

Министерство образования Республики Беларусь
БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ

Кафедра «Электропривод и автоматизация промышленных
установок и технологических комплексов»

А.А. Мигдалёнок

ПРОЕКТИРОВАНИЕ ДИСКРЕТНЫХ
СИСТЕМ УПРАВЛЕНИЯ

Методическое пособие
по выполнению курсовой работы
для студентов специальности
1-53 01 05 «Автоматизированные электроприводы»

Минск
БНТУ
2011

УДК 681.511.22-025.13:378.147.091.313(075.8)

ББК 32.965.6я7

М 57

Рецензенты:

Н.Н. Михеев, О.Ф. Опейко

Мигдалёнок, А.А.

М 57 Проектирование дискретных систем управления: методическое пособие по выполнению курсовой работы для студентов специальности 1-53 01 05 «Автоматизированные электроприводы» / А.А. Мигдалёнок. – Минск: БНТУ, 2011. – 55 с.

ISBN 978-985-525-533-9.

В методическом пособии отражены требования к курсовому проектированию и рассмотрено содержание разделов курсовой работы. На примере синтеза системы управления перемещением механизма по заданному циклу показаны методики проектирования дискретных систем управления. Рассмотрены вопросы построения формального описания дискретных управляющих устройств, его преобразования с целью получения оптимальных схемных решений. Приводятся методики минимизации памяти дискретных автоматов, кодирования внутренних состояний, определения функций возбуждения памяти и выходных функций. Даны рекомендации по построению функциональных и принципиальных схем устройства управления на бесконтактных элементах с реализацией памяти петлями обратных связей или на RS-триггерах.

УДК 681.511.22-025.13:378.147.091.313(075.8)

ББК 32.965.6я7

ISBN 978-985-525-533-9

© Мигдалёнок А.А., 2011

© БНТУ, 2011

1. ЗАДАНИЕ К КУРСОВОЙ РАБОТЕ. ПОРЯДОК ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ

При выполнении курсовой работы необходимо синтезировать схему управления перемещением механизма каретки между тремя положениями по заданному циклу.

Функциональная схема установки приведена на рис. 1.

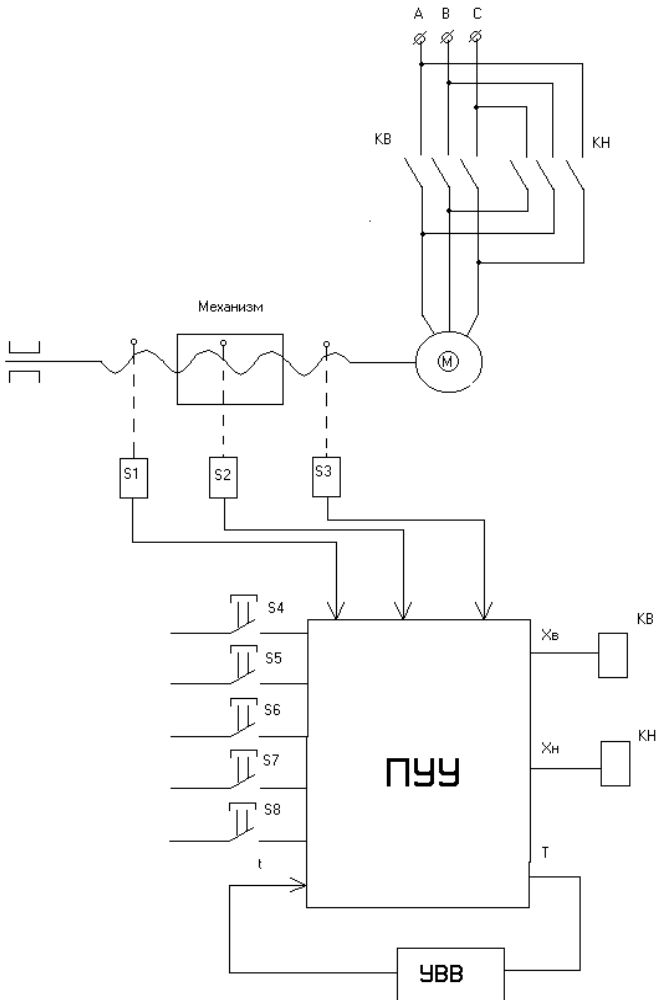


Рис. 1. Функциональная схема установки

На функциональной схеме введены следующие обозначения:

КВ, КН – контакторы движения «вперед», «назад»;

X_v, X_n – сигналы включения контакторов «вперед», «назад»;

$S1, S2, S3$ – конечные выключатели в положениях 1, 2, 3;

$S4$ – кнопка «Пуск» в режиме «Автомат» ($S4 = 1$ соответствует включенному состоянию кнопки «Пуск»);

$S5$ – кнопка «Стоп» в режиме «Автомат»;

$S6$ – кнопка переключения режимов. Если $S6 = 1$, то режим «Автомат», $S6 = 0$ – режим «Наладка»;

$S7$ – кнопка толчкового движения вперед в режиме «Наладка»;

$S8$ – кнопка толчкового движения назад в режиме «Наладка»;

ПУУ – проектируемое устройство управления;

УВВ – устройство выдержки времени;

Т – сигнал включения отсчета выдержки времени;

t – сигнал об окончании выдержки времени ($t = 1$, отсчет выдержки времени закончился).

Схему управления необходимо реализовать на бесконтактных элементах И-НЕ.

Принцип работы установки

Привод механизма каретки осуществляется асинхронным короткозамкнутым двигателем, реверсивное управление которым производится при помощи пускателей КВ (движение вперед) КН (движение назад). Схема может работать в режимах «Автомат» и «Наладка». Выбор режима работы производится переключателем $S6$.

В режиме «Автомат» выполнение автоматического цикла перемещений между положениями 1, 2, 3 начинается при подаче кратковременной команды «Пуск» кнопкой $S4$. Перемещение каретки заканчивается остановкой в исходном положении после отработки всего цикла. Аварийное отключение в режиме «Автомат» осуществляется кнопкой $S5$ «Стоп». Управление производится в функции положения механизма. Контроль положения механизма выполняется с помощью конечных выключателей $S1, S2, S3$, расположенных в положениях 1, 2, 3. При отработке цикла в заданном положении выполняется остановка механизма с выдержкой времени.

В режиме «Наладка» осуществляется перемещение каретки вперед или назад при нажатии и удержании кнопок $S7$ (движение вперед) или $S8$ (движение назад).

При выполнении курсовой работы необходимо выполнить следующие разделы.

1. Запись условий работы устройства управления.
2. Блочный синтез.
3. Синтез отдельных блоков.
 - 3.1. Построение автоматной таблицы выходов и переходов.
 - 3.2. Минимизация памяти автомата (блока).
 - 3.2.1. Построение треугольной таблицы.
 - 3.2.2. Нахождение максимальных групп совместимости.
 - 3.2.3. Построение таблицы покрытия и определение минимального класса совместимости.
 - 3.2.4. Построение минимизированной автоматной таблицы.
 - 3.3. Кодирование внутренних состояний автомата.
 - 3.3.1. Составление внешнего π -разбиения.
 - 3.3.2. Составление внутреннего π -разбиения на основе множеств порядка 1.
 - 3.3.3. Проверка однозначности кодирования.
 - 3.3.4. Построение автоматного графа для проверки возможности возникновения и устранения критических состояний элементов памяти.
 - 3.3.5. Построение минимизированной закодированной исправленной автоматной таблицы.
 - 3.4. Построение функциональной схемы устройства управления (блока).
 - 3.4.1. Определение функций возбуждения памяти.
 - 3.4.2. Определение выходных функций.
 - 3.4.3. Приведение функций в заданный базис.
 - 3.4.4. Построение функциональной схемы.
4. Выбор элементов. Построение принципиальной схемы устройства управления.
5. Проверка правильности работы устройства управления.

Исходными данными к курсовой работе является цикл перемещения каретки механизма между положениями 1, 2, 3. Выдержка времени в заданном положении обозначается буквой t .

Расчетно-пояснительная записка оформляется на стандартных листах печатной бумаги, заполняемых с одной стороны. На последнем листе приводится список используемой литературы, оформляемый в соответствии с ГОСТ 7.1–2003, а на первом – содержание записки по разделам и основным пунктам разделов. Титульный лист оформляется по установленному в БНТУ образцу.

Записка обязательно должна иллюстрироваться схемами, таблицами, рисунками, которые отражают результаты отдельных этапов проектирования. Таблицы и рисунки в записке должны иметь наименования и нумерацию.

Принципиальные схемы должны строиться в соответствии с ГОСТ 2.755–87 «Обозначения условные графические в электрических схемах», ГОСТ 2.710–81 «Обозначения буквенно-цифровые в электрических схемах».

Для планомерной работы студента над курсовой работой в задании указываются ориентировочные сроки выполнения основных разделов.

Готовую курсовую работу студент сдает к установленному сроку руководителю для проверки. Если имеются замечания о правильности и полноте выполненной работы, курсовая работа возвращается студенту для доработки.

Студент защищает курсовую работу перед комиссией из преподавателей кафедры.

2. ЗАПИСЬ УСЛОВИЙ РАБОТЫ УСТРОЙСТВА УПРАВЛЕНИЯ

При записи условий работы устройства управления дается конкретная словесная формулировка, которая устанавливает соответствие между входными и выходными сигналами, а также присваиваются обозначения входным и выходным сигналам.

В качестве примера выполнения курсовой работы рассмотрим синтез схемы управления перемещения механизма по циклу $2 - 1 - 2 - 3 - 2t - 1 - 2$.

На рис. 2 представлена графическая модель цикла работы механизма.

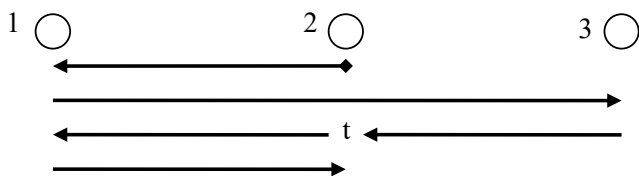


Рис. 2. Графическая модель цикла перемещения механизма

Перемещение каретки в направлении 1, 2, 3 принимают как перемещение вперед, в направлении 3, 2, 1 – как перемещение назад.

В словесной формулировке необходимо описать последовательность поступления входных сигналов устройства управления (с указанием их значений) и указать, какие выходные сигналы при этом формируются. Словесная формулировка условий работы устройства управления для рассматриваемого цикла будет иметь следующий вид.

В режиме «Автомат» (сигнал $S_6 = 1$) при нажатии на кнопку «Пуск» (сигнал $S_4 = 1$) при нахождении каретки в исходном положении 2 (сигнал $S_2 = 1$) включается контактор КН (назад) и включается двигатель М (сигнал $X_n = 1$). Начинается перемещение механизма из положения 2 в положение 1. При отпускании кнопки «Пуск» (сигнал $S_4 = 0$) контактор КН остается включенным (сигнал $X_n = 1$) и продолжается движение механизма в направлении назад. Конечный выключатель S_2 отключается (сигнал $S_2 = 0$), контактор КН остается включенным ($X_n = 1$) и продолжается движение механизма в направлении назад. При достижении кареткой положения 1 срабатывает конечный выключатель S_1 (сигнал $S_1 = 1$), отключается контактор КН (сигнал $X_n = 0$), включается контактор КВ (сигнал $X_b = 1$), и начинается перемещение каретки вперед в положение 2. Конечный выключатель S_1 выключается ($S_1 = 0$), контактор КВ остается включенным (сигнал $X_b = 1$), продолжается перемещение каретки вперед. При достижении кареткой положения 2 срабатывает конечный выключатель S_2 (сигнал $S_2 = 1$), контактор КВ остается включенным (сигнал $X_b = 1$), и продолжается перемещение вперед в положение 3. Конечный выключатель S_2 отключается (сигнал $S_2 = 0$), контактор КВ остается включенным (сигнал $X_b = 1$). При достижении кареткой положения 3 срабатывает конечный выключатель S_3 (сигнал $S_3 = 1$), отключается контактор КВ (сигнал $X_b = 0$), включается контактор КН (сигнал $X_n = 1$), и начинается перемещение механизма назад в положение 2. Конечный выключатель S_3 выключается (сигнал $S_3 = 0$), контактор КН остается

включенным (сигнал $X_n = 1$), продолжается перемещение назад в положение 2. При достижении кареткой положения 2 срабатывает конечный выключатель $S2$ (сигнал $S2 = 1$), отключается контактор КН (сигнал $X_n = 0$), отключается двигатель М, останавливается каретка, по сигналу $T = 1$ включается устройство выдержки времени УВВ. Происходит отсчет выдержки времени. По окончании отсчета выдержки времени на выходе устройства выдержки времени УВВ появляется сигнал $t = 1$. При этом включается контактор КН (сигнал $X_n = 1$) и происходит перемещение каретки назад в положение 1. Конечный выключатель $S2$ отключается (сигнал $S2 = 0$), контактор КН остается включенным (сигнал $X_n = 1$) и продолжается перемещение каретки в направлении назад. Реле времени отключается (сигналы $T = 0, t = 0$), контактор КН остается включенным (сигнал $X_n = 1$), и продолжается перемещение каретки в положение 1. При достижении кареткой положения 1 срабатывает конечный выключатель $S1$ (сигнал $S1 = 1$), отключается контактор КН (сигнал $X_n = 0$), включается контактор КВ (сигнал $X_v = 1$), и начинается перемещение каретки вперед в положение 2. Конечный выключатель $S1$ (сигнал $S1 = 0$) отключается, контактор КВ остается включенным (сигнал $X_v = 1$), и продолжается перемещение каретки в положение 2. При достижении кареткой положения 2 срабатывает конечный выключатель $S2$ (сигнал $S2 = 1$), отключается контактор КВ (сигнал $X_v = 0$), выключается двигатель, и каретка останавливается.

