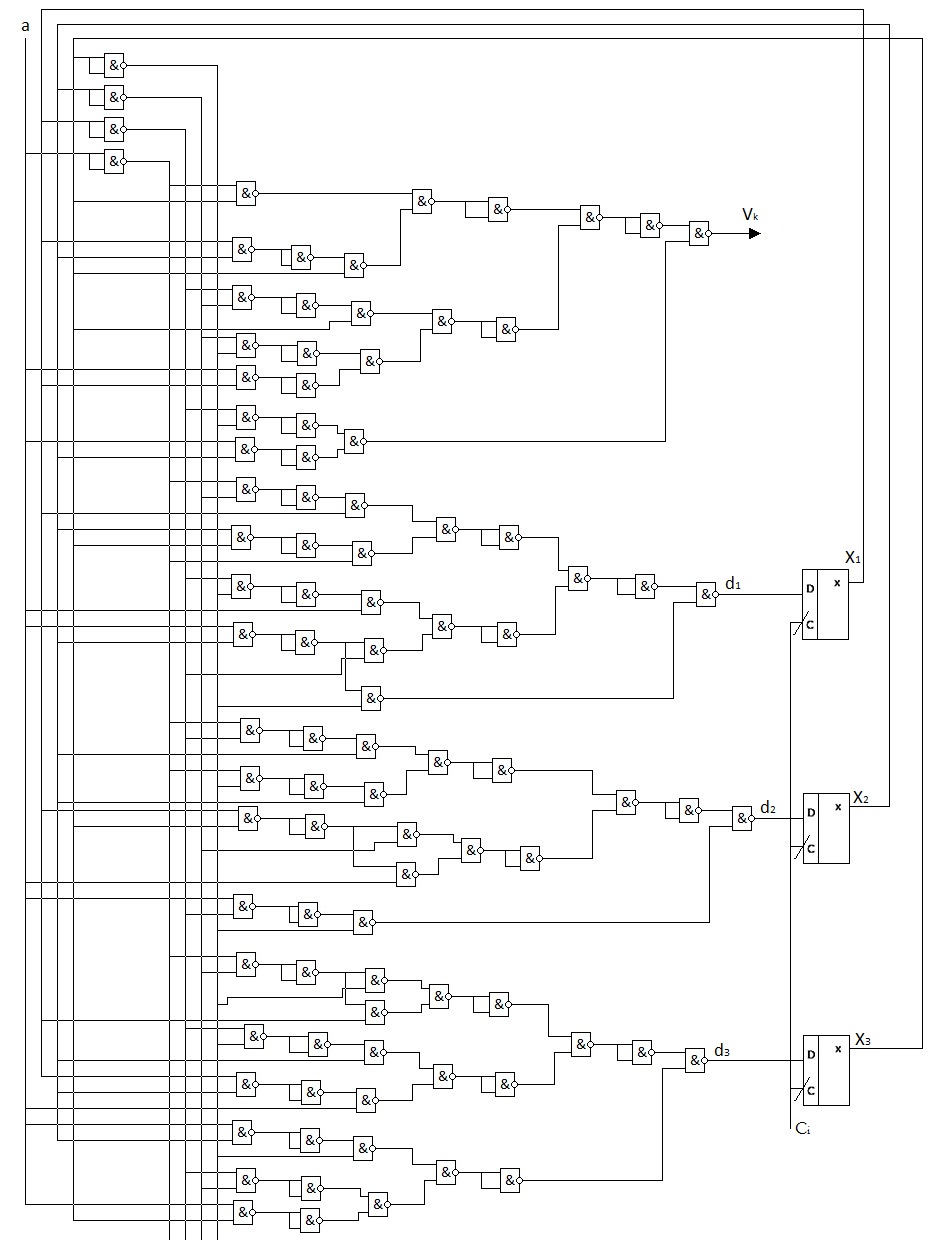
Проведён анализ работы схемы по диаграммам логического анализатора, по которым видно, что сигналы на диаграммах совпадают с функциями из таблиц истинности переходов и выходов автомата, и построенная схема на выходе выдаёт нужный сигнал.

**ЛИТЕРАТУРА**

1. Конспект лекций по курсу "Прикладная теория цифровых автоматов"
2. http://logic-bratsk.ru/radio/ewb/
3. Интернет источник: http://ru.wikipedia.org
4. Алгебраическая теория автоматов, языки и полугруппы. / М. А. Арбиба
5. http://www.rayax.ru

**Приложение А**

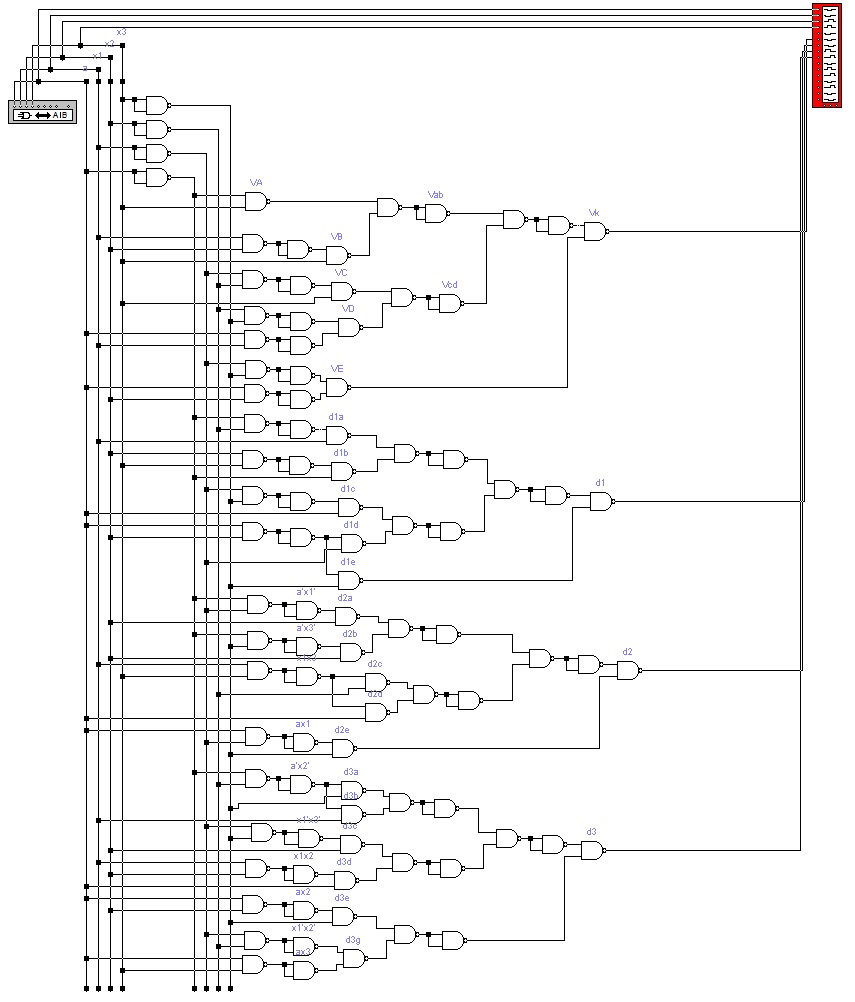
Теоретическая функциональная схема КЦА



**Приложение Б.1**

Функциональная схема конечного автомата в EWB

Без применения устройства, осуществляющего задержку между переходами автомата



**Приложение Б.2**

С применением устройства, осуществляющего задержку между переходами автомата

