

Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Оренбургский государственный университет»

Кафедра вычислительной техники и защиты информации

Е.В. Бурькова

СХЕМОТЕХНИКА

Рекомендовано к изданию Редакционно-издательским советом
федерального государственного бюджетного образовательного
учреждения высшего образования «Оренбургский
государственный университет» в качестве методических
указаний для студентов, обучающихся по программе
высшего образования по направлению подготовки
10.03.01 Информационная безопасность

Оренбург
2016

УДК 621.38(076.5)
ББК 32.85 я 73
Б 91

Рецензент – кандидат технических наук, доцент В.В. Извозчикова

Бурькова Е.В.
Б 91 Схемотехника: методические указания к лабораторным работам / Е.В. Бурькова; – Оренбургский гос. ун-т. – Оренбург: ОГУ, 2016. – 68 с.

В методических указаниях представлены теоретические сведения о цифровых электронных устройствах, принципах их действия, примерах их применения в сфере систем защиты объектов информатизации. Методические указания содержат материалы для проведения лабораторных работ по курсу «Электроника и схемотехника», приведены примеры структурных и функциональных схем, даны задания, вопросы для самопроверки.

Методические указания предназначены для студентов направления подготовки 10.03.01 Информационная безопасность.

УДК 621.38(076.5)
ББК 32.85 я 73

© Бурькова Е.В., 2016
© ОГУ, 2016

Содержание

	Введение.....	5
1	Лабораторная работа № 1. Синтез комбинационных схем с одним выходом	6
1.1	Цель работы.....	6
1.2	Теоретические сведения.....	6
1.3	Задание.....	13
1.4	Контрольные вопросы.....	15
2	Лабораторная работа № 2. Реализация логических функций на мультиплексорах.	16
2.1	Цель работы.....	16
2.2	Теоретические сведения.....	16
2.3	Задание.....	19
2.4	Контрольные вопросы.....	25
3	Лабораторная работа № 3. Синтез комбинационных схем с множеством выходов. Реализация схемы дешифратора.	26
3.1	Цель работы.....	26
3.2	Теоретические сведения.....	26
3.3	Задание.....	31
3.4	Контрольные вопросы.....	34
4	Лабораторная работа № 4. Цифровые компараторы.....	35
4.1	Цель работы.....	35
4.2	Теоретические сведения.....	35
4.3	Задание.....	37
4.4	Контрольные вопросы.....	39
5	Лабораторная работа № 5. Синтез сумматоров.....	40
5.1	Цель работы.....	40
5.2	Теоретические сведения.....	40
5.3	Задание.....	44

5.4	Контрольные вопросы.....	45
6	Лабораторная работа № 6. Триггеры и их преобразования	46
6.1	Цель работы.....	46
6.2	Теоретические сведения.....	46
6.3	Задание.....	50
6.4	Контрольные вопросы.....	51
7	Лабораторная работа № 7. Синтез регистров.	52
7.1	Цель работы.....	52
7.2	Теоретические сведения.....	52
7.3	Задание.....	58
7.4	Контрольные вопросы.....	59
8	Лабораторная работа № 7. Синтез счетчиков.....	60
8.1	Цель работы.....	60
8.2	Теоретические сведения.....	60
8.3	Задание.....	66
8.4	Контрольные вопросы.....	67
	Список использованных источников.....	68

Введение

Схемотехника - это научно-техническое направление, охватывающее проблемы проектирования и исследования схем электронных устройств радиотехники и связи, вычислительной техники, автоматики и других областей техники. Цель данного курса заключается в решении задач схемотехнической реализации типовых элементов, узлов, блоков цифровой вычислительной техники.

Основной задачей схемотехники является синтез (определение структуры) электронных схем, обеспечивающих выполнение определённых функций, и расчёт параметров входящих в них элементов. Теоретической базой схемотехники служат теория линейных и нелинейных электрических цепей, электродинамика, математическое программирование, теория автоматов. Достоинства цифровых электронных устройств:

- высокая помехоустойчивость,
- высокая надёжность,
- возможность длительного хранения информации,
- совместимость с интегральной технологией

Цифровые электронные устройства входят в состав систем обеспечения безопасности различных объектов информатизации. Изучение основ схемотехники является важным для понимания принципов работы таких систем, помогает формировать профессиональные компетенции бакалавров в области информационной безопасности.

Данные методические указания предназначены для проведения лабораторных работ по курсу «Электроника и схемотехника» для студентов направления подготовки 10.03.01 Информационная безопасность.

Методические указания содержат восемь разделов, в которых даны материалы для проведения лабораторных работ по курсу «Электроника и схемотехника», приведены примеры структурных и функциональных схем, даны задания, вопросы для самопроверки.

1 Лабораторная работа № 1. Синтез комбинационных схем с одним выходом

1.1 Цель работы

Изучить способы построения комбинационных схем с одним выходом в базисах элементов: И, ИЛИ, НЕ; И-НЕ; ИЛИ-НЕ.

1.2 Теоретические сведения

Рассматривается задача реализации переключательной функции (ПФ) $y = f(x_1, x_2, \dots, x_n)$, то есть построения схемы, имеющей n -входов x_1, x_2, \dots, x_n и выход y , функционирующей в соответствии с функцией f . При подаче на входы комбинации значений x_1, x_2, \dots, x_n , соответствующей входному набору ПФ, схема формирует на выходе значение сигнала y , соответствующее значению ПФ на этом наборе. Таким образом, рассматриваемые схемы, которые называют логическими схемами (ЛС) или комбинационными схемами (КС) с одним выходом, реализуют ПФ.

КС строят из логических элементов (ЛЭ), каждый из которых реализует элементарную ПФ – логическую операцию. Набор типов ЛЭ обычно ограничивают. Разрешенный набор типов ЛЭ называют базисом, а соответствующие типы – базовыми. Задача реализации ПФ является решаемой, если базис является функционально полным. На занятии рассматриваются способы построения КС в базисах элементов:

- И, ИЛИ, НЕ;
- И-НЕ;
- ИЛИ-НЕ.

Если задача синтеза имеет решение, то оно, как правило, не единственное. Эти решения можно сравнивать между собой и устанавливать между ними отношения предпочтения. Качество решения задачи синтеза КС оценивают по затратам оборудования и быстродействию. Каждый из этих критериев качества

может иметь более высокий приоритет. Тогда задача реализации ПФ может быть сформулирована в следующих вариантах:

- построить КС с минимальными затратами оборудования, удовлетворяющую заданным требованиям к быстродействию;
- построить КС с максимальным быстродействием при ограниченных затратах оборудования.

Если не определена элементная база (типы интегральных схем), то для оценки затрат оборудования используют следующие показатели:

- критерий Квайна – суммарное число входов всех ЛЭ;
- число ЛЭ в схеме.

Если задана элементная база, то затраты оборудования оцениваются числом корпусов интегральных схем или требуемой площадью кристалла БИС. При этом задача синтеза КС решается в условиях ограничений по:

- допустимому числу входов ЛЭ;
- допустимой нагрузочной способности ЛЭ (допустимому числу входов ЛЭ, которые можно подключить к выходу данного ЛЭ).

Быстродействие КС оценивается задержкой распространения сигнала $t_{зп}$, то есть временным интервалом, на который задержаны изменения значений выходного сигнала относительно вызвавших их изменений одного из входных сигналов. В общем случае значения задержек по разным входам могут быть разными, поэтому быстродействие оценивается по наихудшему случаю

$$t_{зп} = \max t_{зп i}, \quad i=1, n \quad (1.1)$$

где $t_{зп i}$ - задержка распространения по i -му входу, равная сумме задержек ЛЭ, лежащих на самом длинном пути от i -го входа до выхода схемы. Чтобы упростить определение быстродействия КС обычно считают, что все ЛЭ имеют одинаковые задержки $t_{лэ}$. Тогда задержки можно определять в относительных единицах $t_{зп} / t_{лэ}$.

Для снижения затрат оборудования при реализации ПФ используют их минимальные формы записи. Решение задачи синтеза КС сводится к выполнению следующих шагов:

- минимизация ПФ (построение МДНФ и МКНФ);
- преобразование нормальных форм к виду, удобному для реализации, то есть к суперпозиции функций из элементного базиса;
- построение схемы по формуле ПФ, то есть графа вычислений ПФ, вершинами которого являются соответствующие ЛЭ.

Пример 1 - Синтезировать КС, реализующую ПФ, которая задана картой Карно (рисунок 1.1). минимизацию можно реализовать как по «1», так и по «0».

		x1 x2			
x3x4		00	01	11	10
00	$\overline{1}$	1	$\overline{1}$	1	
01	1	1	1	1	
11	0	0	0	0	
10	$\overline{\overline{1}}$	0	$\overline{\overline{1}}$	0	

Рисунок 1.1 – Карта Карно для примера 1

В результате минимизации получим минимальную дизъюнктивную нормальную функцию (МДНФ) для ПФ и ее инверсии

$$y = \overline{x1} \cdot \overline{x2} \cdot \overline{x4} + x1 \cdot x2 \cdot \overline{x4} + \overline{x3}, \quad (1.2)$$

$$\overline{y} = x3 \cdot x4 + \overline{x1} \cdot x2 \cdot x3 + x1 \cdot \overline{x2} \cdot x3, \quad (1.3)$$

а также минимальную конъюнктивную нормальную функцию (МКНФ) для ПФ и ее инверсии

$$y = (\bar{x}_3 + \bar{x}_4) \cdot (x_1 + \bar{x}_2 + \bar{x}_3) \cdot (\bar{x}_1 + x_2 + \bar{x}_3), \quad (1.4)$$

$$\bar{y} = (x_1 + x_2 + x_4) \cdot (\bar{x}_1 + \bar{x}_2 + x_4) \cdot x_3, \quad (1.5)$$

Любая из этих форм ПФ может быть использована для построения КС в базисе Буля (элементов И, ИЛИ, НЕ). Из сравнения этих форм следует, что минимальное число ЛЭ требуется для реализации МКНФ инверсии ПФ. Результат синтеза схемы для МКНФ приведен на рисунке 1.2. Минимальная задержка будет у КС, реализующих прямые формы ПФ, так как для их реализации не требуется дополнительный инвертор для инвертирования функции. На рисунке 1.3 приведена схема синтеза для МДНФ функции.