В режиме «Наладка» (сигнал $S6 = 0$) перемещение механизма происходит при нажатии и удержании кнопок «Вперед» (сигнал $S7 = 1$) или «Назад» (сигнал $S8 = 1$) независимо от конечных выключателей.

При нажатии на кнопку «Стоп» (сигнал $S5 = 1$) происходит остановка привода механизма в любом месте цикла.

3. БЛОЧНЫЙ СИНТЕЗ

В рассматриваемой схеме управления присутствует большое количество входных сигналов, что существенно затрудняет ее формальное описание и синтез. Так как сигналы разделены по функциональному значению, целесообразно для упрощения проектирования разделить схему на отдельные функциональные блоки и наметить обмен информацией между ними.



Рис. 3. Блочная схема проектируемого устройства

В зависимости от принципа действия устройство управления можно разделить на четыре функциональных блока. Для связи между блоками введем дополнительные внутренние сигналы. Дадим описание каждого блока по отдельности.

Блок Б1 – блок, формирующий команду на обработку автоматического цикла.

Входные сигналы блока Б1:

$S4$ – сигнал с кнопки «Пуск»;

$S5$ – сигнал с кнопки «Стоп»;

$S6$ – сигнал выбора режима;

c – сигнал об окончании цикла перемещений.

Выходные сигналы блока Б1:

d – сигнал разрешения обработки автоматического цикла.

Блок работает следующим образом. При нажатии на кнопку «Пуск» $S4$ и при отсутствии сигналов «Стоп» $S5$ и c на выходе блока формируется сигнал $d = 1$, поступающий в блок Б3. При нажатии на кнопку «Стоп» $S5$ или при поступлении сигнала об окончании

цикла $c = 1$ сигнал d на выходе блока становится равным нулю и отработка цикла перемещений прекращается.

Блок Б2 – блок режима «Наладка» выполняет перемещение каретки в режиме наладки.

Входные сигналы блока Б2:

$S6$ – сигнал выбора режима;

$S7$ – сигнал с кнопки движения вперед в режиме «Наладка»;

$S8$ – сигнал с кнопки движения назад в режиме «Наладка».

Выходные сигналы блокам Б2:

$Z_{\text{в}}$ – сигнал перемещения вперед;

$Z_{\text{н}}$ – сигнал перемещения назад.

При $S6 = 0$ и нажатой кнопке $S7$ происходит перемещение каретки вперед ($Z_{\text{в}} = 1, Z_{\text{н}} = 0$). При нажатой $S8$ каретка движется назад ($Z_{\text{в}} = 0, Z_{\text{н}} = 1$). Одновременное нажатие кнопок недопустимо и должно блокироваться.

Блок Б3 – блок автоматического цикла.

Входные сигналы

$S1, S2, S3$ – сигналы с конечных выключателей в положениях 1, 2, 3;

d – сигнал разрешения отработки автоматического цикла;

t – сигнал об окончании выдержки времени.

Выходные сигналы

$Y_{\text{в}}, Y_{\text{н}}$ – сигналы перемещения вперед, назад;

T – сигнал на включение выдержки времени;

c – сигнал об окончании цикла перемещений.

Блок Б3 начинает автоматическое выполнение цикла перемещений при поступлении сигнала «Начало цикла» (сигнал $d = 1$) из блока Б1. По включенному (сигнал $S = 1$) или по выключенному (сигнал $S = 0$) положению конечных выключателей $S1, S2, S3$ блок обеспечивает выполнение цикла заданных перемещений, формируя команды на движение вперед ($Y_{\text{в}} = 1$) или назад ($Y_{\text{н}} = 1$). Кроме этого в соответствующем положении в соответствии с циклом блок Б3 формирует сигнал на включение выдержки времени (сигнал $T = 1$). После отработки всего цикла перемещений на выходе блока формируется сигнал об окончании цикла ($c = 1$).

Блок Б4 – блок формирования сигналов силовых контакторов.

Производит формирование сигналов включения пускателей вперед ($X_{\text{в}} = 1$) или назад (сигнал $X_{\text{н}} = 1$) из сигналов, поступающих из блоков Б2 ($Z_{\text{в}}, Z_{\text{н}}$) и Б3 ($Y_{\text{в}}, Y_{\text{н}}$). Кроме этого обеспечивает защиту от

одновременного срабатывания контакторов X_B , X_H для исключения возможности короткого замыкания.

4. СИНТЕЗ ОТДЕЛЬНЫХ БЛОКОВ

Наиболее сложный алгоритм работы имеет блок автоматического цикла БЗ, поэтому рассмотрим синтез данного блока первым.

4.1. Построение автоматной таблицы выходов и переходов блока БЗ

Для построения автоматной таблицы на основании словесной формулировки составим первоначальную таблицу истинности, в которой указывается последовательность поступления входных сигналов и отражается связь между входными и выходными сигналами. Столбцы таблицы истинности соответствуют входным и выходным сигналам устройства управления, строки – входным наборам. Каждому входному набору ставится в соответствие определенное значение выходных сигналов.

Таблица 1

Первоначальная таблица истинности

№ пер.	Входные сигналы					Выходные сигналы			
	d	$S1$	$S2$	$S3$	t	Y_B	Y_H	T	c
1	0	0	1	0	0	0	0	0	0
2	1	0	1	0	0	0	1	0	0
3	1	0	0	0	0	0	1	0	0
4	1	1	0	0	0	1	0	0	0
5	1	0	0	0	0	1	0	0	0
6	1	0	1	0	0	1	0	0	0
7	1	0	0	0	0	1	0	0	0
8	1	0	0	1	0	0	1	0	0
9	1	0	0	0	0	0	1	0	0
10	1	0	1	0	0	0	0	1	0
11	1	0	1	0	1	0	1	1	0
12	1	0	0	0	1	0	1	0	0
13	1	0	0	0	0	0	1	0	0
14	1	1	0	0	0	1	0	0	0
15	1	0	0	0	0	1	0	0	0

16	1	0	1	0	0	0	0	0	1
17	0	0	1	0	0	0	0	0	0

Из первоначальной таблицы истинности видно, что проектируемое устройство управления является дискретным автоматом с памятью. Это связано с тем, что при одинаковых входных наборах на выходе устройства формируются различные выходные сигналы. Для формального описания таких устройств используется автоматная таблица выходов и переходов [1, с. 149–156].

Таблица переходов – это таблица, строки которой соответствуют исходным состояниям, столбцы – входным наборам, в клетках на пересечении строк и столбцов проставляется состояние, в которое переходит автомат из данного исходного состояния, определяемого строкой, под действием данной входной последовательности, определяемой столбцом.

Таблица выходов – это таблица, строки которой соответствуют исходным состояниям, столбцы – входным наборам, в клетках на пересечении строк и столбцов проставляется состояние выхода, формируемого в данном исходном состоянии под действием входной последовательности, определяемой столбцом.

Часто таблицу переходов и выходов совмещают и пользуются совмещенной автоматной таблицей.

Для полного и точного описания блока БЗ строится совмещенная избыточная автоматная таблица выходов и переходов, в которой любое изменение входного или выходного сигнала будет приводить к переходу в новое состояние. Выходные сигналы Y_v , Y_n будем отмечать по модели Мура (в отдельных столбцах), сигналы T , c – по модели Мили (в клетках автоматной таблицы, T – первый сигнал после номера перехода, c – второй сигнал после номера перехода) [1, с. 144–149]. Переход в последующее состояние происходит по следующей входной последовательности в соответствии с циклом перемещений (изменение сигналов $S1$, $S2$, $S3$) или при исчезновении сигнала $d = 1$ (аварийная ситуация). Поэтому в каждой строке будет заполнено три клетки: приход в данное состояние и два выхода из него.

Автоматная таблица выходов и переходов для блока БЗ по заданному циклу представлена в табл. 2.

Таблица 2

Автоматная таблица выходов и переходов блока БЗ

	t	t	t	t	t	t	t	t	t		У _В	У _Н
	S ₃		S ₃		S ₃		S ₃					
	S ₂				S ₂							
	S ₁											
d												
0				1				0			0 0	0 0
				00				00			0000	0000
1	2			1							0	0
	00			00							00	01
2	2							3				0
	00							00				00
3	4							3	0			1
	00							0000				0
4	4			5								0
	00			00								00
5	6			5						0		1
	00			00						00		0
6	6	7										0
	00	00										00
7	8	7								0		0
	00	00								00		1
8	8			9								0
	00			0								00
9				10						0		0
				0						0		0
10	11			10						0		0
	0			0						0		1
11	12	11									0	0
	0000										00	01
12	12							13				0
	00							00				00
13	14							13	0			1
	00							0000				0
14	14			15								0
	00			0								00
15				15						0		0
				01						0		0

A_0A_1	A_2	A_3A_4				$A_5A_6A_7A_8$				A_9A_{10}	A_{11}	$A_{12}A_{13}$
----------	-------	----------	--	--	--	----------------	--	--	--	-------------	----------	----------------

4.2. Минимизация памяти автомата

Целью минимизации автомата является получение эквивалентного автомата с наименьшим числом внутренних состояний [1, с. 165–173]. Чем меньше состояний в автоматной таблице, тем меньше элементов памяти необходимо для их кодирования. Минимизация заключается в нахождении совместимых состояний в автоматной таблице и их замене одним состоянием нового эквивалентного автомата.

Группа попарно совместимых состояний образует группу совместимости. Группы совместимости, покрывающие все состояния исходного автомата, образуют класс совместимости.

Для получения минимального автомата необходимо построить минимальный класс совместимости, т. е. такой, который содержит минимум групп совместимости, группы максимальны по размеру и данное разбиение покрывает все состояния исходного автомата. Поиск совместимых состояний сводится к поиску совместимых строк автоматной таблицы и замене их одной строкой.

При поиске совместимых состояний в автоматной таблице используют два признака:

1) из обоих состояний под действием любой допустимой входной последовательности осуществляется переходы в совместимые или непротиворечивые состояния;

2) этим состояниям должны соответствовать непротиворечивые или совместимые выходы.

При этом в зависимости от того, как накладываются переходы, два состояния могут быть условно совместимыми.

Два состояния называются условно совместимыми, если они совместимы при условии совместимости состояний, в которые автомат переходит из этих состояний.

Для определения совместимых состояний и построения минимального класса совместимости наиболее часто используется метод треугольной таблицы.

4.2.1. Построение треугольной таблицы

В треугольной таблице по горизонтали проставляются состояния от нулевого до предпоследнего, по вертикали – от последнего до первого (исключая нулевое состояние). Каждая клетка треугольной таб-

лицы (табл. 3) соответствует паре состояний. По автоматной таблице просматриваем все пары состояний в порядке, указанном в треугольной таблице справа на лево, снизу вверх и отмечаем совместимые (знак V), несовместимые (знак X) и условно совместимые состояния (записывается условие и проверяется, выполняется оно или нет).

Таблица 3

Треугольная таблица

1	X														
2	X	V													
3	X	X	X												
4	X	X	X	V											
5	X	X	X	$\sqrt{4,6}$	$\sqrt{4,6}$										
6	X	X	X	$\sqrt{4,6}$	V	V									
7	X	$\sqrt{2,8}$	$\sqrt{2,8}$	X	X	X	X								
8	X	X	V	X	X	X	X	V							
9	X	X	X	X	X	X	X	X	X						
10	X	V	V	X	X	X	X	V	V	X					
11	X	X	X	X	X	X	X	$\sqrt{8,12}$	$\sqrt{8,12}$	X	V				
12	X	X	X	X	X	X	X	$\sqrt{8,12}$	V	X	V	V			
13	X	X	X	X	X	$\sqrt{6,14}$	$\sqrt{6,14}$	X	X	X	X	X	X		
14	X	X	X	X	X	X	V	X	X	X	X	X	X	V	
15	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

4.2.2. Нахождение максимальных групп совместимости

Для нахождения максимальных групп совместимости просматриваем треугольную таблицу и объединяем в группы попарно совместимые между собой состояния. Все состояния, входящие в группу совместимости, должны быть между собой попарно совместимы!!! Группы совместимости должны отражать все совместимости в треугольной таблице.

Выписываем группы совместимости и обозначаем их буквами латинского алфавита.

- 6, 13, 14 – А
- 7, 8, 10, 11, 12 – В
- 6, 5, 13 – С
- 3, 4, 5, 6 – D
- 2, 7, 8, 10 – E
- 1, 2, 7, 10 – F

4.2.3. Построение таблицы покрытия и определение минимального класса совместимости

Минимальный класс совместимости представляет собой наилучший вариант объединения состояний исходной автоматной таблицы для получения таблицы с минимальным количеством внутренних состояний. Для определения минимального класса совместимости строится таблица покрытия. Строки таблицы покрытия соответствуют группам совместимости, столбцы – исходным состояниям. На пересечении строк (групп совместимости) со столбцами (исходными состояниями) отмечается вхождение данного состояния в данную группу совместимости.

На основании таблицы покрытия составляется формула покрытия. Для этого для каждого состояния записываем дизъюнкцию тех групп совместимости, в которые входит данное состояние. Дизъюнкции для каждого состояния связываются конъюнктивно. К полученной формуле применяются законы алгебры-логики, раскрываются все скобки и получают в результате дизъюнктивно связанные классы совместимости. Из полученных классов необходимо выбрать тот, который содержит наименьшее количество групп совместимости. Это и будет минимальный класс совместимости.