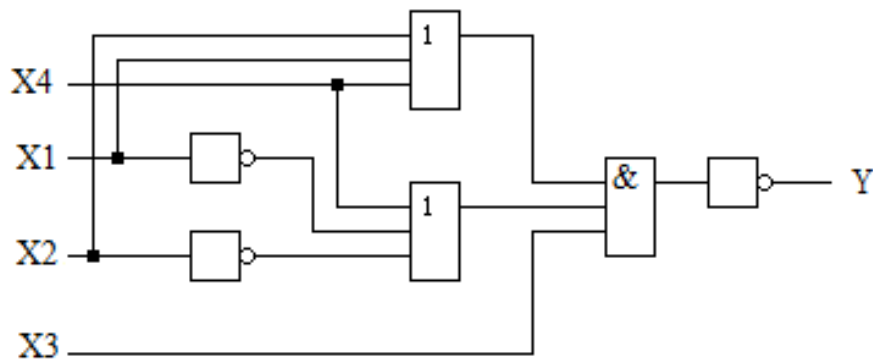


Рисунок 1.2 - Схема синтеза для МКНФ функции

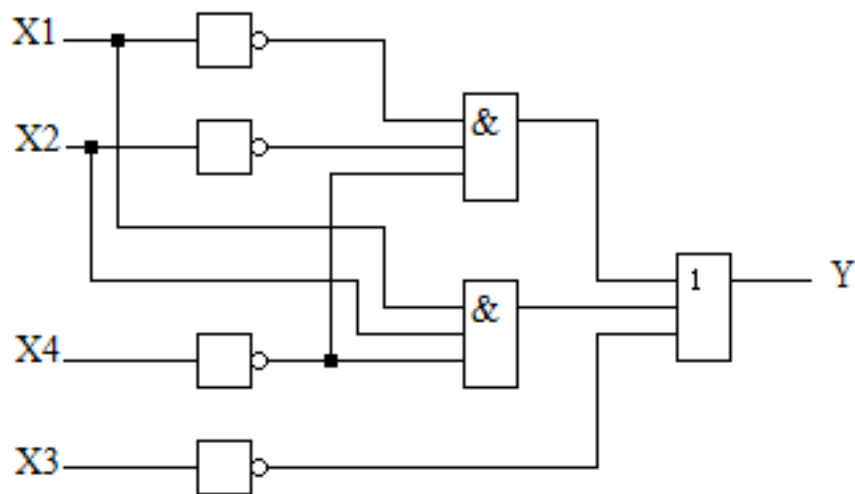


Рисунок 1.3 - Схема синтеза для МДНФ функции

Для реализации ПФ в базисе И-НЕ используют МДНФ. Выполняют ее двойную инверсии и преобразование Де-Моргана

$$y = \overline{\overline{x1 \cdot x2 \cdot x4} \cdot \overline{x1 \cdot x2 \cdot x4} \cdot x3}, \quad (1.6)$$

В дальнейшем будем считать инвертор эквивалентным одноходовым элементом И-НЕ, ИЛИ-НЕ. Тогда КС в базисе И-НЕ будет содержать три инвертора, три элемента 3И-НЕ. Схема представлена на рисунке 1.4.

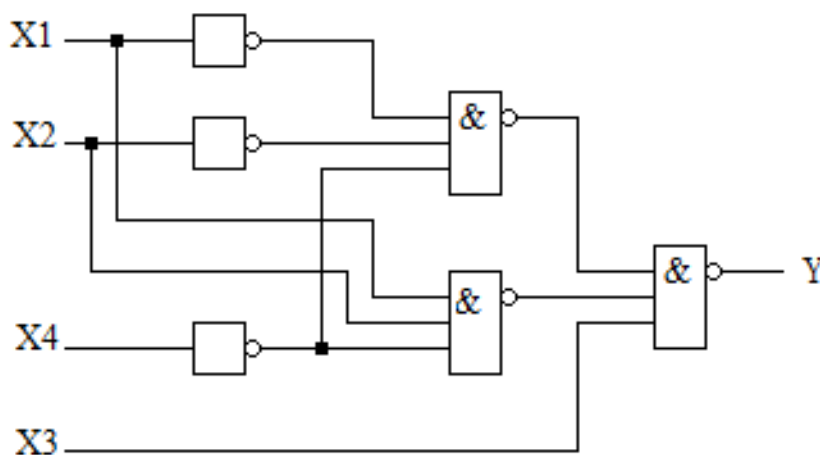


Рисунок 1.4 - Схема в базисе И-НЕ

Для реализации ПФ в базисе ИЛИ-НЕ используют МКНФ. Выполняют ее двойную инверсии и преобразование Де-Моргана. В данном примере более простая реализация будет для преобразованной МДНФ инверсии ПФ

$$\bar{y} = \overline{x1 + x2 + x4} + \overline{\bar{x1} + \bar{x2} + x4} + \bar{x3}, \quad (1.7)$$

КС в базисе ИЛИ-НЕ будет содержать три инвертора, три элемента 3ИЛИ-НЕ. Схема в базисе ИЛИ-НЕ представлена на рисунке 1.5. Если имеются ограничения на число входов ЛЭ и ранг конъюнкции (дизъюнкции) больше допустимого числа входов ЛЭ, то обычно выполняют дополнительные преобразования ПФ, например, факторизацию.

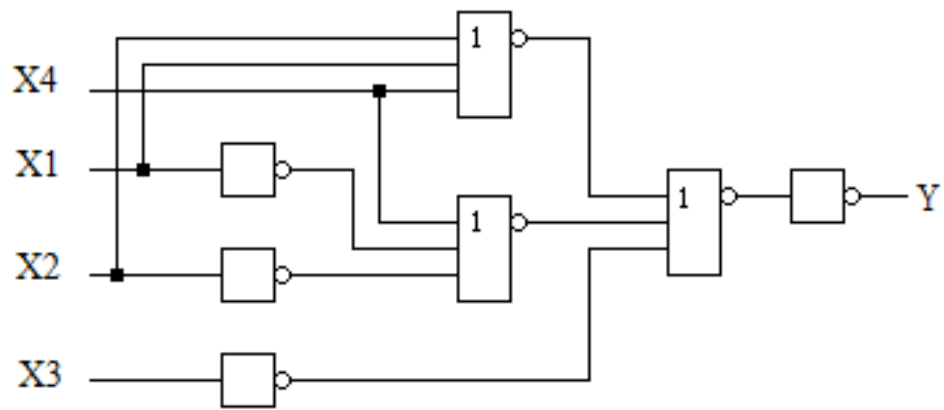


Рисунок 1.5 - Схема в базисе ИЛИ-НЕ

Реализуем ПФ на элементах 2И-НЕ:

$$y = \bar{x1} \cdot \bar{x2} \cdot \bar{x4} + x1 \cdot x2 \cdot \bar{x4} + \bar{x3} = (\bar{x1} \cdot \bar{x2} + x1 \cdot x2) \cdot \bar{x4} + \bar{x3} = \underline{\underline{\bar{x1} \cdot \bar{x2} \cdot x1 \cdot x2 \cdot \bar{x4} \cdot x3}}. \quad (1.8)$$

Схема в базисе И-НЕ представлена на рисунке 1.6.

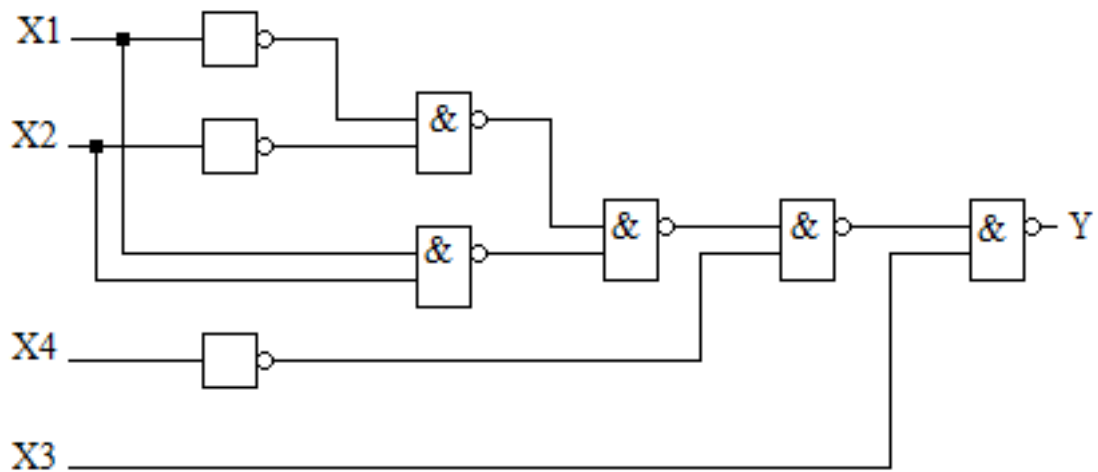


Рисунок 1.6 - Схема в базисе И-НЕ

Показатели оценки сложности и быстродействия КС для рассмотренных вариантов реализации ПФ приведены в таблице 1.1.

Таблица 1.1 - Показатели оценки сложности и быстродействия КС

Показатель	Вариант реализации				
	Рисунок 1.2	Рисунок 1.3	Рисунок 1.4	Рисунок 1.5	Рисунок 1.6
Число элементов	6	7	6	7	8
Критерий Квайна	12	13	12	13	13
Относительная задержка	4	3	3	4	5

Пример реализации схемы в базисе И-НЕ приведен на рисунке 1.7.