Сстроим таблицу покрытия (табл. 4).

Таблица 4

Таблица покрытия

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
A							X							X	X	
B								X	X		X	X	X			
C						X	X							X		
D				X	X	X	X									

E		X				X	X		X				
F	X	X				X			X				

Формула покрытия будет иметь следующий вид:

$$f = A \cdot (A + C) \cdot B \cdot B \cdot (B + E + F) \cdot (B + E) \cdot (B + E + F) \times \\ \times (A + C + D) \cdot (C \cdot D) \cdot D \cdot D \cdot (E + F) \cdot F = A \cdot B \cdot D \cdot F.$$

Как видно из таблицы покрытия состояния 0, 9 и 15 не вошли ни в одну из групп совместимости. Эти состояния будут образовывать отдельные группы совместимости, состоящие из одного состояния (эти состояния удалять нельзя!!!).

В полученном минимальном классе необходимо выполнить редукцию, т. е. убрать повторяющиеся состояния. Состояние 6 входит в группы А и D. Удалим данное состояние из группы А. Состояния 7, 10 входят в группы В и F. Удалим данные состояния из группы F.

При удалении состояний необходимо, чтобы в оставшихся группах совместимости не нарушались условия совместимости.

Запишем минимальный класс совместимости и обозначим полученные группы совместимости новыми номерами состояний эквивалентного автомата.

A	- 6, 13, 14	13, 14	- 5
B	- 7, 8, 10, 11, 12	7, 8, 10, 11, 12	- 3
D	- 3, 4, 5, 6	3, 4, 5, 6	- 2
F	- 1, 2, 7, 10	1, 2	- 1
	- 0	0	- 0
	- 9	9	- 4
	- 15	15	- 6

По полученным новым состояниям эквивалентного автомата можем построить минимизированную автоматную таблицу.

4.2.4. Построение минимизированной автоматной таблицы

Для построения минимизированной автоматной таблицы необходимо в исходной автоматной таблице старые номера состояний

заменить на новые и строки с одинаковыми номерами объединить в одну путем наложения (табл. 5).

Минимизированная автоматная таблица

	t		t			t		t			t		t			Y _B	Y _H			
	S ₃		S ₂			S ₃		S ₃			S ₂									
	S ₂			S ₁			S ₃			S ₃										
	d																			
0					1 ₁					0 ₀₀				0 ₀₀	0 ₀₀	0 ₀₀	0	0		
1	1 ₀₀				1 ₀₀					2 ₀₀				0 ₀₀		0 ₀₀	0	1		
2	2 ₀₀				2 ₀₀					2 ₀₀	0 ₀₀			0 ₀₀		0 ₀₀	1	0		
3	3 ₀₀	3 ₀₀			3 ₀₀					3 ₀₀				5 ₀₀		0 ₀₀	0	1		
4					3 ₁₀	4 ₁₀										0 _{~0}	0	0		
5	5 ₀₀				6 ₀₀					5 ₀₀	0 ₀₀						0 ₀₀	1	0	
6					6 ₀₁											0 _{~0}	0	0		
	A ₀	A ₁	A ₂		A ₃	A ₄				A ₅	A ₆				A ₇	A ₈		A ₉	A ₁₀	A ₁₁

Правильность построения минимизированной автоматной таблицы можно проверить следующим образом. При объединении строк исходной автоматной таблицы в клетках на соответствующих наборах должны накладываться одинаковые номера. Если накладываются два разных номера, следовательно, в группу совместимости, соответствующую данному новому состоянию, входят несовместимые состояния. Это может произойти из-за неправильной записи группы совместимости или из-за нарушения условий совместимости.

Кроме этого, после построения минимизированной таблицы можно проверить, выполняются ли все переходы в соответствии с циклом перемещений, отмеченные в исходной автоматной таблице. Обычно в минимизированной автоматной таблице последовательность переходов отображается стрелками. Если все переходы выполняются, следовательно, минимизированная автоматная таблица построена верно (см. табл. 5).

4.3. Кодирование внутренних состояний автомата

При кодировании каждому внутреннему состоянию автомата присписывается свой, неповторяющийся набор внутренних сигналов элементов памяти (определяется код состояния) [1, с. 177–183].

В зависимости от значения данной переменной кодирования все внутренние состояния автомата разделяются на два блока. В первый блок входят состояния, в которых данная переменная кодирования равна единице, во второй блок – состояния, в которых переменная кодирования равна нулю.

Разбиением множества внутренних состояний P называют совокупность подмножеств состояний P_i таких, что их объединение есть P . Подмножества называются блоками разбиения и отмечаются чертой.

Произведением двух разбиений Π_i и Π_j называют такое третье разбиение Π_{ij} , каждый блок которого образован пересечением блоков разбиений Π_i и Π_j .

Для однозначного кодирования необходимо, чтобы пересечение разбиений, в соответствии с которыми определяется распределение переменных кодирования, было равно нулевому разбиению.

В нулевом разбиении каждое состояние образует один блок, и, следовательно, все состояния между собой различимы.

При кодировании могут использоваться внешнее и внутреннее разбиения.

При кодировании состояний по внешнему разбиению значения переменных кодирования в соответствующих состояниях приравниваются к значениям выходных переменных в этих же состояниях. При этом, чем больше совпадают значения переменных кодирования с выходными сигналами, тем проще комбинационная схема выходов.

При кодировании по внутреннему разбиению в один блок включаются наиболее близкие по переходам состояния. Чем ближе по переходам состояния, включаемые в один блок, тем проще схема возбуждения памяти.

4.3.1. Составление внешнего π -разбиения

Выполним кодирование по внешнему разбиению (приравняем значения переменных кодирования $p1$ и $p2$ к значениям Y_v и Y_n . Получим разбиения следующего вида:

$$\begin{aligned} \pi_{1Y_v} &= \overline{0, 1, 3, 4, 6} \quad \overline{2, 5} \\ \pi_{2Y_n} &= \overline{0, 2, 4, 5, 6} \quad \overline{1, 3} \end{aligned}$$

Определим пересечение полученных разбиений.

$$\pi_1 \cdot \pi_2 = \overline{(0, 1, 3, 4, 6 \quad 2, 5)} \cdot \overline{(0, 2, 4, 5, 6 \quad 1, 3)} = \overline{0, 4, 6 \quad 1, 3 \quad 2, 5}.$$

Из полученного пересечения видно, что неразличимыми по внешнему разбиению остались 1 и 3 состояния, 2 и 5 состояния и 0, 4 и 6 состояния (имеют одинаковый код). При этом для того чтобы различить три неразличимых состояния, необходимо ввести еще две переменные кодирования. Таким образом, при использовании внешнего разбиения для кодирования всех семи состояний необходимо четыре переменные кодирования. Данный вариант кодирования является неоптимальным, т. к. для кодирования семи состояний достаточно трех переменных кодирования. Поэтому в данном случае целесообразно использовать только внутреннее разбиение для трех переменных кодирования.

В общем случае, если при кодировании внутренних состояний автомата с использованием внешнего разбиения достаточно трех переменных кодирования, необходимо использовать внешнее разбиение. Если трех переменных недостаточно, необходимо использовать внут-

реннее разбиение и три переменные кодирования. Это верно для устройств, которые реализуются на бесконтактных элементах и содержат в минимизированной автоматной таблице не больше восьми состояний. Если состояний больше или схема реализуется на контактных элементах, необходимо использовать внешнее разбиение.

4.3.2. Составление внутреннего π -разбиения на основе множеств порядка единица

Для уменьшения переменных кодирования для нашей схемы используем внутреннее разбиение.

Воспользуемся методом на основе множеств порядка единица.

Множеством порядка единица для состояния pi называется множество тех состояний, из которых возможен непосредственный переход в состояние pi .

Порядок определения внутреннего разбиения:

1) для каждого состояния выписываем множество порядка единица. Выписываем номера состояний, из которых переходим в данное состояние;

2) выписываем объединенные наборы таких состояний, которые имеют максимальное пересечение множеств порядка единица;

3) из объединенных наборов составляем двухблочное разбиение таким образом, чтобы оно содержало в разных блоках состояния, не различимые по внешнему разбиению;

4) проверяем, равно ли пересечение полученных разбиений нулевому разбиению.

Запишем множества порядка единица:

0 – 1, 2, 3, 4, 5, 6

1 – 0, 1

2 – 1, 2

3 – 2, 3, 4

4 – 3, 4

5 – 3, 5

6 – 5, 6

Запишем объединенные наборы:

0, 3 – пересекаются по трем состояниям;

0, 1 0, 2 0, 4 0, 5 0, 6 3, 4 – пересекаются по двум состояниям.

Примем разбиения следующего вида:

$$\pi_1 = \overline{0, 5, 6} \quad \overline{1, 2, 3, 4};$$

$$\pi_2 = \overline{0, 1, 4, 6} \quad \overline{2, 3, 5};$$

$$\pi_3 = \overline{0, 1, 2} \quad \overline{3, 4, 5, 6}.$$

4.3.3. Проверка однозначности кодирования

Для проверки однозначности кодирования необходимо определить произведение полученных разбиений.

$$\pi_1 \cdot \pi_2 \cdot \pi_3 = \overline{0} \quad \overline{1} \quad \overline{2} \quad \overline{3} \quad \overline{4} \quad \overline{5} \quad \overline{6}.$$

Так как произведение полученных разбиений равно нулевому разбиению (каждое состояние образует отдельный блок), следовательно, каждое состояние имеет свой индивидуальный неповторяющийся код.

Распределим значения переменных кодирования по состояниям (табл. 6).

Таблица 6

Минимизированная закодированная автоматная таблица

	t		t		t		t		t		t		t		t													
	S ₃		S ₂				S ₃		S ₁				S ₂		S ₃		Y _a	Y ₁	p ₁	p ₂	p ₃							
d																												
0					1						0					0	0	0	0	0	0							
1	1				00						2					0				0	0							
2	00				00						0000					00				0	00							
3	00	3			00						2	0				0				0	00							
4	00	00	3		00						0000					0				0	00							
5	00	00	00	3	4						5					0				0	00							
6	00	00	00	00	10~0						00					~0				0	00							
					3	4										0				0	00							
					1010											~0				0	00							
					6						5	0								0	00							
					0~						0000									0	00							
					6											0				0	00							
					01											~0				0	00							
	A ₀ A ₁	A ₂	A ₃ A ₄			A ₅ A ₆					A ₇ A ₈	A ₉	A ₁₀ A ₁₁															

4.3.4. Построение автоматного графа для проверки возможности возникновения и устранения критических состояний элементов памяти

Полученную минимизированную закодированную автоматную таблицу необходимо проанализировать с точки зрения возможности возникновения состязаний элементов памяти при переходах [1, с. 209–212].

Состязания элементов памяти возникают в том случае, если при переходе из состояния в состояние меняют свое значение два и более разряда кода. При этом за счет наличия задержек на срабатывание элементов памяти один разряд кода может переключиться быстрее. Это приводит к появлению промежуточных кодов, которые могут привести к нарушению алгоритма работы автомата.

Состязания элементов памяти могут быть критическими и некритическими.

Критические – это состязания, при которых нарушается алгоритм работы устройства (автомат застревает в непредусмотренном состоянии или осуществляются непредусмотренные переходы с формированием другой выходной последовательности).

Некритические – это состязания, при которых автомат переходит в другое используемое состояние или в неиспользуемое состояние, но при этом алгоритм работы не нарушается, формируется заданная выходная последовательность.

Так как проектируемое устройство относится к асинхронным автоматам, то необходимо устранить как критические, так и некритические состязания элементов памяти.

Для определения возможности возникновения критических состязаний элементов памяти по закодированной минимизированной автоматной таблице строится автоматный граф. Вершины автоматного графа соответствуют состояниям автомата, дуги – переходам из состояния в состояние. Для дискретного автомата Мура у вершин графа проставляются коды состояний и соответствующие им выходные наборы, над дугами отмечаются входные наборы, под действием которых осуществляется данный переход. Просматриваются все переходы и выписываются те из них, на которых меняют свое значение два и более разряда кода, т. е. возникает состязание элементов памяти. Граф состояний и переходов представлен на рис. 4.