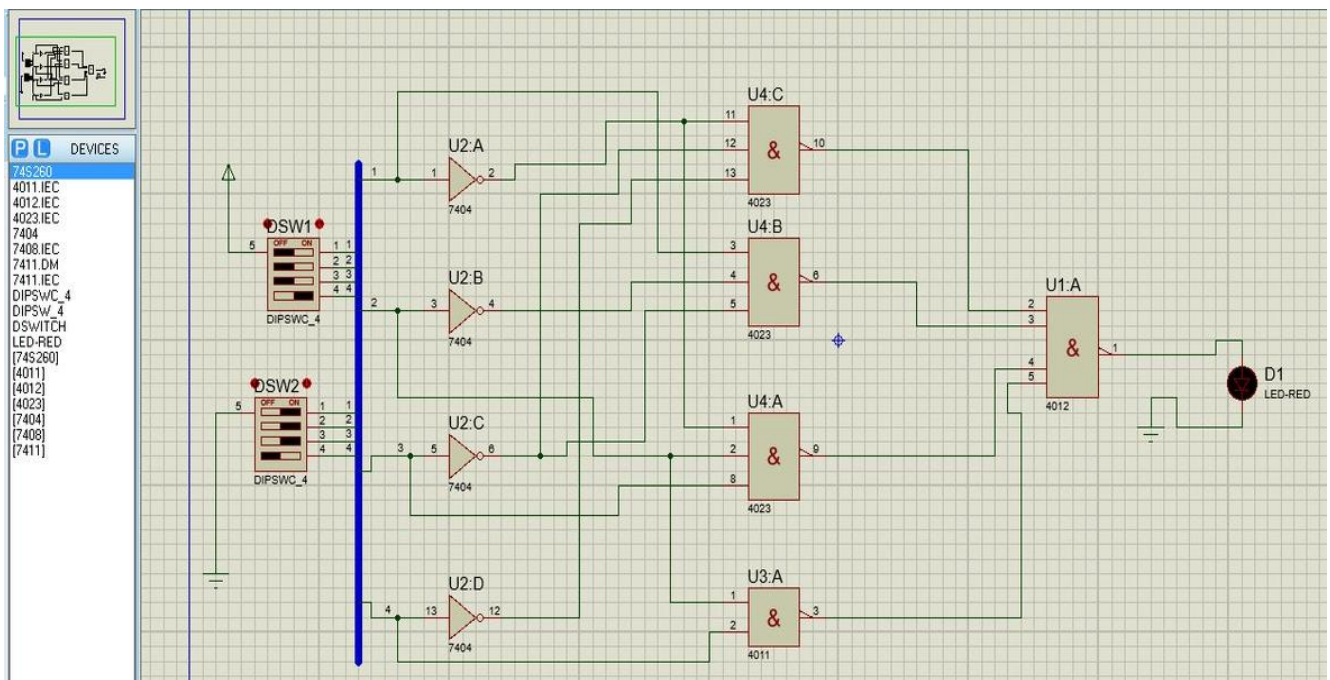


Рисунок 1.7 - Пример реализации схемы в базисе И-НЕ

Пример реализации схемы в базисе И-ИЛИ-НЕ приведен на рисунке 1.8.

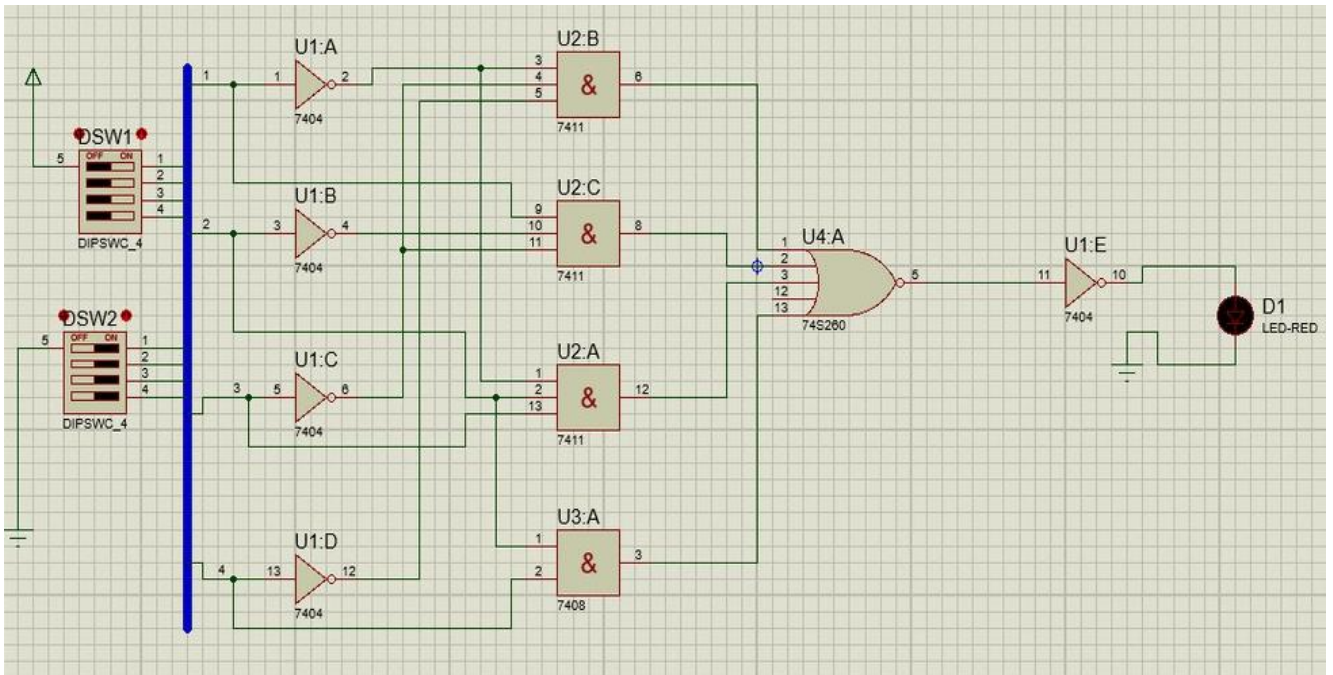


Рисунок 1.7 - Пример реализации схемы в базисе И-ИЛИ-НЕ

1.3 Задание

1.3.1 Составить таблицу истинности для заданной переключательной функции в соответствии с вариантом задания (таблица 1.2).

1.3.2 Составить карту Карно в соответствии с заданной функцией по указанному варианту.

1.3.3 Провести минимизацию функции. Записать выражения для СДНФ и СКНФ после исключения повторяющихся членов.

1.3.4 Разработать три электрические схемы в базисах логических элементов: И, ИЛИ, НЕ; И-НЕ; ИЛИ-НЕ.

1.3.5 Провести моделирование электронных схем в программной среде моделирования.

1.3.6 Выполнить сравнительный анализ полученных схем по сложности и быстродействию.

1.3.6 Составить отчет о выполненной работе.

Таблица 1.2 - Варианты заданий

Вариант	Функция алгебры логики
1	$f(x_1, x_2, x_3, x_4) = \bigvee_1(0, 1, 2, 4, 5, 6, 10, 12)$
2	$f(x_1, x_2, x_3, x_4) = \bigvee_1(0, 2, 3, 8, 9, 10, 11, 14, 15)$
3	$f(x_1, x_2, x_3, x_4) = \bigvee_1(0, 2, 4, 6, 7, 9, 12, 15)$
4	$f(x_1, x_2, x_3, x_4) = \bigvee_1(0, 1, 2, 3, 8, 10, 11, 12, 13, 15)$
5	$f(x_1, x_2, x_3, x_4) = \bigvee_1(1, 3, 5, 6, 7, 8, 10, 14, 15)$
6	$f(x_1, x_2, x_3, x_4) = \bigvee_1(3, 4, 5, 9, 10, 12, 14, 15)$
7	$f(x_1, x_2, x_3, x_4) = \bigvee_1(2, 4, 5, 6, 8, 9, 10, 14)$
8	$f(x_1, x_2, x_3, x_4) = \bigvee_1(1, 4, 8, 9, 10, 12, 13, 14, 15)$
9	$f(x_1, x_2, x_3, x_4) = \bigvee_1(1, 2, 4, 8, 9, 10, 12, 13, 15)$
10	$f(x_1, x_2, x_3, x_4) = \bigvee_1(0, 4, 5, 6, 7, 8, 9, 13, 15)$
11	$f(x_1, x_2, x_3, x_4) = \bigvee_1(4, 6, 7, 9, 10, 11, 12, 14, 15)$
12	$f(x_1, x_2, x_3, x_4) = \bigvee_1(3, 4, 5, 6, 9, 10, 12, 13, 14)$
13	$f(x_1, x_2, x_3, x_4) = \bigvee_1(2, 6, 7, 8, 11, 12, 13, 14, 15)$
14	$f(x_1, x_2, x_3, x_4) = \bigvee_1(1, 2, 3, 4, 7, 9, 10, 11, 14, 15)$
15	$f(x_1, x_2, x_3, x_4) = \bigvee_1(1, 4, 7, 8, 9, 10, 12, 13, 14, 15)$
16	$f(x_1, x_2, x_3, x_4) = \bigvee_1(3, 5, 6, 7, 8, 9, 10, 12, 13)$
17	$f(x_1, x_2, x_3, x_4) = \bigvee_1(4, 8, 9, 10, 11, 12, 14, 15)$
18	$f(x_1, x_2, x_3, x_4) = \bigvee_1(2, 3, 4, 6, 7, 8, 9, 11, 12)$
19	$f(x_1, x_2, x_3, x_4) = \bigvee_1(0, 6, 7, 8, 9, 10, 12, 13, 14, 15)$
20	$f(x_1, x_2, x_3, x_4) = \bigvee_1(1, 2, 4, 5, 6, 7, 11, 12, 15)$
21	$f(x_1, x_2, x_3, x_4) = \bigvee_1(1, 2, 4, 5, 7, 8, 10, 11, 12, 13)$
22	$f(x_1, x_2, x_3, x_4) = \bigvee_1(0, 1, 2, 5, 6, 7, 10, 11, 12, 15)$
23	$f(x_1, x_2, x_3, x_4) = \bigvee_1(0, 5, 6, 7, 8, 9, 10, 13, 14)$
24	$f(x_1, x_2, x_3, x_4) = \bigvee_1(3, 4, 7, 8, 9, 10, 13, 14, 15)$
25	$f(x_1, x_2, x_3, x_4) = \bigvee_1(1, 2, 3, 6, 7, 9, 10, 12, 13, 14)$

1.4 Контрольные вопросы

1 Что такое схемотехника и каковы основные задачи этой дисциплины?

2 Дать определение аналоговых и дискретных электронных устройств. Перечислить их достоинства и недостатки.

3 Дать характеристику импульсных, релейных и цифровых электронных устройств. Пояснить с помощью графиков виды квантования.

4 Дать характеристику элементной базы цифровых устройств.

5 Дать определение логического элемента. Назвать основные виды ЛЭ, привести их таблицы истинности и условные обозначения.

6 Что такое положительная и отрицательная логика: показать на графиках.

7 Назвать способы представления логических функций, привести примеры.

8 Дать определение и назначение минимизации переключательных функций. Пояснить порядок минимизации по картам Карно на примере.

9 Дать определение логического базиса. Назвать существующие базисы.

10 Привести примеры реализации логических функций в разных базисах.

11 Дать характеристику параметров реального импульса. Назвать способы получения импульсных сигналов.

12 Назвать основные статические характеристики логических микросхем.

13 Назвать основные динамические характеристики логических микросхем.