Из автоматного графа видно, что состязания элементов памяти возникают на следующих переходах:

Переход	Переключение кода	Наборы
2 → 0	110 → 000	A6, A7, A11
3 → 0	111 → 000	A8, A9, A10, A11
4 → 0	101 → 000	A7
5 → 0	011 → 000	A6, A11

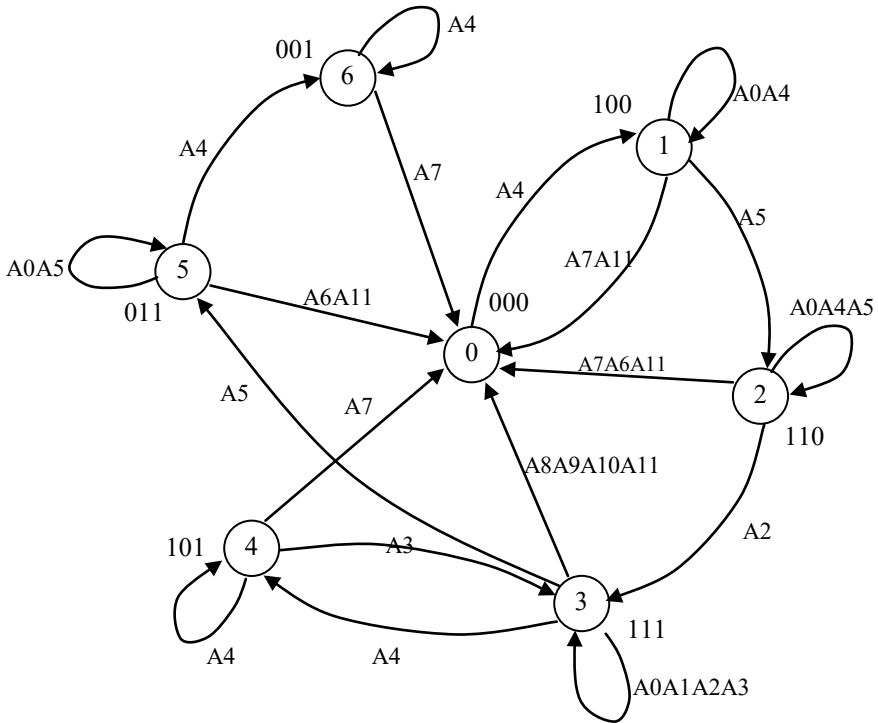


Рис. 4. Граф состояний и переходов

Для устранения критических состязаний элементов памяти выполняют следующие действия:

1) выполняют вместо заданного перехода переход через другое неустойчивое состояние с последовательной сменой кода;

2) если такой переход выполнить невозможно, вводят дополнительные состояния с неиспользуемыми кодами и выполняют переход через эти состояния;

3) если неиспользуемые коды отсутствуют, вводят дополнительную переменную кодирования для получения дополнительных кодов и выполняют переход через дополнительные состояния.

Устраним состязания, выполнив переходы через другие неустойчивые состояния. Для этого в тех состояниях, через которые будем осуществлять переходы, наборы, на которых исправляются переходы, должны быть не заняты. Выполним переходы следующим образом.

Новые переходы	Переключение кода
$2 \rightarrow 1 \rightarrow 0$	$110 \rightarrow 100 \rightarrow 000$
$3 \rightarrow 2 \rightarrow 1 \rightarrow 0$	$111 \rightarrow 110 \rightarrow 100 \rightarrow 000$
$4 \rightarrow 1 \rightarrow 0$	$101 \rightarrow 100 \rightarrow 000$
$5 \rightarrow 6 \rightarrow 0$	$011 \rightarrow 001 \rightarrow 000$

Строим исправленный автоматный граф (рис. 5).

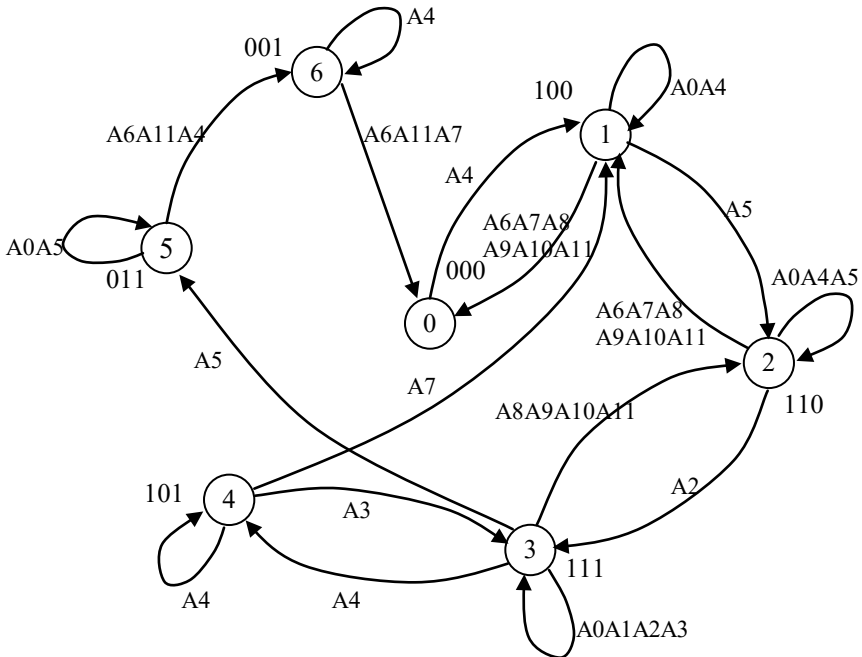


Рис. 5. Исправленный граф состояний и переходов

4.3.5. Построение минимизированной закодированной исправленной автоматной таблицы

На основании исправленного графа можем построить исправленную минимизированную закодированную автоматную таблицу (табл. 7).

Таблица 7

Исправленная закодированная минимизированная автоматная таблица

	<u>t</u>	<u>t</u>	<u>t</u>	<u>t</u>	<u>t</u>	<u>t</u>	<u>t</u>	<u>t</u>	<u>t</u>	<u>t</u>										
	S ₃		S ₂			S ₃		S ₁				S ₂		S ₃						
<u>d</u>																				
0				1			0					0 0	0 0	0 0	0 0	0 0	0 0	0 0	0 0	
1	1			1			2 0					0 0	0 0	0 0	0 0	0 0	0 1	1 0	0 0	
2	2	3		2			2 1					1 1	1 1	1 1	1 1	1 1	1 0	1 1	1 0	
3	3	3	3	3 4			5					2	2	2 2	2 2	0 1	1 1	1 1	1 1	
4				3 4								1				0 0	1 0	1 0	1 1	
5	5			6			5 6								6	1 0	0 0	1 1	1 1	
6				6			0					0			0	0 0	0 0	0 0	0 1	
	A ₁ A ₃	A ₂	A ₃ A ₄				A ₅ A ₆					A ₇ A ₈	A ₉	A ₁₀ A ₁₁						

4.4. Построение функциональной схемы устройства управления

Для построения функциональной схемы устройства управления необходимо записать аналитические выражения для функций возбуждения памяти и выходных функций. В бесконтактных схемах память может реализовываться с помощью петель обратных или на триггерах [1, с. 183–188].

4.4.1. Определение функций возбуждения памяти и выходных функций при реализации памяти петлями обратных связей

Для построения функциональной схемы на бесконтактных элементах И-НЕ с реализацией памяти петлями обратных связей необходимо выполнить следующие действия:

1) построить карту Карно для каждой переменной кодирования. При этом для реализации петель обратных связей в карту Карно включаются как входные переменные, так и внутренние переменные;

2) по кодам распределить состояния автоматной таблицы по строкам карты Карно;

3) заполнить клетки карты Карно. Для этого необходимо номер состояния в соответствующей клетке заменить на значение данной переменной в данном состоянии;

4) в соответствии со стандартными правилами построить контуры и записать минимизированные выражения для функций возбуждения памяти;

5) если необходимо, построить карты Карно для выходных переменных (если применялось только внутреннее разбиение). При этом для модели Мура указывается зависимость выходов только от внутренних переменных, для модели Мили – зависимость выходов от входов и внутренних переменных. По данным картам Карно записать минимизированные выражения для функций выхода;

6) полученные выражения преобразовать в заданный базис;

7) по полученным выражениям построить комбинационную схему возбуждения памяти (КСВП) и комбинационную схему выходов (КСВ).

Для того чтобы найти минимизированное выражение функции в виде ДНФ по карте Карно необходимо:

1) все единицы карты Карно охватить минимальным числом контуров. При этом, контуры должны быть максимальны по размеру, включать 2^k клеток (1, 2, 4, 8, 16, 32, 64 и т. д.) и быть симметричными относительно осей симметрии всей карты Карно, половины карты Карно, четверти и т. д.;

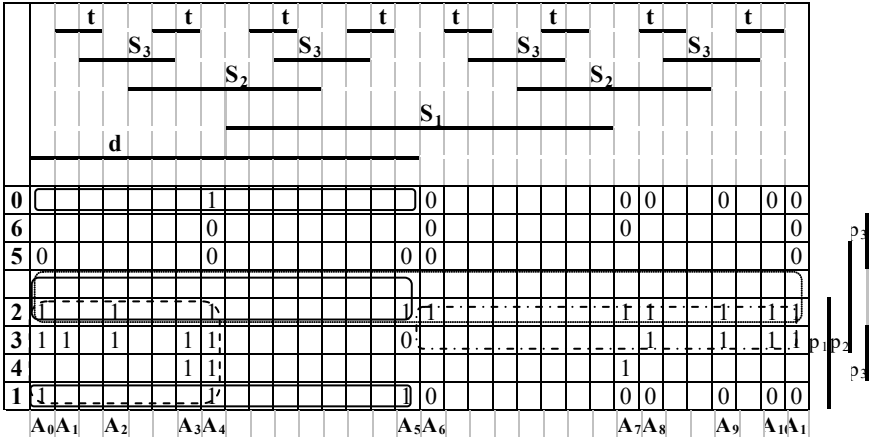
2) каждый контур дает в общее выражение минимизированной функции конъюнкцию тех переменных, которые в данном контуре не меняют своего значения. Причем если в контуре переменная в наборах равна 1, то в конъюнкцию записывается сама переменная, если равна 0 – ее инверсия;

3) минимизированное выражение получается как дизъюнкция конъюнкций, соответствующих каждому контуру.

Построим карты Карно для переменных p_1, p_2, p_3 и определим функции возбуждения памяти (табл. 8, 9, 10).

Таблица 8

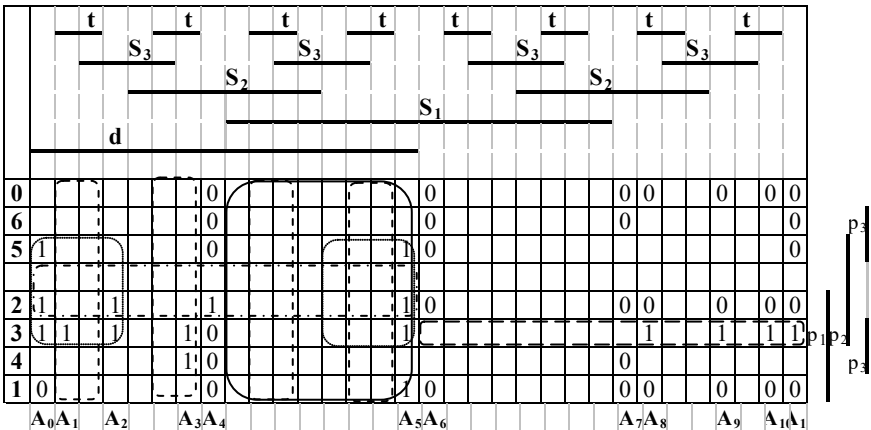
Карта Карно для переменной p_1



$$p_1 = d \cdot \overline{p_3} + d \cdot \overline{S_1} \cdot p_1 + \overline{d} \cdot p_1 \cdot p_2 + p_2 \cdot \overline{p_3}.$$

Таблица 9

Карта Карно для переменной p_2



$$p_2 = d \cdot S_1 + d \cdot t + d \cdot \overline{S_2} \cdot p_2 + d \cdot p_2 \cdot \overline{p_3} + \overline{d} \cdot p_1 \cdot p_2 \cdot p_3.$$

Таблица 10

Карта Карно для переменной p_3

	t		t		t		t		t		t		t											
	S ₃				S ₃				S ₃				S ₃											
	S ₂						S ₂						S ₂											
	S ₁												S ₁											
	d												d											
0															0	0	0	0	0					
6															0				0					
5	1														1				1					
2	0														0	0			0	0				
3	1	1	1	1											1				0	0				
4																			0	0				
1	0														0	0			0	0				
	A ₀ A ₁	A ₂	A ₃ A ₄		A ₅ A ₆				A ₇ A ₈				A ₉	A ₁₀ A ₁₁										

$$p_3 = d \cdot p_3 + d \cdot S_3 + \overline{p_1} \cdot p_2.$$

Построим карты Карно и запишем выражения для выходных переменных T и c (табл. 11, 12).

Таблица 11

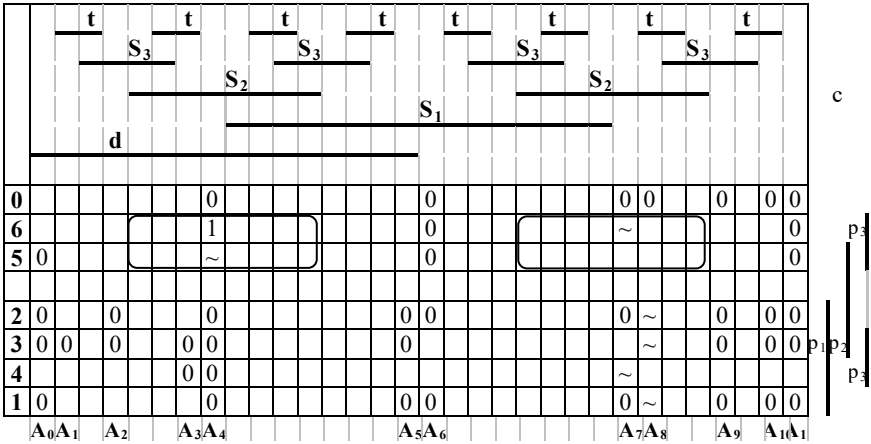
Карта Карно для переменной T

	t		t		t		t		t		t		t											
	S ₃				S ₃				S ₃				S ₃											
	S ₂						S ₂						S ₂											
	S ₁												S ₁											
	d												d											
0															0	0	0	0	0	0				
6															0				0	0				
5	0														0				0	0				
2	0		0												0	0			0	0				
3	0	0	0	1	~										0				0	0				
4				1	1														~	~				
1	0														0	0			0	0				
	A ₀ A ₁	A ₂	A ₃ A ₄		A ₅ A ₆				A ₇ A ₈				A ₉	A ₁₀ A ₁₁										

$$T = S_2 \cdot p_1 \cdot p_3.$$

Таблица 12

Карта Карно для переменной c

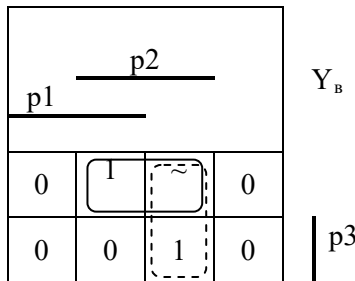


$$c = S2 \cdot \overline{p1} \cdot p3.$$

Для выходных переменных Y_B , Y_H строятся карты Карно, в которых указывается их зависимость только от внутренних переменных $p1, p2, p3$ (табл. 13, 14).

Таблица 13

Карта Карно для переменной Y_B



$$Y_B = p2 \cdot \overline{p3} + p2 \cdot \overline{p1}.$$

Карта Карно для переменной Y_n

$p2$ <hr style="width: 50%; margin: 0 auto;"/>				Y_n
$p1$ <hr style="width: 50%; margin: 0 auto;"/>				
1	0	~	0	$p3$
0	1	0	0	

$$Y_n = p1 \cdot \overline{p2} \cdot \overline{p3} + p1 \cdot p2 \cdot p3.$$

4.4.2. Определение функций возбуждения памяти при реализации памяти на RS-триггерах

Для реализации памяти на триггерах необходимо записать функцию возбуждения памяти для каждого входа триггера. Для этого необходимо выполнить следующие действия:

1) закодированную автоматную таблицу представить в виде карт Карно. При этом для каждой переменной кодирования построить две карты Карно для входа R и входа S триггера;

2) заполнить карты Карно для каждой переменной кодирования. Вместо номера состояния в клетку вписывается значение входного сигнала триггера, которое обеспечивает его переключение из исходного состояния, определяемого строкой, в последующее, определяемое номером в клетке;

3) по каждой заполненной карте Карно записать минимизированное логическое выражение функции возбуждения памяти для искомой переменной;

4) полученные выражения преобразовать в заданный базис;

5) построить комбинационную схему возбуждения памяти;

6) построить карты Карно для выходных переменных, записать выражения выходных функций в заданном базисе и построить комбинационную схему выходов.

Для заполнения карт Карно можно использовать таблицу переходов RS-триггера:

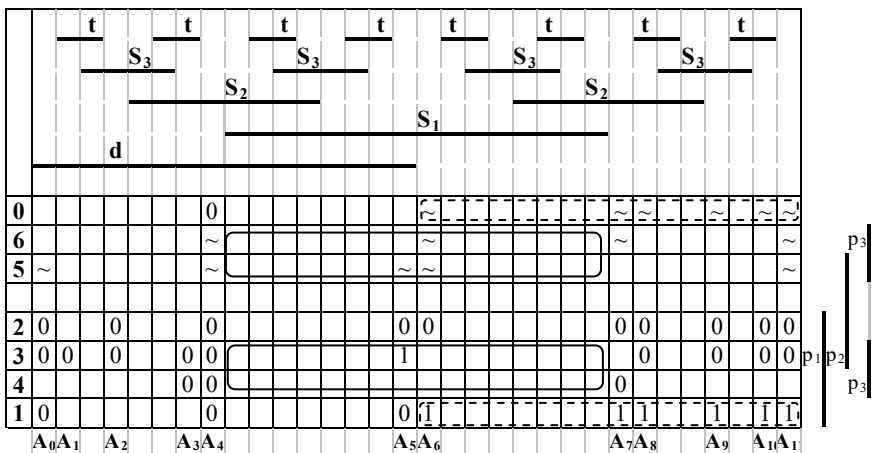
$X_{\text{исходное}}$	RS	$X_{\text{послед}}$
0	~ 0	0
0	0 1	1
1	1 0	0
1	0 \sim	1

Так как у нас для кодирования используются три переменные кодирования, то для каждой из них строим две карты Карно.

Строим карты Карно для входов триггеров $R1, S1, R2, S2, R3, S3$ (табл. 15–20).

Таблица 15

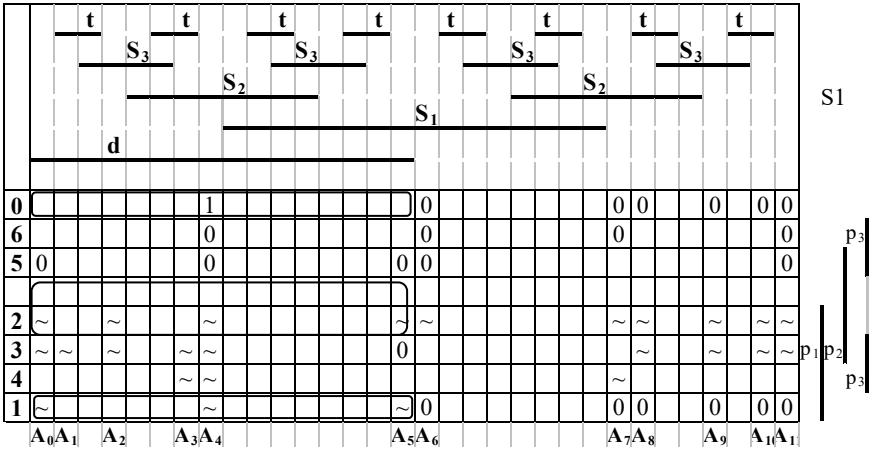
Карта Карно для переменной $R1$



$$R1 = S1 \cdot p3 + \bar{d} \cdot \bar{p2} \cdot \bar{p3}.$$

Таблица 16

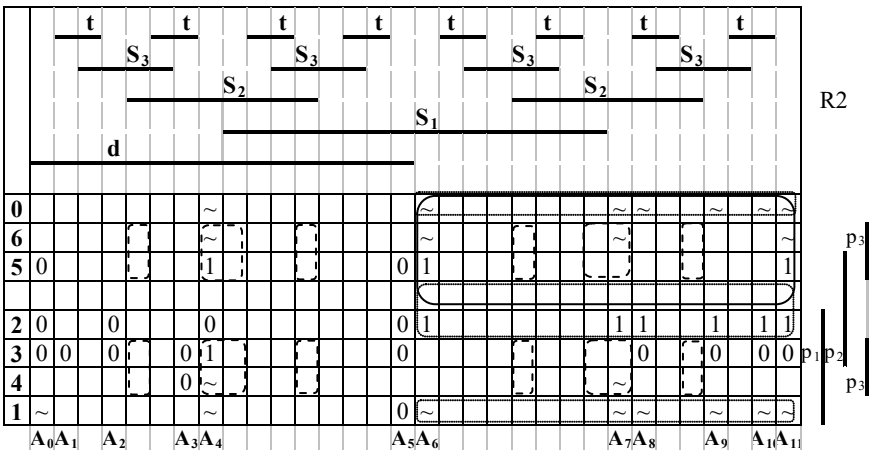
Карта Карно для переменной S1



$$S1 = d \cdot \overline{p3}.$$

Таблица 17

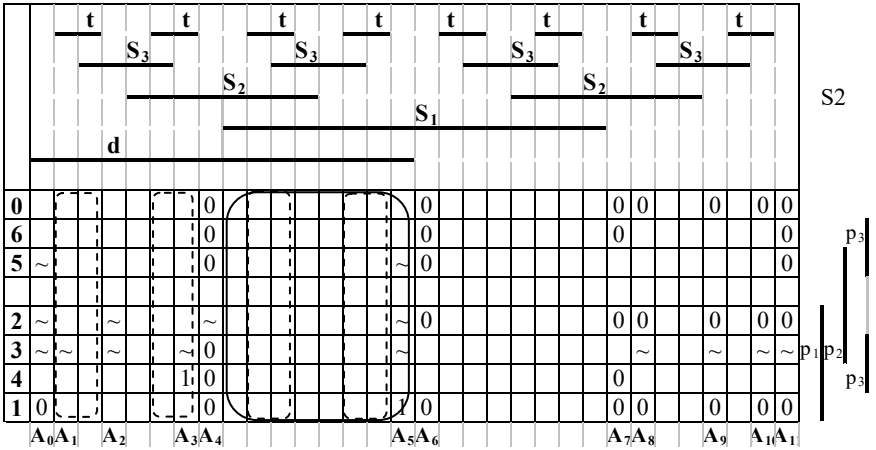
Карта Карно для переменной R2



$$R2 = \overline{d} \cdot \overline{p1} + \overline{d} \cdot \overline{p3} + S2 \cdot \overline{t} \cdot p3.$$

Таблица 18

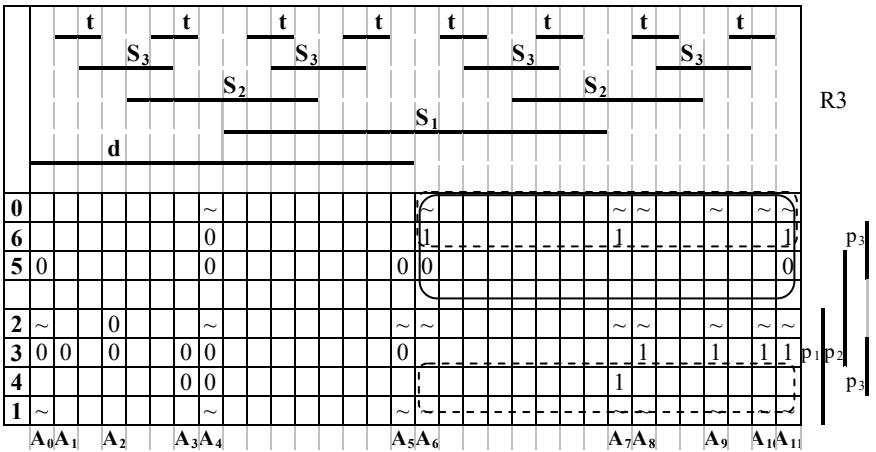
Карта Карно для переменной S2



$$S2 = d \cdot S1 + d \cdot t.$$

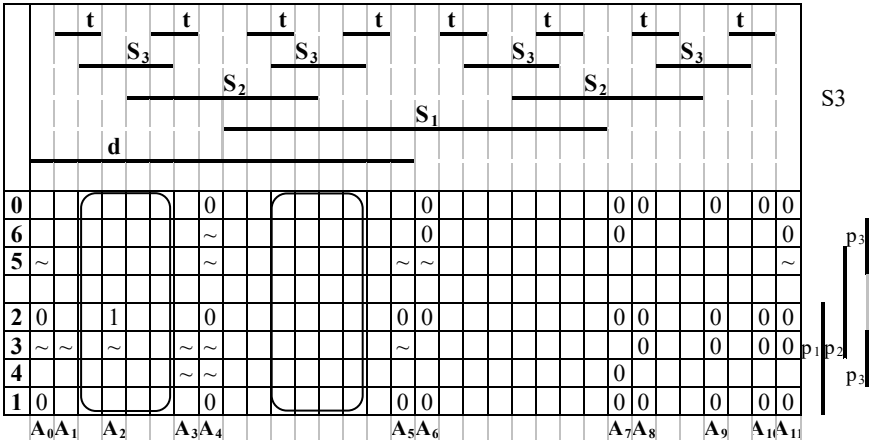
Таблица 19

Карта Карно для переменной R3



$$R3 = \bar{d} \cdot p1 + \bar{d} \cdot p2.$$

Карта Карно для переменной S3



$$S3 = d \cdot S3.$$

4.4.3. Приведение функций в заданный базис

Так как схему необходимо реализовать на элементах И-НЕ необходимо полученные выражения преобразовать в базис операции Шеффера [1, с. 91–93]. Операция Шеффера имеет вид

$$f = a | b = \overline{a \cdot b}.$$

Для преобразования выражения в базис операции Шеффера необходимо взять двойную инверсию от выражения, после чего раскрыть нижнюю инверсию, используя закон де-Моргана. В полученном выражении ввести обозначение операции Шеффера.