14 Назвать типы логик, дать их краткую характеристику, назвать достоинства и недостатки.

2 Лабораторная работа № 2. Реализация логических функций на мультиплексорах

2.1 Цель работы

Освоить навыки построения функциональных схем на мультиплексорах с заданным количеством входов.

2.2 Теоретические сведения

Комбинационное цифровое устройство, которое осуществляет выборку одного из нескольких входов и подключает его к своему выходу под действием управляющего (адресного) кода, называется мультиплексором.

Мультиплексор имеет информационные входы ($X_0, X_1 \dots X_n$), адресные входы ($A_0, A_1 \dots A_m$), вход для подачи разрешающего сигнала E и один выход F . Вход стробирования E позволяет избежать появления на выходах ложных сигналов при чередующихся на входе наборах, кодовое расстояние между которыми больше единицы. Условное графическое обозначение мультиплексора на схемах показано на рисунке 2.1.

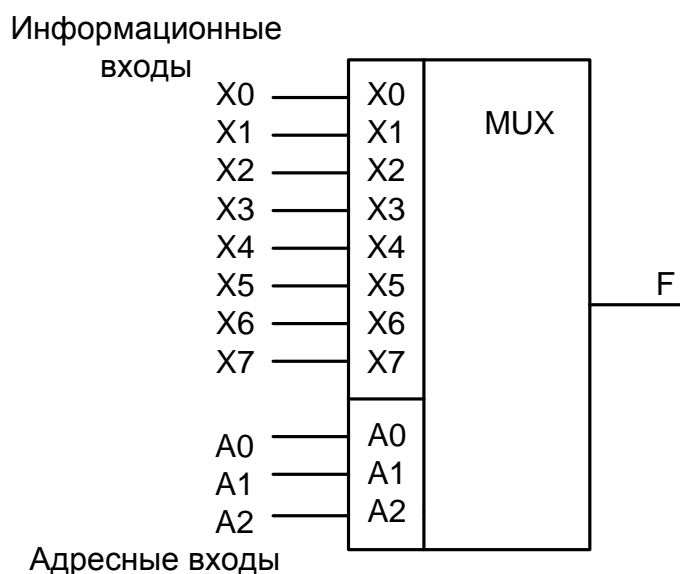


Рисунок 2.1 - Условное графическое обозначение мультиплексора

Символически мультиплексор можно представить многоканальным коммутатором, имеющим одностороннюю передачу данных (рисунок 2.2).

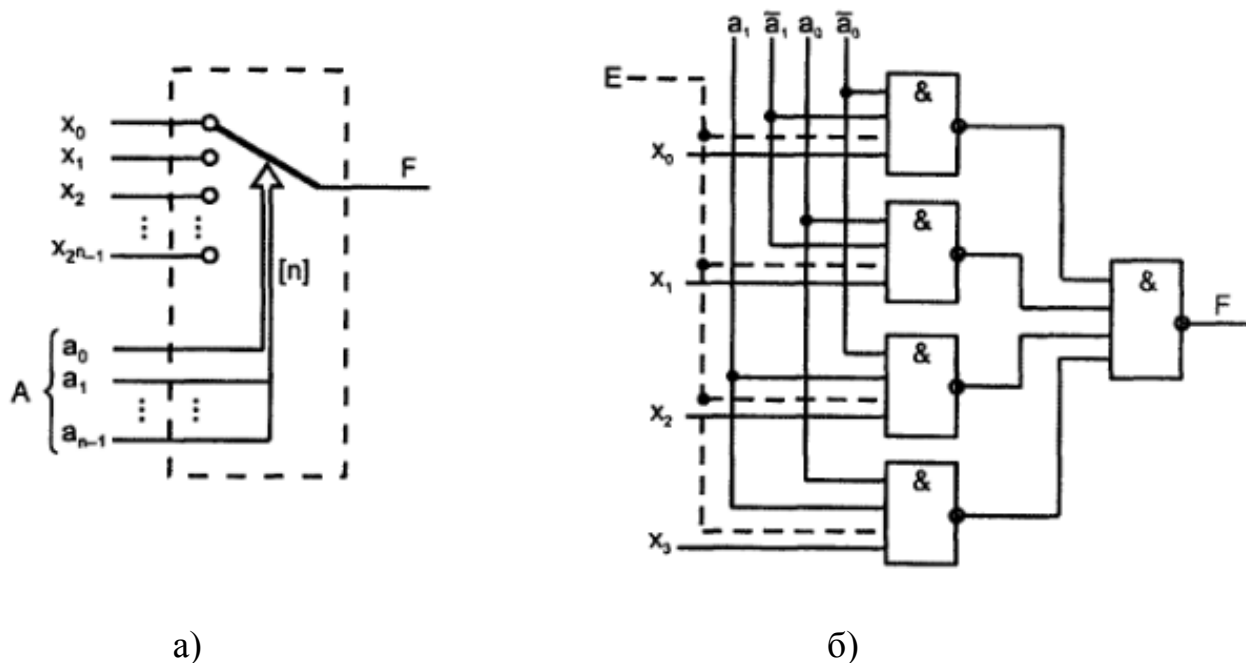


Рисунок 2.2 – Символическое изображение мультиплексора (а), реализация мультиплексора на элементах И-НЕ (б)

Адресующий код A задает переключателю определенное положение, соединяя с выходом F один из информационных входов X . При нулевом адресующем коде переключатель занимает верхнее положение $X0$, с увеличением кода на единицу переходит в соседнее положение $X1$ и т. д. Работа мультиплексора описывается мультиплексной формулой

$$F = x_0 \bar{a}_{n-1} \bar{a}_{n-2} \dots \bar{a}_1 \bar{a}_0 \vee x_1 \bar{a}_{n-1} \bar{a}_{n-2} \dots \bar{a}_1 a_0 \vee \dots x_{2^n-1} a_{n-1} a_{n-2} \dots a_1 a_0, \quad (2.1)$$

При любом значении адресующего кода все слагаемые, кроме одного, равны нулю. Ненулевое слагаемое равно X_i , где i — значение текущего адресного кода. На рис. 2.1, б показан мультиплексор с четырьмя информационными входами, двумя адресными входами и входом разрешения работы. При отсутствии разрешения работы ($E = 0$) выход F становится нулевым независимо от информационных и адресных сигналов.

Для обозначения коммутационных возможностей мультиплексора пользуются условной записью $(n \rightarrow 1)$, где n – это число информационных входов. В зависимости от соотношения числа информационных входов n и числа адресных входов m мультиплексоры делятся на полные и неполные. Если выполняется условие, что $n=2^m$, то мультиплексор полный, в противном случае – неполный. Наибольшее распространение получили мультиплексоры:

$$(2 \rightarrow 1) \quad n=2 \quad m=1;$$

$$(4 \rightarrow 1) \quad n=4 \quad m=2;$$

$$(8 \rightarrow 1) \quad n=8 \quad m=3;$$

$$(16 \rightarrow 1) \quad n=16 \quad m=4.$$

Для неполных мультиплексоров число входов может быть любым, но не больше, чем $n=2^m$. Выходы мультиплексора называют разрядами. Например, 4-канальный 2-разрядный мультиплексор имеет 2 выхода, на каждый из которых может передаваться один из 4-х входных сигналов.

Некоторые мультиплексоры могут иметь выход с тремя состояниями: 0; 1; третье состояние - отключенный выход (выходное сопротивление равно бесконечности), обозначается ОЕ.

Наращивание размерности. Нарастивание размерности мультиплексоров возможно с помощью пирамидальной структуры из нескольких мультиплексоров. При этом первый ярус схемы представляет собою столбец, содержащий столько мультиплексоров, сколько необходимо для получения нужного числа информационных входов. Все мультиплексоры столбца адресуются одним и тем же кодом, составленным из соответствующего числа младших разрядов общего адресного кода (если число информационных входов схемы равно 2^n , то общее число адресных разрядов равно n , младшее поле $n1$ адресного кода используется для адресации мультиплексоров первого яруса). Старшие разряды адресного кода, число которых равно $n - n1$, используются во втором ярусе, мультиплексор которого обеспечивает поочередную работу мультиплексоров первого яруса на общий выходной канал. Схема, выполняющая функции мультиплексора «32-1» и построенная на мультиплексорах меньшей размерности, показана на рисунке 2.3.

Мультиплексоры могут использоваться в качестве многофункциональных элементов для реализации различных логических функций.

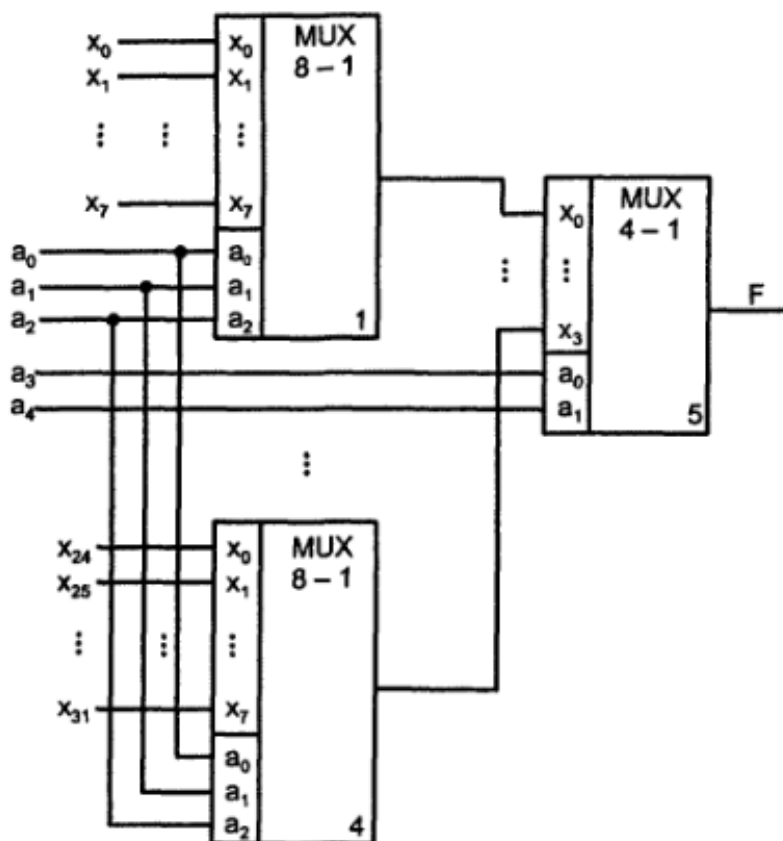


Рисунок 2.3- Схема наращивания мультиплексоров

2.3 Задание

Варианты заданий заданы в таблице 2.1.