Если исходная функция имела общую инверсию, то эту инверсию можно реализовать, используя выражения:

$$a | a = \overline{a}; \quad a | 1 = \overline{a}.$$

Запишем полученные выражения для внутренних и выходных переменных в базисе операции Шеффера.

$$p1 = \overline{\overline{d \cdot p3 + d \cdot S1 \cdot p1 + \bar{d} \cdot p1 \cdot p2 + p2 \cdot p3}} = \overline{\overline{d \cdot p3 \cdot d \cdot S1 \cdot p1 \cdot \bar{d} \cdot p1 \cdot p2 \cdot p2 \cdot p3}} = \\ = (d | \bar{p3}) | (d | \bar{S1} | p1) | (\bar{d} | p1 | p2) | (p2 | \bar{p3});$$

$$p2 = \overline{\overline{d \cdot S1 + d \cdot t + d \cdot S2 \cdot p2 + d \cdot p2 \cdot p3 + \bar{d} \cdot p1 \cdot p2 \cdot p3}} = \\ = \overline{\overline{d \cdot S1 \cdot \bar{d} \cdot t \cdot d \cdot S2 \cdot p2 \cdot d \cdot p2 \cdot p3 \cdot \bar{d} \cdot p1 \cdot p2 \cdot p3}} = \\ = (d | S1) | (d | t) | (d | \bar{S2} | p2) | (d | p2 | \bar{p3}) | (\bar{d} | p1 | p2 | p3);$$

$$p3 = \overline{\overline{d \cdot p3 + d \cdot S3 + p1 \cdot p2}} = \overline{\overline{d \cdot p3 \cdot d \cdot S3 \cdot p1 \cdot p2}} = (d | p3) | (d | S3) | (\bar{p1} | p2);$$

$$R1 = \overline{\overline{S1 \cdot p3 + \bar{d} \cdot p2 \cdot p3}} = \overline{\overline{S1 \cdot p3 \cdot \bar{d} \cdot p2 \cdot p3}} = (S1 | p3) | (\bar{d} | \bar{p2} | \bar{p3});$$

$$S1 = \overline{\overline{d \cdot p3}} = \overline{\overline{d | p3}} = (d | \bar{p3}) | (d | \bar{p3});$$

$$R2 = \overline{\overline{\bar{d} \cdot p1 + \bar{d} \cdot p3 + S2 \cdot t \cdot p3}} = \overline{\overline{\bar{d} \cdot p1 \cdot \bar{d} \cdot p3 \cdot S2 \cdot t \cdot p3}} = \\ = (\bar{d} | \bar{p1}) | (\bar{d} | \bar{p3}) | (S2 | \bar{t} | p3);$$

$$S2 = \overline{\overline{d \cdot S1 + d \cdot t}} = \overline{\overline{d \cdot S1 \cdot d \cdot t}} = (d | S1) | (d | t);$$

$$R3 = \overline{\overline{\bar{d} \cdot p1 + \bar{d} \cdot p2}} = \overline{\overline{\bar{d} \cdot p1 \cdot \bar{d} \cdot p2}} = (\bar{d} | \bar{p1}) | (\bar{d} | \bar{p2});$$

$$S3 = \overline{\overline{d \cdot S3}} = \overline{\overline{d | S3}} = (d | S3) | (d | S3);$$

$$T = \overline{\overline{S2 \cdot p1 \cdot p3}} = \overline{\overline{S2 | p1 | p3}} = (S2 | p1 | p3) | (S2 | p1 | p3);$$

$$c = \overline{\overline{S2 \cdot p1 \cdot p3}} = \overline{\overline{S2 | \bar{p1} | p3}} = (S2 | \bar{p1} | p3) | (S2 | \bar{p1} | p3);$$

$$Y_b = \overline{\overline{p2 \cdot p3 + p2 \cdot p1}} = \overline{\overline{p2 \cdot p3 \cdot p2 \cdot p1}} = (p2 | \bar{p3}) | (p2 | \bar{p1});$$

$$Y_H = \overline{\overline{p1 \cdot p2 \cdot p3 + p1 \cdot p2 \cdot p2}} = \overline{\overline{p1 \cdot p2 \cdot p3 \cdot p1 \cdot p2 \cdot p2}} = \\ = (p1 | \bar{p2} | \bar{p3}) | (p1 | p2 | p3).$$

По полученным выражениям строим функциональную схему блока БЗ.

Функциональная схема блока БЗ на бесконтактных элементах И-НЕ с реализацией памяти петлями обратных связей представлена на рис. 6.

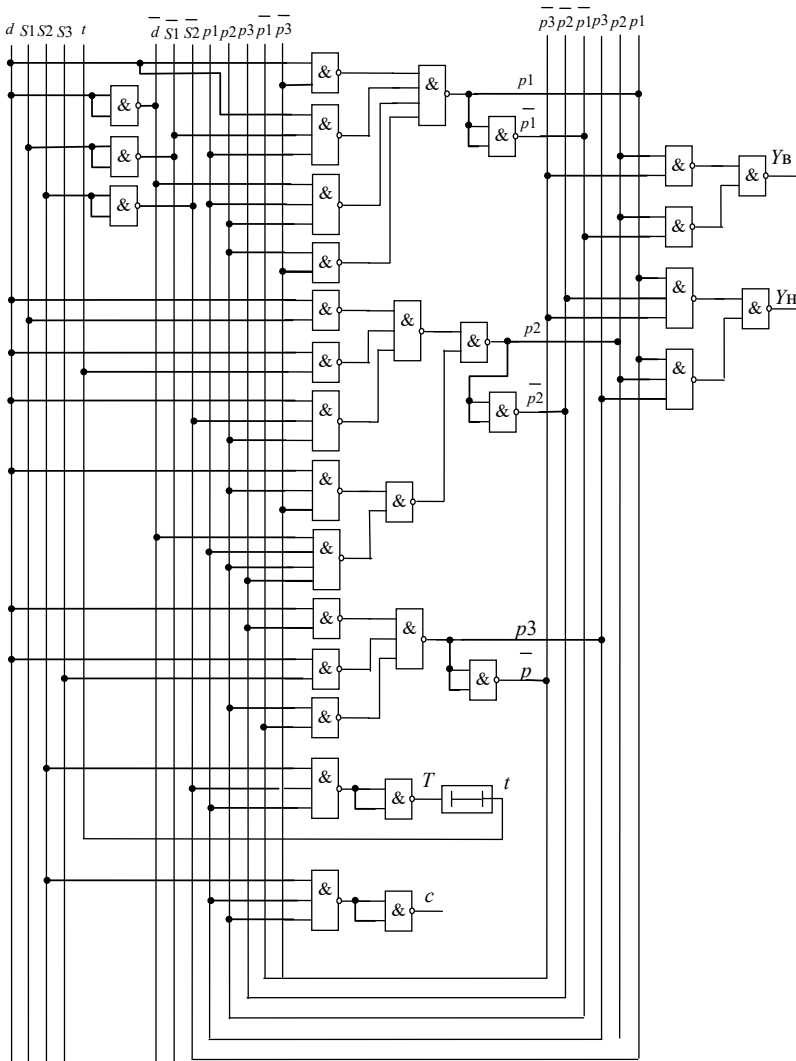


Рис. 6. Функциональная схема блока БЗ на элементах И-НЕ с реализацией памяти петлями обратных связей

Функциональная схема блока БЗ на бесконтактных элементах И-НЕ с реализацией памяти на RS-триггерах представлена на рис. 7.

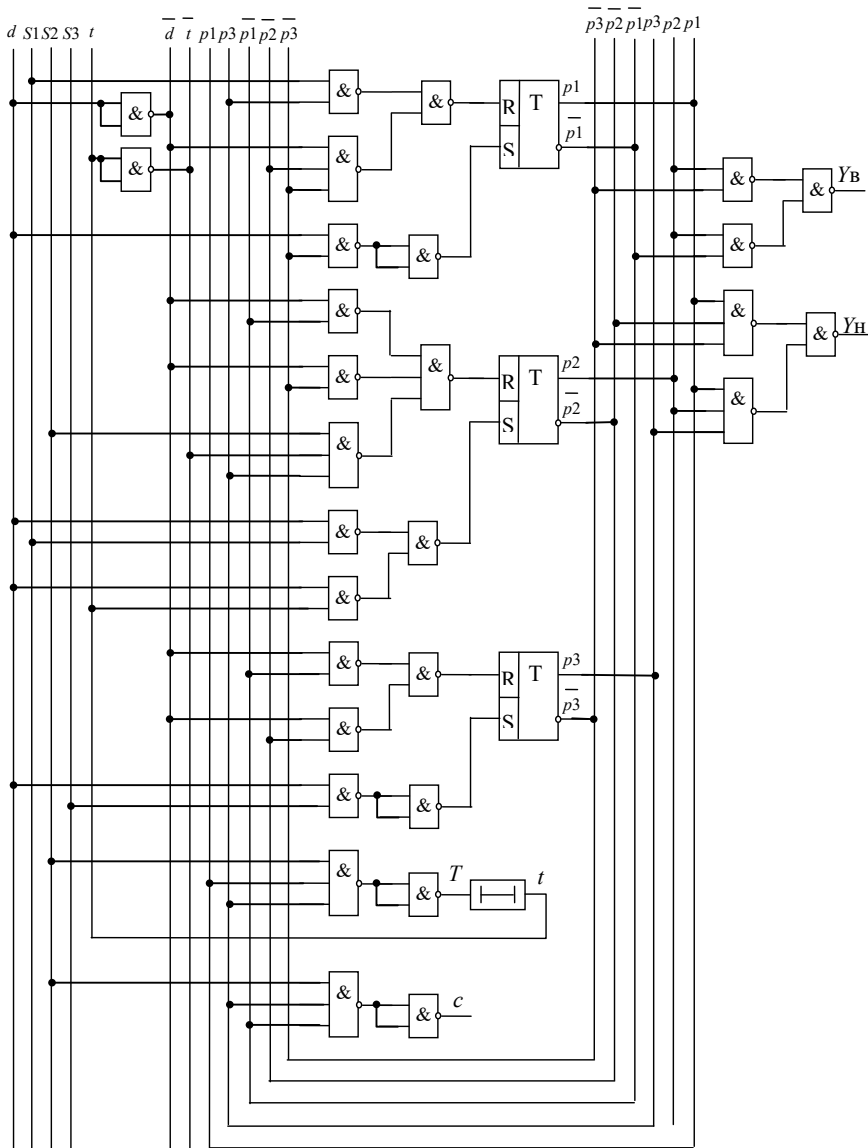


Рис. 7. Функциональная схема блока БЗ на элементах И-НЕ с реализацией памяти на RS-триггерах

4.5. Синтез блока, формирующего команду отработки автоматического цикла Б1

Входные и выходные сигналы блока представлены на рис. 8.

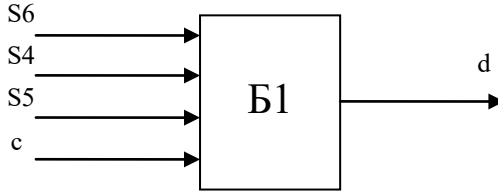


Рис. 8. Входные и выходные сигналы блока Б1

Для формального описания принципа работы блока составим автоматную таблицу. Так как последовательность поступления сигналов на вход блока неизвестна, следовательно, в автоматной таблице необходимо рассмотреть реакцию блока на любую допустимую входную последовательность. Количество состояний в автоматной таблице будет определяться количеством комбинаций значений выходного сигнала (2 комбинации: $d = 0$; $d = 1$). Автоматная таблица для блока Б1 будет иметь вид табл. 21.

Таблица 21

Автоматная таблица выходов и переходов блока Б1

		c		S ₅				c		S ₄				c		S ₅				d
		S ₆																		
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1

Данная таблица является минимизированной, т. к. содержит два состояния, которые несовместимы между собой по выходам. В качестве кодов состояний можно использовать значение выходной переменной d .

Для определения логического выражения для выходной переменной построим карту Карно (табл. 22).

Карта Карно для сигнала d

		\overline{c}				c									
		S_5				S_4				S_5					
		S_6													
0c	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
1c	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0

По карте Карно запишем минимизированное выражение для переменной d .

$$d = S_4 \cdot S_6 \cdot \overline{S_5} \cdot \overline{c} + d \cdot S_6 \cdot \overline{S_5} \cdot \overline{c}.$$

Преобразуем полученное выражение в базис операции Шеффера.

$$\begin{aligned} d &= \overline{\overline{S_4 \cdot S_6 \cdot \overline{S_5} \cdot \overline{c} + d \cdot S_6 \cdot \overline{S_5} \cdot \overline{c}}} = \overline{\overline{S_4 \cdot S_6 \cdot \overline{S_5} \cdot \overline{c}} \cdot \overline{d \cdot S_6 \cdot \overline{S_5} \cdot \overline{c}}} = \\ &= (S_4 | S_6 | \overline{S_5} | \overline{c}) | (d | S_6 | \overline{S_5} | \overline{c}). \end{aligned}$$

Функциональная схема блока Б1 представлена на рис. 9.

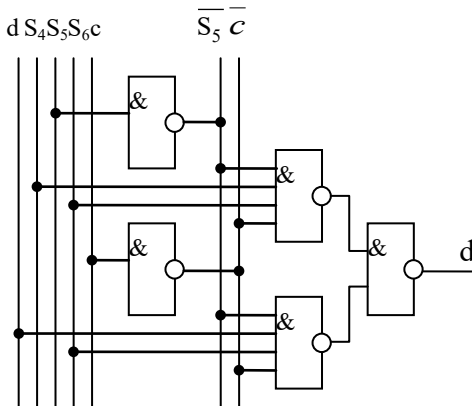


Рис. 9. Функциональная схема блока Б1 на элементах И-НЕ

4.6. Синтез блока режима «Наладка» Б2

Входные и выходные сигналы блока представлены на рис. 10.

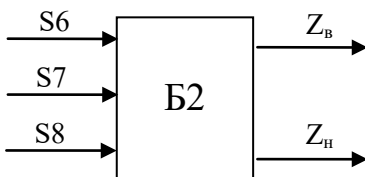


Рис. 10. Входные и выходные сигналы блока Б2

Автоматная таблица для данного блока составляется аналогично предыдущему блоку. Таблица будет содержать три внутренних состояния по числу допустимых комбинаций значений выходных сигналов ($Z_{в}, Z_{н} = 00; 01; 10$). Комбинация $Z_{в}, Z_{н} = 11$ является запрещенной. В автоматной таблице необходимо предусмотреть блокировку от одновременного нажатия кнопок $S7, S8$. Если $S7 = 1$ и $S8 = 1$, то на выходе должен формироваться сигнал $Z_{в} = 0, Z_{н} = 0$ для исключения короткого замыкания.

Автоматная таблица блока Б2 будет иметь вид табл. 23.

Таблица 23

Автоматная таблица выходов и переходов блока Б2

	S_6	S_8		S_7		S_8		$Z_{в}$	$Z_{н}$	
0	0	0	0	0	1	0	2	0	0	0
1	0	0	0	0	1	0	0	0	1	0
2	0	0	0	0	0	0	2	0	0	1

Данная таблица является минимизированной, т. к. содержит три состояния, которые несовместимы между собой по выходам. В качестве кодов состояний можно использовать значения выходных сигналов $Z_{в}, Z_{н}$.