2.3.1 Составить таблицу истинности для мультиплексора с 16 входами и построить функциональную схему, при этом на управляющие входы мультиплексора подать младшие переменные.

2.3.2 Составить таблицу истинности для мультиплексора с 8 входами и построить функциональную схему;

2.3.3 Составить таблицу истинности для мультиплексора с 4 входами и построить функциональную схему;

2.3.4 Построить схемы принципиальные электрические для всех трех вариантов реализации логической функции на интегральных мультиплексорах в среде Proteus. Провести отладку схем.

Таблица 2.1- Варианты заданий

Вариант	Функция
1	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (3, 4, 5, 9, 10, 12, 14, 15)$
2	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (2, 4, 5, 6, 8, 9, 10, 14)$
3	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (1, 4, 8, 9, 10, 12, 13, 14, 15)$
4	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (1, 2, 4, 8, 9, 10, 12, 13, 15)$
5	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (0, 4, 5, 6, 7, 8, 9, 13, 15)$
6	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (4, 6, 7, 9, 10, 11, 12, 14, 15)$
7	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (3, 4, 5, 6, 9, 10, 12, 13, 14)$
8	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (2, 6, 7, 8, 11, 12, 13, 14, 15)$
9	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (1, 2, 3, 4, 7, 9, 10, 11, 14, 15)$
10	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (1, 4, 7, 8, 9, 10, 12, 13, 14, 15)$
11	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (3, 5, 6, 7, 8, 9, 10, 12, 13)$
12	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (4, 8, 9, 10, 11, 12, 14, 15)$
13	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (2, 3, 4, 6, 7, 8, 9, 11, 12)$
14	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (0, 6, 7, 8, 9, 10, 12, 13, 14, 15)$
15	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (1, 2, 4, 5, 6, 7, 11, 12, 15)$
16	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (1, 2, 4, 5, 7, 8, 10, 11, 12, 13)$
17	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (0, 1, 2, 5, 6, 7, 10, 11, 12, 15)$
18	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (0, 1, 2, 4, 5, 6, 10, 12)$
19	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (3, 4, 7, 8, 9, 10, 13, 14, 15)$
20	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (1, 2, 3, 6, 7, 9, 10, 12, 13, 14)$
21	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (2, 3, 4, 6, 8, 10, 11, 12, 15)$
22	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (0, 2, 3, 4, 5, 7, 8, 9, 10, 12, 15)$
23	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (3, 5, 6, 7, 8, 9, 11, 12, 13)$
24	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (2, 4, 6, 7, 8, 9, 10, 13, 14, 15)$
25	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (1, 5, 6, 7, 8, 9, 10, 12, 13, 14)$
26	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (0, 1, 2, 4, 5, 6, 10, 12)$
27	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (0, 2, 3, 8, 9, 10, 11, 14, 15)$
28	$f(x_1, x_2, x_3, x_4) = \bigvee_1 (0, 2, 4, 6, 7, 9, 12, 15)$

Пример.

Дана функция $f(x_1, x_2, x_3, x_4) = \bigvee_1(0,5,6,7,8,9,10,13,14)$. Составить таблицы

истинности и функциональные схемы на основе мультиплексоров и составить электрические принципиальные схемы.

Решение.

1 Составим таблицу истинности мультиплексора 16→1 (таблица 2.2). Функциональная схема представлена на рисунке 2.4

Таблица 2.2 - Таблица истинности мультиплексора 16→1

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

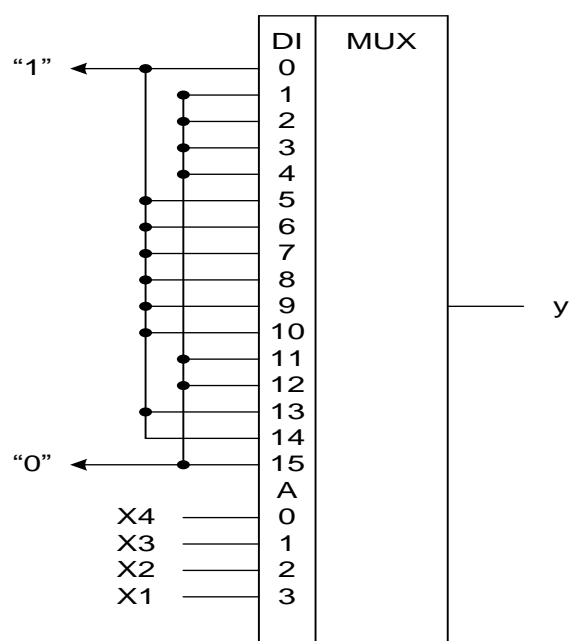


Рисунок 2.4 - Функциональная схема мультиплексора 16→1

2 Составим таблицу истинности (таблица 2.3) для мультиплексора с 8-ю входным набором данных.

Таблица 2.3 - Таблица истинности мультиплексора 8→1

i	i	x ₁	x ₂	x ₃	x ₄	y	y
0	0	0	0	0	0	1	$\overline{x_4}$
	1	0	0	0	1	0	
1	2	0	0	1	0	0	0
	3	0	0	1	1	0	
2	4	0	1	0	0	0	x ₄
	5	0	1	0	1	1	
3	6	0	1	1	0	1	1
	7	0	1	1	1	1	
4	8	1	0	0	0	1	1
	9	1	0	0	1	1	
5	10	1	0	1	0	1	$\overline{x_4}$
	11	1	0	1	1	0	
6	12	1	1	0	0	0	x ₄
	13	1	1	0	1	1	
7	14	1	1	1	0	1	$\overline{x_4}$
	15	1	1	1	1	0	

Составим функциональную схему (рисунок 2.5).

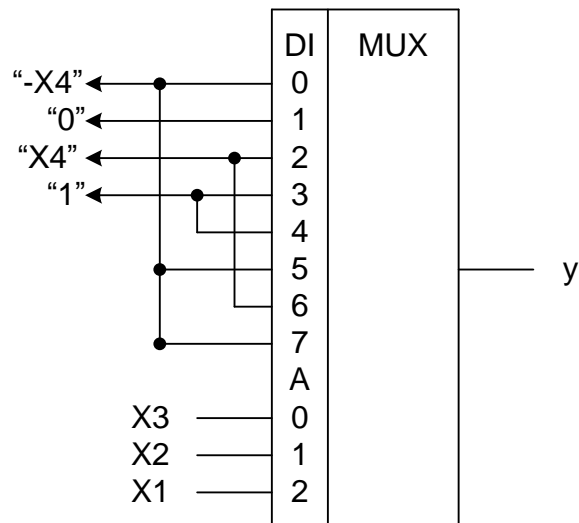


Рисунок 2.5 - Функциональная схема

3 Составим таблицу истинности (таблица 2.4) для мультиплексора с 4-мя входным набором данных.

Таблица 3 - Таблица истинности мультиплексора 4→1

i	i	x ₁	x ₂	x ₃	x ₄	y	y
0	0	0	0	0	0	1	$\overline{x_3} \vee \overline{x_4}$
	1	0	0	0	1	0	
	2	0	0	1	0	0	
	3	0	0	1	1	0	
1	4	0	1	0	0	0	$x_3 \vee x_4$
	5	0	1	0	1	1	
	6	0	1	1	0	1	
	7	0	1	1	1	1	
2	8	1	0	0	0	1	$\overline{x_3} \vee \overline{x_4}$
	9	1	0	0	1	1	
	10	1	0	1	0	1	
	11	1	0	1	1	0	
3	12	1	1	0	0	0	$x_3 \oplus x_4$
	13	1	1	0	1	1	
	14	1	1	1	0	1	
	15	1	1	1	1	0	

Составим функциональную схему (рисунок 2.6).

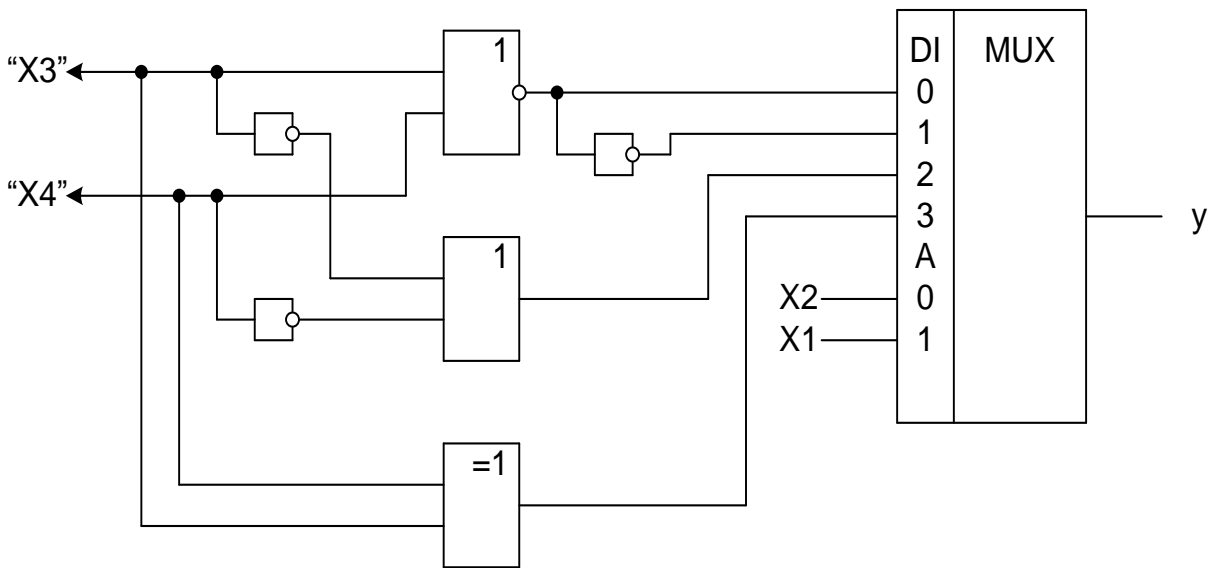


Рисунок 2.6 - Функциональная схема

Пример реализации в Proteus схемы с другой переключательной функцией представлен на рисунке 2.7, 2.8.

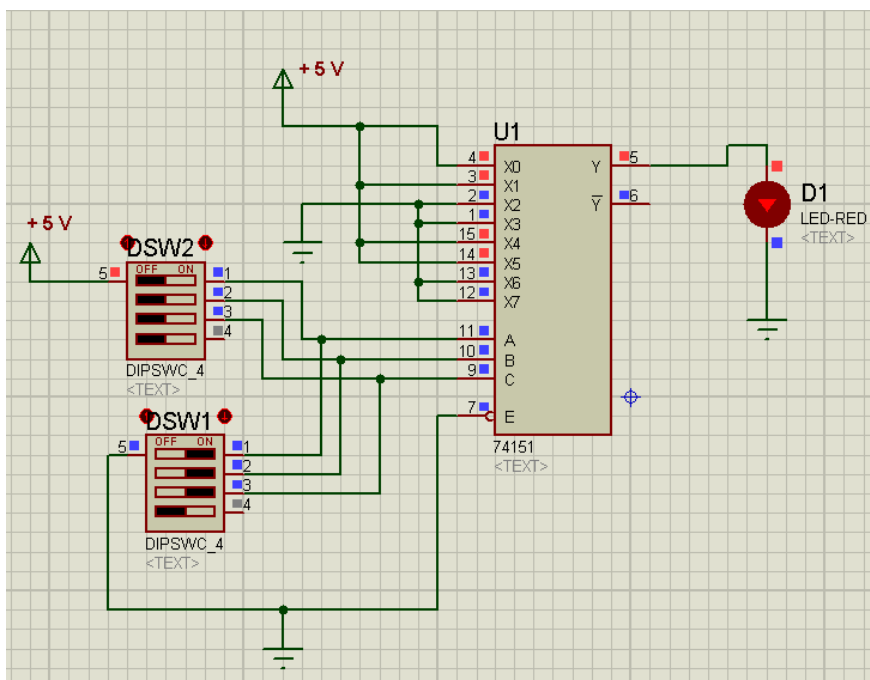


Рисунок 2.7 – Пример реализации в Proteus схемы мультиплексора 8→1

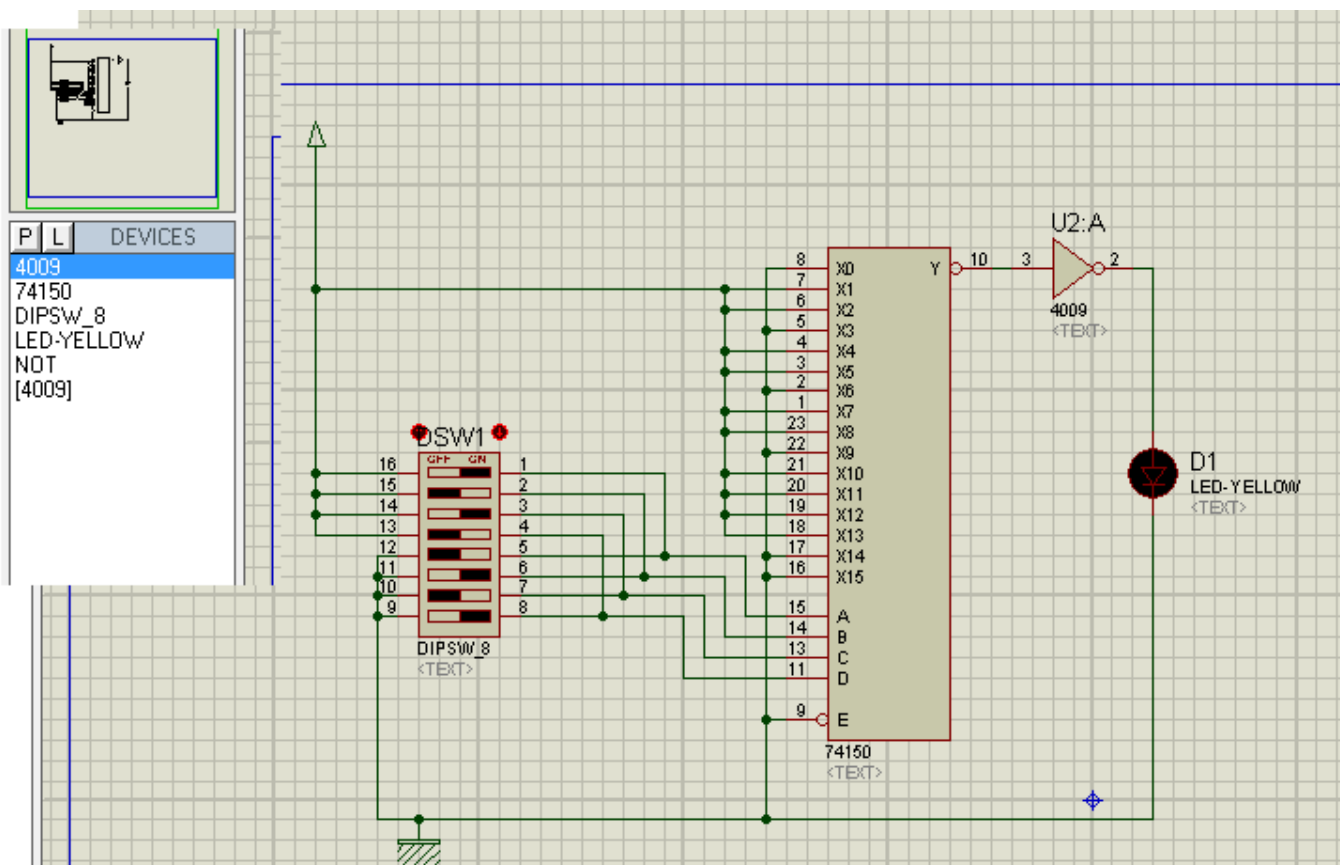


Рисунок 2.8 – Пример реализации в Proteus схемы 16→1

2.4 Контрольные вопросы

1 Дать определение комбинационных и последовательностных логических устройств. В чем их различие? Назвать примеры цифровых устройства, относящихся к каждой из этих классов.

2 Дать определение мультиплексора, привести упрощенное представление мультиплексора многопозиционным ключом. Привести условное графическое обозначение мультиплексора на электрических схемах, пояснить назначение выводов, соотношение их количества.

3 Что такое мультиплексная формула, дать определение полного и неполного мультиплексора, привести примеры. Записать мультиплексную формулу для мультиплексора $8 \rightarrow 1$.

4 Для чего используются разрешающие входы мультиплексора? Привести пример использования разрешающих входов.

5 Привести схемы расширения числа входов мультиплексора, объяснить принцип их работы.

6 Назвать области и привести примеры использования мультиплексоров в электронных схемах.

7 Показать реализацию мультиплексора на элементах И-НЕ, пояснить принцип работы схемы.

8 Дать определение демультиплексора, привести условное графическое обозначение демультиплексора на электрических схемах, пояснить назначение выводов, соотношение их количества.

9 Объяснить принцип работы демультиплексора.

10 Какую схему используют для наращивания демультиплексоров? Поясните принцип работы такой схемы.

11 Что такое универсальные логические модули на основе мультиплексоров? Привести примеры.

12 Пояснить способы настройки универсальных логических модулей.

ций системы и минимум рангов этих термов. При этом один и тот же терм может входить в покрытие нескольких функций.

Пример 1.

На рисунке 3.1 приведены карты Карно системы ПФ y_1, y_2, y_3 , на которых отмечено её минимальное покрытие из четырех склеек.

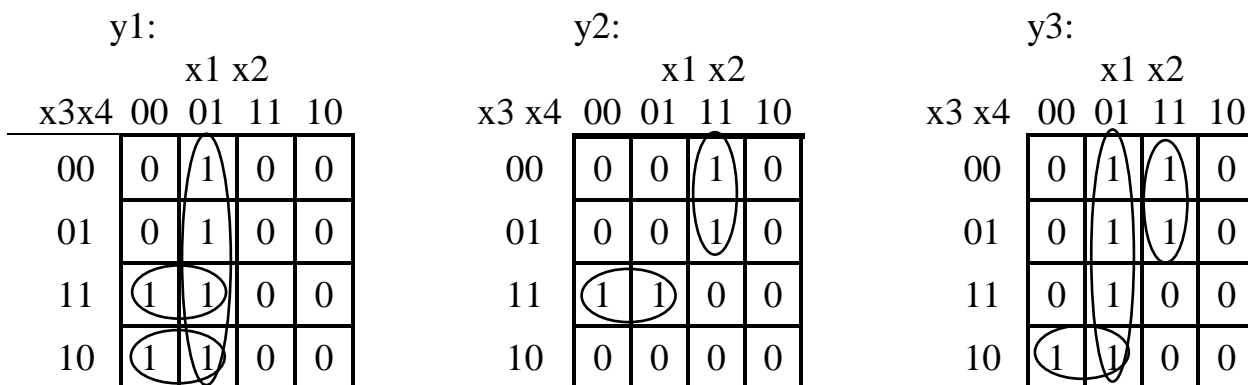


Рисунок 3.1 – Карты Карно системы ПФ y_1, y_2, y_3

Следует отметить, что ДНФ функций y_1 и y_3 не являются минимальными

$$\begin{aligned}
 y_1 &= \bar{x}_1 \cdot x_3 \cdot x_4 + \bar{x}_1 \cdot x_3 \cdot \bar{x}_4 + \bar{x}_1 \cdot x_2, \\
 y_2 &= x_1 \cdot x_2 \cdot x_3 + \bar{x}_1 \cdot x_3 \cdot x_4, \\
 y_3 &= x_1 \cdot x_2 \cdot x_3 + \bar{x}_1 \cdot x_2 + \bar{x}_1 \cdot x_3 \cdot \bar{x}_4.
 \end{aligned}
 \tag{3.2}$$

Полученная функциональная схема приведена на рисунке 3.2. При ее формировании использована линия групповой связи, которая обозначает множество электрических линий связи.

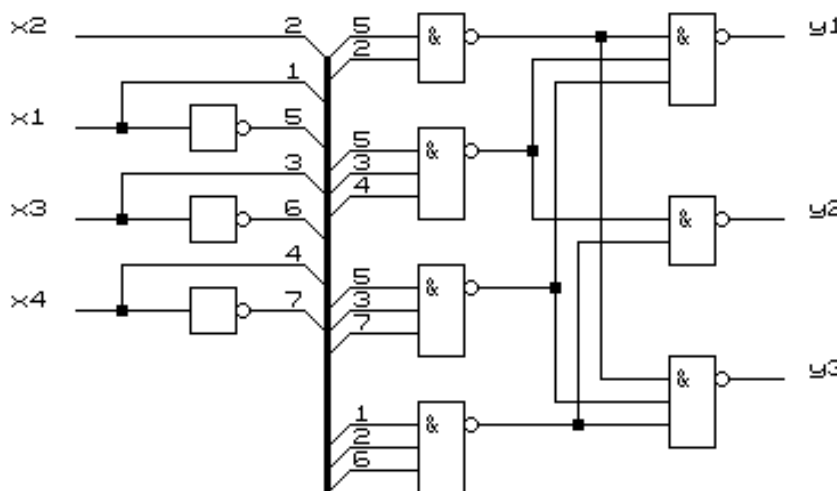


Рисунок 3.2 – Функциональная схема переключательных функций

Пример 2.