Для определения выходных функций строим карты Карно для переменных $Z_{в}, Z_{н}$ (табл. 24, 25).

Таблица 24

Карта Карно для переменной Z_B

		S_8				S_8		
					S_7			
	S_6							
0c	0	0	0	0	$\overline{1}$	0	0	0
2c	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
1c	0	0	0	0	$\overline{1}$	0	0	0

Z_H |
| Z_B |

$$Z_B = S7 \cdot \overline{S6} \cdot \overline{S8} \cdot \overline{Z_H}.$$

Таблица 25

Карта Карно для переменной Z_H

		S_8				S_8		
					S_7			
	S_6							
0c	0	0	0	0	0	0	1	0
2c	0	0	0	0	0	0	1	0
	0	0	0	0	0	0	0	0
1c	0	0	0	0	0	0	0	0

| Z_H |
| Z_B |

$$Z_H = \overline{S6} \cdot \overline{S7} \cdot S8 \cdot \overline{Z_B}.$$

Преобразуем полученные выражения в базис Шеффера.

$$Z_B = \overline{\overline{S7 \cdot \overline{S6} \cdot \overline{S8} \cdot \overline{Z_H}}} = \overline{S7 | \overline{S6} | \overline{S8} | \overline{Z_H}} = (S7 | \overline{S6} | \overline{S8} | \overline{Z_H}) | (S7 | \overline{S6} | \overline{S8} | \overline{Z_H});$$

$$Z_H = \overline{\overline{\overline{S6 \cdot \overline{S7} \cdot S8 \cdot \overline{Z_B}}}} = \overline{\overline{S6 | \overline{S7} | S8 | \overline{Z_B}}} = (\overline{S6} | \overline{S7} | S8 | \overline{Z_B}) | (\overline{S6} | \overline{S7} | S8 | \overline{Z_B}).$$

На основании полученных выражений можем построить функциональную схему блока Б2 (рис. 11).

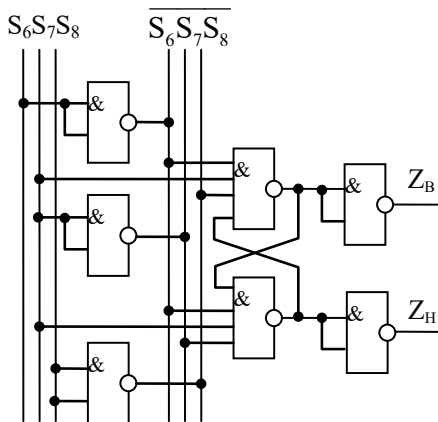


Рис. 11. Функциональная схема блока Б2 на элементах И-НЕ

4.7. Синтез выходного блока Б4

Входные и выходные сигналы блока представлены на рис. 12.

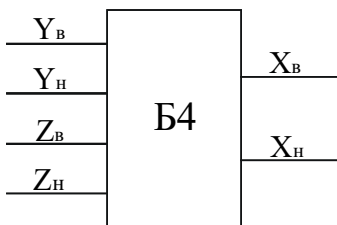


Рис. 12. Входные и выходные сигналы блока Б4

Автоматная таблица выходов и переходов для данного блока будет иметь вид табл. 26.

Таблица 26

Автоматная таблица выходов и переходов блока Б4

	$\overline{Y_H}$				Y_H				$\overline{Y_H}$				Y_H				X_B	X_H
	Z_B	Z_H	Y_B		Z_B	Z_H	Y_B		Z_B	Z_H	Y_B		Z_B	Z_H	Y_B			
0	1	~	~	0	~	~	~	~	1	0	~	~	2	~	2	0	0	0
1	1	~	~	0	~	~	~	~	1	0	~	~	0	~	0	0	1	0
2	0	~	~	0	~	~	~	~	0	0	~	~	2	~	2	0	0	1

На основании автоматной таблицы составим карты Карно и запишем выражения для выходных переменных X_B, X_H .

Таблица 27

Карта Карно для переменной X_B

	Y_H			Y_H				Y_H			Y_H				
	Z_H			Y_B				Z_H			Y_H				
Z_B															
0	1	~	~	0	~	~	~	1	0	~	~	0	~	0	0
2	0	~	~	0	~	~	~	0	0	~	~	0	~	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	~	~	0	~	~	~	1	0	~	~	0	~	0	0

X_H
 X_B

$$X_B = \overline{X_H} \cdot Z_B \cdot \overline{Z_H} + \overline{X_H} \cdot Y_B \cdot \overline{Y_H}.$$

Таблица 28

Карта Карно для переменной X_H

	Y_H			Y_H				Y_H			Y_H				
	Z_H			Y_B				Z_H			Y_H				
Z_B															
0	0	~	~	0	~	~	~	0	0	~	~	1	~	1	0
2	0	~	~	0	~	~	~	0	0	~	~	1	~	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	~	~	0	~	~	~	0	0	~	~	0	~	0	0

X_H
 X_B

$$X_H = \overline{X_B} \cdot Z_H \cdot \overline{Z_B} + \overline{X_B} \cdot Y_H \cdot \overline{Y_B}.$$

Преобразуем полученные выражения в базис Шеффера.

$$X_B = \overline{\overline{\overline{X_H} \cdot Z_B \cdot \overline{Z_H} + \overline{X_H} \cdot Y_B \cdot \overline{Y_H}}} = (\overline{X_H} | Z_B | \overline{Z_H}) | (\overline{X_H} | Y_B | \overline{Y_H});$$

$$X_H = \overline{\overline{\overline{X_B} \cdot Z_H \cdot \overline{Z_B} + \overline{X_B} \cdot Y_H \cdot \overline{Y_B}}} = (\overline{X_B} | Z_H | \overline{Z_B}) | (\overline{X_B} | Y_H | \overline{Y_B}).$$

На основании полученных выражений можем построить функциональную схему блока Б4 (рис. 13).

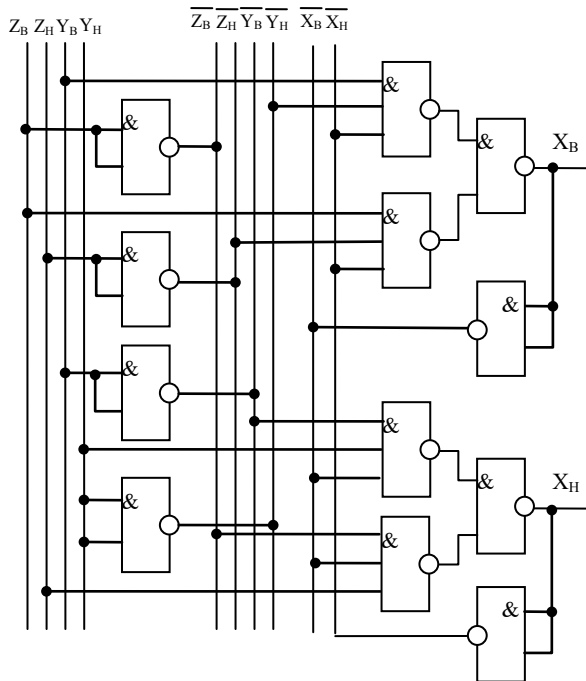


Рис. 13. Функциональная схема блока Б4 на элементах И-НЕ

5. ВЫБОР ЭЛЕМЕНТОВ. ПОСТРОЕНИЕ ПРИНЦИПАЛЬНОЙ СХЕМЫ УСТРОЙСТВА УПРАВЛЕНИЯ

Принципиальная схема устройства управления может быть реализована на любых логических элементах. В нашем случае схема должна быть реализована на элементах И-НЕ.

Логическую операцию И-НЕ реализуют интегральные микросхемы типа ЛА [2]. Каждый из корпусов интегральных схем (ИС) типа ЛА содержит от двух до четырех логических элементов, а микросхемы ЛА2 и ЛА19 содержат по одному логическому элементу И-НЕ на восемь и на двенадцать входов соответственно.

Условные обозначения и цоколевка микросхем типа ЛА представлены на рис. 14

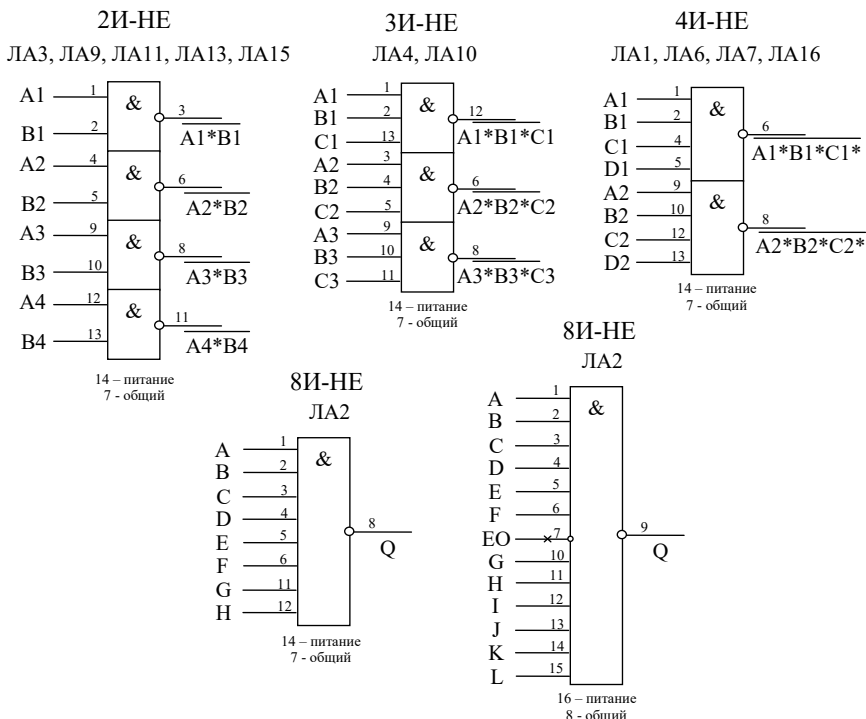


Рис. 14. Условные обозначения и цоколевка микросхем типа ЛА

ЕО – дополнительный вход, дающий разрешение по выходу (ЕО = 1 обеспечивает 3 состояние Z с очень большим выходным сопротивлением).

Микросхемы ЛА7...ЛА11, ЛА13, ЛА18 имеют выходы с открытым коллектором. Для формирования выходного перепада напряжения к выходу такого элемента необходимо подключить нагрузку. В качестве нагрузки могут выступать сегменты индикаторов, лампы накаливания, светодиоды, обмотки реле.

Для построения бесконтактных схем используем микросхемы 155 серии. Также будет использован асинхронный RS-триггер. Вы-

ходные сигналы $X_{в}$, $X_{н}$ будем формировать с помощью микросхем с открытым коллектором для подключения обмоток промежуточных реле, которые будут включать силовые контакторы.

Для первой схемы (без применения RS-триггеров) будем использовать следующие микросхемы:

K155ЛА3 – DD1 – DD9;

K155ЛА9 – DD10;

K155ЛА4 – DD11 – DD15;

K155ЛА1 – DD16 – DD18.

Для второй схемы (с применением RS-триггеров) будем использовать следующие микросхемы

K155ЛА3 – DD1 – DD9;

K155ЛА9 – DD10;

K155ЛА4 – DD11 – DD14;

K155ЛА1 – DD15, DD16

K155ТР2 – DD17.

Принципиальные схемы устройства управления представлены в прил. 1, 2.

6. ПРОВЕРКА ПРАВИЛЬНОСТИ РАБОТЫ УСТРОЙСТВА УПРАВЛЕНИЯ

Для проверки правильности работы устройства управления необходимо построить проверочную таблицу, в которой отмечают реакцию управляющего устройства на входную последовательность сигналов. Для этого по значениям входных и внутренних сигналов в данный момент времени по полученным выражениям вычисляют значения выходных сигналов и внутренних сигналов в следующий момент времени. По этим сигналам проверяется, выполняется ли заданный цикл перемещений.

Проверочная таблица строится для блока БЗ, т. к. данный блок имеет наиболее сложный алгоритм работы.

При заполнении таблицы необходимо придерживаться следующей последовательности. Задаемся первым входным набором (нахождение каретки в исходном положении) и начальным значением внутренних сигналов (код исходного внутреннего состояния). На

основании значений данных сигналов вычисляем значения выходных сигналов и последующих внутренних сигналов. Если значения последующих внутренних сигналов отличаются от значений предыдущих внутренних сигналов (переход из состояния в состояние), то на следующем шаге для вычислений берем тот же входной набор и полученные на предыдущем шаге значения внутренних сигналов. Если значения последующих внутренних сигналов не отличаются от значений предыдущих внутренних сигналов (автомат находится в устойчивом состоянии), то на следующем шаге для вычислений берем следующий входной набор и полученные на предыдущем шаге значения внутренних сигналов. Таким образом, в проверочной таблице должны отражаться все переходы, отмеченные в исправленной закодированной минимизированной автоматной таблице.

Для построения проверочной таблицы может использоваться редактор таблиц Excel.

Для вычисления выходных сигналов RS-триггера может использоваться следующее выражение

$$X(t+1) = S + \bar{R} \cdot X(t),$$

где S , R – значения входных сигналов триггера в данный момент времени;

$X(t)$ – значение выходного сигнала в данный момент времени;

$X(t+1)$ – значение выходного сигнала в последующий момент времени.

Проверочные таблицы для первого и второго вариантов реализации будут иметь вид табл. 29, 30.

Из проверочных таблиц видно, что при поступлении входных последовательностей в соответствии с циклом перемещений на выходе устройства управления формируются заданные значения выходных сигналов. Следовательно, проектирование схемы управления выполнено верно, и устройство управления работает по заданному алгоритму.