Синтезировать в базисе И-НЕ схему дешифратора для управления семисегментным индикатором, работающую согласно таблицы 3.1. При поступлении на вход дешифратора соответствующей комбинации сигналов $x1$, $x2$, $x3$ индикатор должен высвечивать цифры 1, 2, 5, 7.

Таблица 3.1 – Таблица истинности

$x1$	$x2$	$x3$	Индикатор
0	0	0	*
0	0	1	1
0	1	0	2
0	1	1	*
1	0	0	*
1	0	1	5
1	1	0	*
1	1	1	7

Наборы входных переменных, отмеченные знаком *, на вход дешифратора не подаются и, следовательно, система ПФ будет не полностью определенной. Расположение сегментов индикатора показано на рисунке 3.3.

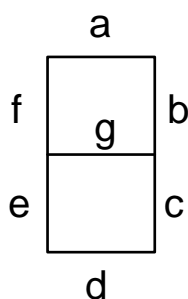


Рисунок 3.3 – Расположение сегментов индикатора

Считая, что излучать будет тот сегмент, на который подана логическая единица, составим систему из семи ПФ, каждая из которых определяет работу одного из сегментов индикатора. Например, чтобы на индикаторе высветилась цифра 5, необходимо подать 1 на сегменты a, f, g, c, d , то есть

$$a = f = g = c = d = 1, \quad b = e = 0.$$

Таблица истинности системы ПФ представлена в таблице 3.2.

Таблица 3.2 – Таблица истинности системы ПФ

x1	x2	x3	Инд	a	b	c	d	e	f	g
0	0	0	*	*	*	*	*	*	*	*
0	0	1	1	0	1	1	0	0	0	0
0	1	0	2	1	1	0	1	1	0	1
0	1	1	*	*	*	*	*	*	*	*
1	0	0	*	*	*	*	*	*	*	*
1	0	1	5	1	0	1	1	0	1	1
1	1	0	*	*	*	*	*	*	*	*
1	1	1	7	1	1	1	0	0	0	0

Доопределив полученные функции и минимизировав их, получим следующие соотношения для каждого сегмента:

$$a = x1 + x2 ; \quad b = \overline{x1} + x2 ; \quad c = x3 ;$$

$$d = x1 \cdot \overline{x2} + \overline{x3} ; \quad e = \overline{x3} ; \quad f = x1 \cdot \overline{x2} ; \quad g = x1 \cdot \overline{x2} + \overline{x3} .$$

Воспользовавшись правилом де-Моргана переведем полученные ПФ в базис И-НЕ:

$$a = \overline{\overline{x1} \cdot \overline{x2}} ; \quad b = \overline{\overline{x1} \cdot \overline{x2}} ; \quad c = x3 ;$$

$$d = \overline{\overline{\overline{x1 \cdot x2 \cdot x3}}} ; \quad e = \overline{x3} ; \quad f = x1 \cdot \overline{x2} ; \quad g = \overline{\overline{\overline{x1 \cdot x2 \cdot x3}}} .$$

Анализируя эти ПФ видим, что $b = \overline{f}$, $d = g$, b участвует в образовании функций d и g :

$$d = g = \overline{b \cdot x3} .$$

Функциональная схема дешифратора для семисегментного индикатора, построенная по полученным выше соотношениям, приведена на рисунке 3.4.

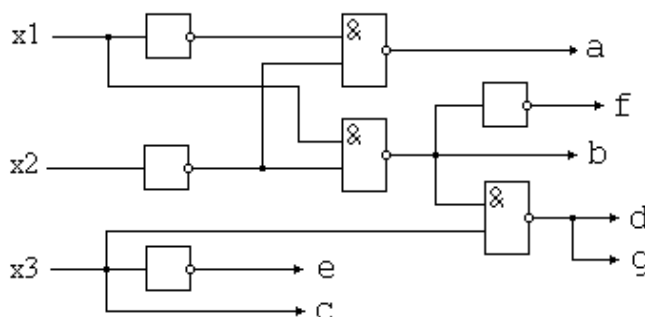


Рисунок 3.4 – Функциональная схема дешифратора

Шифраторы (кодеры) используются чаще всего для преобразования десятичных чисел в двоичный или двоично-десятичный код, например, в микрокалькуляторах, в которых нажатие десятичной клавиши соответствует генерации соответствующего двоичного кода. Поскольку возможно нажатие сразу нескольких клавиш, в шифраторах используется принцип приоритета старшего разряда.

Дешифратор (декодер) – устройство с несколькими входами и выходами, у которого определенным комбинациям входных сигналов соответствует активное состояние одного из выходов, т.е. дешифратор является обращенным по входам демультиплексором, у которого адресные входы стали информационными, а бывший информационный вход стал входом разрешения. Поэтому часто дешифраторы называют дешифраторами-демультиплексорами и наоборот. Условное графическое обозначение дешифратора приведено на рисунке 3.5.

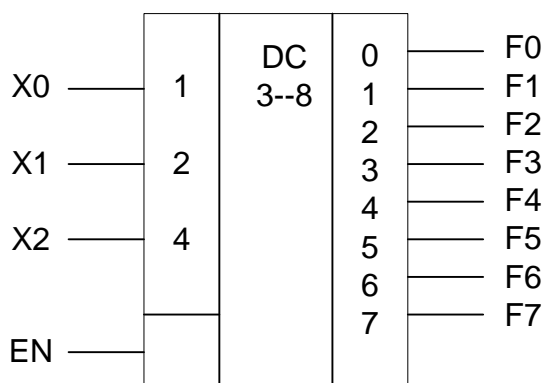


Рисунок 3.5 – Условное графическое обозначение дешифратора

Дешифраторы и демультиплексоры в виде серийных ИМС средней степени интеграции широко используются в информационно-измерительной технике и микропроцессорных системах управления, в частности, в качестве коммутаторов распределителей информационных сигналов и синхроимпульсов, для демультиплексирования данных и адресной логики в запоминающих устройствах, а также для преобразования двоично-десятичного кода в десятичный с целью управления индикаторными и печатающими устройствами.

Дешифраторы как самостоятельные изделия электронной техники имеют 4, 8 или 16 выходов. Если требуется большее число выходов, дешифраторы наращиваются в систему.

3.3 Задание

3.3.1 Построить комбинационную схему с множеством выходов в заданном базисе элементов для заданного варианта, данного в таблице 3.3 (по примеру 1).

3.3.2 Для заданного варианта (таблица 3.4) составить систему ПФ ($f_a, f_b, f_c, f_d, f_e, f_f, f_g$) (по примеру 2).

3.3.3 Реализовать схему для семисегментных индикаторов на интегральной микросхеме дешифратора.

3.3.4 Построить в среде Proteus схемы для пунктов 3.3.2 и 3.3.3. (Количество схем - три).

Таблица 3.3 – Варианты заданий

Вариант	Функция алгебры логики	Базис
1	$Y1(x1,x2,x3,x4) = \bigvee_1(0, 1, 2, 4, 5, 6, 10, 12)$	И, ИЛИ, НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(0, 2, 3, 8, 9, 10, 11, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(0, 2, 4, 6, 7, 9, 12, 15)$	
2	$Y1(x1,x2,x3,x4) = \bigvee_1(0, 1, 2, 3, 8, 10, 11, 12, 13, 15)$	И-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(1, 3, 5, 6, 7, 8, 10, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(3, 4, 5, 9, 10, 12, 14, 15)$	
3	$Y1(x1,x2,x3,x4) = \bigvee_1(2, 4, 5, 6, 8, 9, 10, 14)$	ИЛИ-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(1, 4, 8, 9, 10, 12, 13, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(1, 2, 4, 8, 9, 10, 12, 13, 15)$	
4	$Y1(x1,x2,x3,x4) = \bigvee_1(0, 4, 5, 6, 7, 8, 9, 13, 15)$	ИЛИ-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(4, 6, 7, 9, 10, 11, 12, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(3, 4, 5, 6, 9, 10, 12, 13, 14)$	

Продолжение таблицы 3.3

Вариант	Функция алгебры логики	Базис
5	$Y1(x1,x2,x3,x4) = \bigvee_1(2, 6, 7, 8, 11, 12, 13, 14, 15)$	И, ИЛИ, НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(1, 2, 3, 4, 7, 9, 10, 11, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(1, 4, 7, 8, 9, 10, 12, 13, 14, 15)$	
6	$Y1(x1,x2,x3,x4) = \bigvee_1(3, 5, 6, 7, 8, 9, 10, 12, 13)$	И-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(4, 8, 9, 10, 11, 12, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(2, 3, 4, 6, 7, 8, 9, 11, 12)$	
7	$Y1(x1,x2,x3,x4) = \bigvee_1(0, 6, 7, 8, 9, 10, 12, 13, 14, 15)$	ИЛИ-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(1, 2, 4, 5, 6, 7, 11, 12, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(1, 2, 4, 5, 7, 8, 10, 11, 12, 13)$	
8	$Y1(x1,x2,x3,x4) = \bigvee_1(0, 1, 2, 5, 6, 7, 10, 11, 12, 15)$	И-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(0, 5, 6, 7, 8, 9, 10, 13, 14)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(3, 4, 7, 8, 9, 10, 13, 14, 15)$	
9	$Y1(x1,x2,x3,x4) = \bigvee_1(1, 2, 3, 6, 7, 9, 10, 12, 13, 14)$	ИЛИ-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(4, 8, 9, 10, 11, 12, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(2, 3, 4, 6, 7, 8, 9, 11, 12)$	
10	$Y1(x1,x2,x3,x4) = \bigvee_1(3, 5, 6, 7, 8, 9, 10, 12, 13)$	И, ИЛИ, НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(4, 6, 7, 9, 10, 11, 12, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(3, 4, 5, 6, 9, 10, 12, 13, 14)$	
11	$Y1(x1,x2,x3,x4) = \bigvee_1(1, 2, 4, 5, 6, 7, 11, 12, 15)$	ИЛИ-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(1, 2, 4, 5, 7, 8, 10, 11, 12, 13)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(0, 1, 2, 5, 6, 7, 10, 11, 12, 15)$	
12	$Y1(x1,x2,x3,x4) = \bigvee_1(0, 5, 6, 7, 8, 9, 10, 13, 14)$	И, ИЛИ, НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(3, 4, 7, 8, 9, 10, 13, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(1, 2, 3, 6, 7, 9, 10, 12, 13, 14)$	
13	$Y1(x1,x2,x3,x4) = \bigvee_1(2, 3, 4, 6, 8, 10, 11, 12, 15)$	ИЛИ-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(0, 2, 3, 4, 5, 7, 8, 9, 10, 12, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(3, 5, 6, 7, 8, 9, 11, 12, 13)$	