Таблица 29

Проверочная таблица устройства управления
для первого варианта реализации

№ п/п	Входные сигналы					Предыдущие внутренние			Последующие внутренние			Выходные сигналы			
	<i>d</i>	<i>S1</i>	<i>S2</i>	<i>S3</i>	<i>t</i>	<i>p1</i>	<i>p2</i>	<i>p3</i>	<i>p1</i>	<i>p2</i>	<i>p3</i>	<i>Y_в</i>	<i>Y_п</i>	<i>T</i>	<i>c</i>
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
2	1	0	1	0	0	0	0	0	1	0	0	0	1	0	0
3	1	0	1	0	0	1	0	0	1	0	0	0	1	0	0
4	1	0	0	0	0	1	0	0	1	0	0	0	1	0	0
5	1	1	0	0	0	1	0	0	1	1	0	1	0	0	0
6	1	1	0	0	0	1	1	0	1	1	0	1	0	0	0
7	1	0	0	0	0	1	1	0	1	1	0	1	0	0	0
8	1	0	1	0	0	1	1	0	1	1	0	1	0	0	0
9	1	0	0	0	0	1	1	0	1	1	0	1	0	0	0
10	1	0	0	1	0	1	1	0	1	1	1	0	1	0	0
11	1	0	0	1	0	1	1	1	1	1	1	0	1	0	0
12	1	0	0	0	0	1	1	1	1	1	1	0	1	0	0
13	1	0	1	0	0	1	1	1	1	0	1	0	0	1	0
14	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0
15	1	0	1	0	1	1	0	1	1	1	1	0	1	1	0
16	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0
17	1	0	0	0	1	1	1	1	1	1	1	0	1	0	0
18	1	0	0	0	0	1	1	1	1	1	1	0	1	0	0
19	1	1	0	0	0	1	1	1	0	1	1	1	0	0	0
20	1	1	0	0	0	0	1	1	0	1	1	1	0	0	0
21	1	0	0	0	0	0	1	1	0	1	1	1	0	0	0
22	1	0	1	0	0	0	1	1	0	0	1	0	0	0	1
23	1	0	1	0	0	0	0	1	0	0	1	0	0	0	1
24	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
25	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

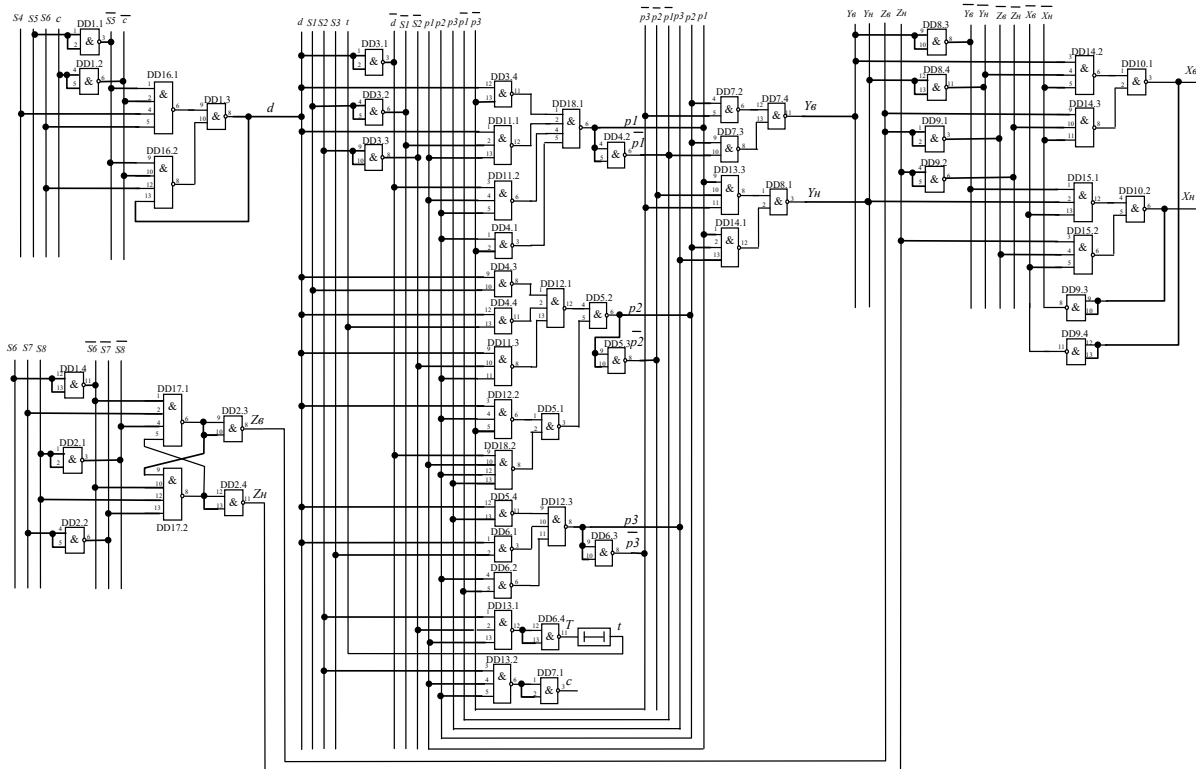
Проверочная таблица устройства управления
для второго варианта реализации

№ п/п	Входные сигналы					Предыдущие внутренние			Входные сигналы триггеров						Последующие внутренние			Выходные сигналы			
	<i>d</i>	<i>S1</i>	<i>S2</i>	<i>S3</i>	<i>t</i>	<i>p1</i>	<i>p2</i>	<i>p3</i>	<i>R1</i>	<i>S1</i>	<i>R2</i>	<i>S2</i>	<i>R3</i>	<i>S3</i>	<i>p1</i>	<i>p2</i>	<i>p3</i>	<i>Y_в</i>	<i>Y_н</i>	<i>T</i>	<i>c</i>
1	0	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
2	1	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0
3	1	0	1	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0
4	1	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0
5	1	1	0	0	0	1	0	0	0	1	0	1	0	0	1	1	0	1	0	0	0
6	1	1	0	0	0	1	1	0	0	1	0	1	0	0	1	1	0	1	0	0	0
7	1	0	0	0	0	1	1	0	0	1	0	0	0	0	1	1	0	1	0	0	0
8	1	0	1	0	0	1	1	0	0	1	0	0	0	0	1	1	0	1	0	0	0
9	1	0	0	0	0	1	1	0	0	1	0	0	0	0	1	1	0	1	0	0	0
10	1	0	0	1	0	1	1	0	0	1	0	0	0	1	1	1	1	0	1	0	0
11	1	0	0	1	0	1	1	1	0	0	0	0	0	1	1	1	1	0	1	0	0
12	1	0	0	0	0	1	1	1	0	0	0	0	0	0	1	1	1	0	1	0	0
13	1	0	1	0	0	1	1	1	0	0	1	0	0	0	1	0	1	0	0	1	0
14	1	0	1	0	0	1	0	1	0	0	1	0	0	0	1	0	1	0	0	1	0
15	1	0	1	0	1	1	0	1	0	0	0	1	0	0	1	1	1	0	1	1	0
16	1	0	1	0	1	1	1	1	0	0	0	1	0	0	1	1	1	0	1	1	0
17	1	0	0	0	1	1	1	1	0	0	0	1	0	0	1	1	1	0	1	0	0
18	1	0	0	0	0	1	1	1	0	0	0	0	0	0	1	1	1	0	1	0	0
19	1	1	0	0	0	1	1	1	1	0	0	1	0	0	0	1	1	1	0	0	0
20	1	1	0	0	0	0	1	1	1	0	0	1	0	0	0	1	1	1	0	0	0
21	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	1	1	0	0	0
22	1	0	1	0	0	0	1	1	0	0	1	0	0	0	0	0	1	0	0	0	1
23	1	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	1
24	0	0	1	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0
25	0	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0

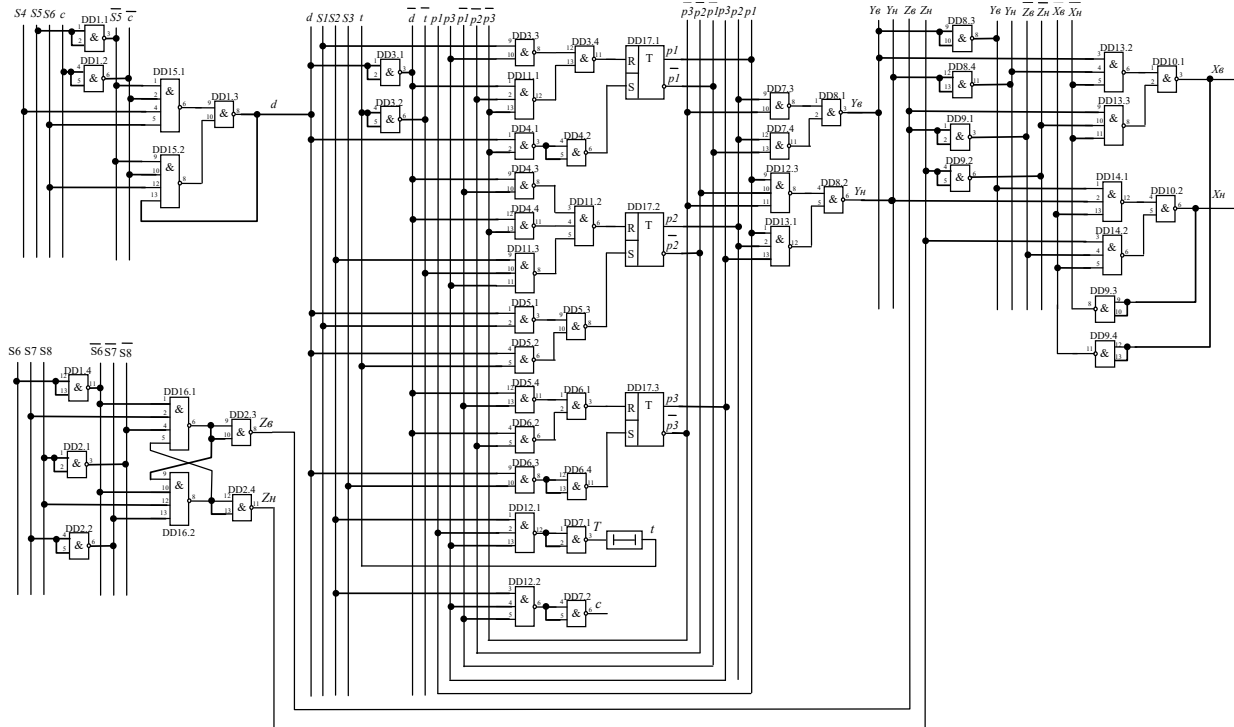
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Ершова, Э.Б. Основы дискретной автоматики в электросвязи / Э.Б. Ершова, В.Н. Рогинский, Н.П. Маркин. – М.: Связь, 1980.
2. Цифровые интегральные микросхемы: справочник / М.И. Богданович [и др.]. – 2-е изд., перераб. и доп. – Минск: Беларусь, Польша, 1996.

Принципиальная схема устройства управления при реализации только на бесконтактных элементах И-НЕ



Принципиальная схема устройства управления при реализации на бесконтактных элементах И-НЕ и RS-триггерах



ОГЛАВЛЕНИЕ

1. ЗАДАНИЕ К КУРСОВОЙ РАБОТЕ. ПОРЯДОК ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ.	3
2. ЗАПИСЬ УСЛОВИЙ РАБОТЫ УСТРОЙСТВА УПРАВЛЕНИЯ.	6
3. БЛОЧНЫЙ СИНТЕЗ.	8
4. СИНТЕЗ ОТДЕЛЬНЫХ БЛОКОВ.	11
4.1. Построение автоматной таблицы выходов и переходов блока БЗ.	11
4.2. Минимизация памяти автомата.	14
4.3. Кодирование внутренних состояний автомата.	19
4.4. Построение функциональной схемы устройства управления.	26
4.5. Синтез блока, формирующего команду отработки автоматического цикла Б1.	39
4.6. Синтез блока режима «Наладка» Б2.	41
4.7. Синтез выходного блока Б4.	43
5. ВЫБОР ЭЛЕМЕНТОВ. ПОСТРОЕНИЕ ПРИНЦИПИАЛЬНОЙ СХЕМЫ УСТРОЙСТВА УПРАВЛЕНИЯ.	45
6. ПРОВЕРКА ПРАВИЛЬНОСТИ РАБОТЫ УСТРОЙСТВА УПРАВЛЕНИЯ.	47
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.	51
ПРИЛОЖЕНИЕ 1.	52
ПРИЛОЖЕНИЕ 2.	53

Учебное издание

МИГДАЛЁНОК Александр Анатольевич

ПРОЕКТИРОВАНИЕ ДИСКРЕТНЫХ
СИСТЕМ УПРАВЛЕНИЯ

Методическое пособие
по выполнению курсовой работы
для студентов специальности
1-53 01 05 «Автоматизированные электроприводы»

Редактор Е.О. Коржуева
Компьютерная верстка Н.А. Школьниковой

Подписано в печать 21.04.2011.

Формат 60×84¹/₁₆. Бумага офсетная.

Отпечатано на ризографе. Гарнитура Таймс.

Усл. печ. л. 3,2. Уч.-изд. л. 2,5. Тираж 100. Заказ 1182.

Издатель и полиграфическое исполнение:

Белорусский национальный технический университет.

ЛИ № 02330/0494349 от 16.03.2009.

Проспект Независимости, 65. 220013, Минск.