Продолжение таблицы 3.3

Вариант	Функция алгебры логики	Базис
14	$Y1(x1,x2,x3,x4) = \bigvee_1(2, 4, 6, 7, 8, 9, 10, 13, 14, 15)$	И-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(1, 5, 6, 7, 8, 9, 10, 12, 13, 14)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(3, 4, 5, 6, 9, 10, 12, 13, 14)$	
15	$Y1(x1,x2,x3,x4) = \bigvee_1(2, 6, 7, 8, 11, 12, 13, 14, 15)$	И, ИЛИ, НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(1, 2, 3, 4, 7, 9, 10, 11, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(1, 4, 7, 8, 9, 10, 12, 13, 14, 15)$	
16	$Y1(x1,x2,x3,x4) = \bigvee_1(3, 5, 6, 7, 8, 9, 10, 12, 13)$	ИЛИ-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(4, 8, 9, 10, 11, 12, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(2, 3, 4, 6, 7, 8, 9, 11, 12)$	
17	$Y1(x1,x2,x3,x4) = \bigvee_1(0, 6, 7, 8, 9, 10, 12, 13, 14, 15)$	И-НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(0, 2, 3, 8, 9, 10, 11, 14, 15)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(0, 2, 4, 6, 7, 9, 12, 15)$	
18	$Y1(x1,x2,x3,x4) = \bigvee_1(2, 3, 4, 6, 8, 10, 11, 12, 15)$	И, ИЛИ, НЕ
	$Y2(x1,x2,x3,x4) = \bigvee_1(1, 5, 6, 7, 8, 9, 10, 12, 13, 14)$	
	$Y3(x1,x2,x3,x4) = \bigvee_1(3, 5, 6, 7, 8, 9, 11, 12, 13)$	

Таблица 3.4- Варианты заданий для индикаторов

x2 x3 x4	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0 0 0	A	*	b	U	P	d	C	*	U	P	*	*	A	У	*	F	H	d
0 0 1	b	C	*	*	O	F	F	b	O	A	C	P	C	H	U	L	O	*
0 1 0	C	E	C	У	U	E	A	d	*	d	У	b	L	P	b	H	*	E
0 1 1	*	F	P	O	d	U	*	*	L	L	P	*	F	*	*	O	*	O
1 0 0	O	L	*	H	C	*	L	E	b	E	L	E	*	F	C	*	b	U
1 0 1	d	*	H	L	*	b	O	O	E	H	*	H	U	d	F	P	У	*
1 1 0	E	A	L	*	F	0	d	У	*	*	O	*	P	C	O	U	P	У
1 1 1	P	d	F	*	У	A	U	*	A	У	H	У	H	*	*	У	U	A

3.4 Контрольные вопросы

1 Дать определение дешифратора. Привести условное обозначение микросхемы дешифратора на схемах. Пояснить назначение выводов и принцип работы.

2 Привести схемотехническое представление дешифратора и его формулу, таблицу истинности. Что такое полный и неполный дешифратор?

3 Построить схему наращивания дешифраторов для получения 16 выходов, используя дешифраторы 2-4.

4 Построить схему на дешифраторах совместно со схемами ИЛИ для воспроизведения логических функций

$$F_1 = \bar{x}_3 \bar{x}_2 x_1 \vee x_3 \bar{x}_1$$

$$F_2 = \bar{x}_3 \bar{x}_2 \vee x_2 \bar{x}_1$$

5 Построить схему на дешифраторах совместно со схемами ИЛИ для воспроизведения логических функций

$$F_1 = \bar{x}_2 x_1 \vee x_3 \bar{x}_1$$

$$F_2 = \bar{x}_3 \bar{x}_2 \vee x_2 \bar{x}_1 x_3$$

6 Привести примеры применения дешифраторов.

7 Что такое шифратор? Дать определение полного, неполного шифратора, привести условное обозначение, пояснить назначение выводов.

8 Дать определение приоритетного и двоичного шифратора, объяснить принцип работы.

4 Лабораторная работа № 4. Цифровые компараторы

4.1 Цель работы

- Изучение принципа работы компаратора.
- Синтез компараторов на логических элементах.

4.2 Теоретические сведения

Цифровые компараторы (от английского compare – сравнивать) выполняют сравнение двух чисел A , B одинаковой разрядности, заданных в двоичном коде, имеют три выхода. В зависимости от схемного исполнения компараторы могут определять равенство $A=B$ или неравенства $A<B$, $A>B$. Результат сравнения отображается в виде логического сигнала на одноименных выходах.

Цифровые компараторы применяются для выявления нужного числа в цифровых последовательностях, для отметки времени в часовых приборах, для выполнения условных переходов в вычислительных устройствах, а также в адресных селекторах.

В интегральных сериях цифровых элементов обычно имеются компараторы с тремя выходами: «равно», «больше» и «меньше». На рисунке 4.1 показано условное графическое обозначение интегральной микросхемы компаратора.



Рисунок 4.1 - Условное графическое обозначение компаратора

На рисунке 4.1 показан четырехразрядный компаратор, который выполняет сравнение четырехразрядных двоичных чисел А и В.

Для простоты рассмотрим схему одноразрядного компаратора, которая представляет собой структуру логического элемента «исключающее ИЛИ-НЕ», представленную на рисунке 4.2.

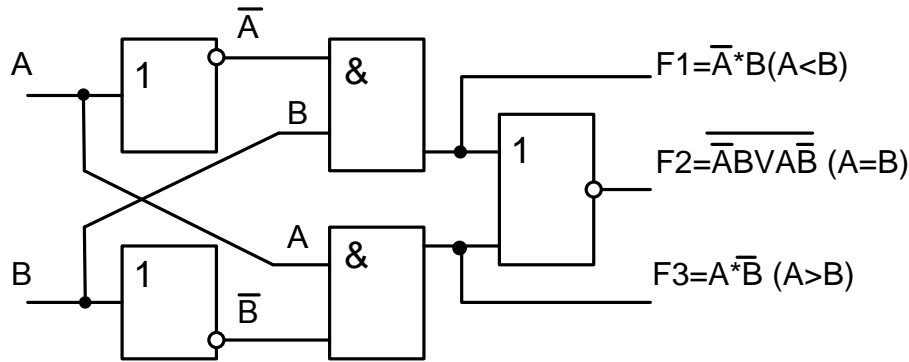


Рисунок 4.2 - Схема одноразрядного компаратора

Из анализа схемы следует, что если $A=B$, то $F2=1$. При $A>B$, $F3=1$, при $A<B$, $F1=1$. Если сравниваемые числа четырехразрядные, то сравнение происходит поразрядно.

Устройства сравнения на равенство строятся на основе поразрядных операций над одноименными разрядами обоих слов. Слова равны, если равны все одноименные их разряды, т. е. если в обоих нули или единицы. Признак равенства разрядов:

$$r_i = a_i b_i \vee \bar{a}_i \bar{b}_i = \overline{a_i \bar{b}_i \vee \bar{a}_i b_i} = \overline{a_i \bar{b}_i \cdot \bar{a}_i b_i} = \overline{a_i \oplus b_i}. \quad (4.1)$$

Признак неравенства разрядов:

$$\bar{r}_i = a_i \bar{b}_i \vee \bar{a}_i b_i = \overline{a_i \bar{b}_i \cdot \bar{a}_i b_i} = \overline{a_i \bar{b}_i \cdot \bar{a}_i b_i} = a_i \oplus b_i. \quad (4.2)$$

Признак равенства слов:

$$R = r_{n-1} r_{n-2} \dots r_0. \quad (4.3)$$

Схема двухразрядного компаратора на равенство в базисе И-НЕ показана на рисунке 4.3.

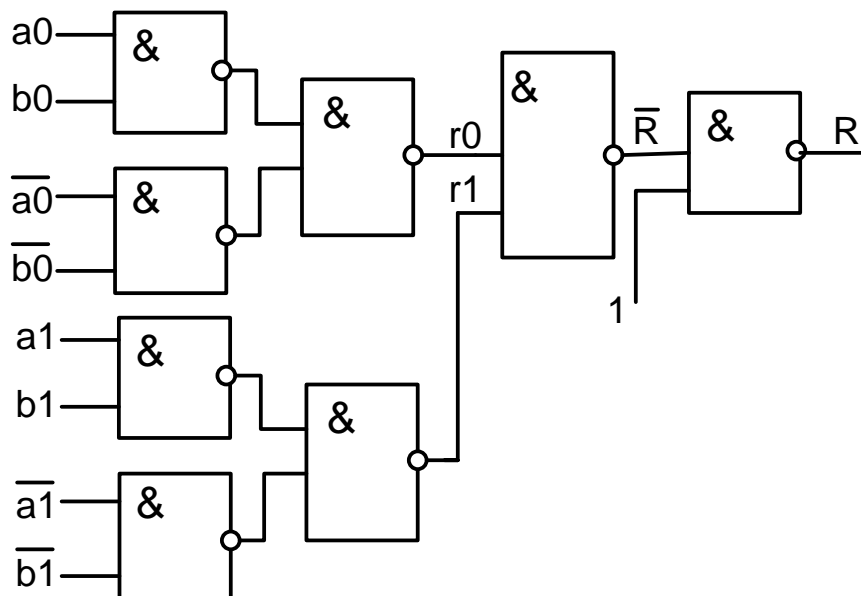


Рисунок 4.3 - Схема компаратора на равенство в базисе И-НЕ

Построение компаратора «на больше» для одноразрядных слов реализуется по выражению

$$F_{a>b} = a \cdot \bar{b} \quad (4.4)$$

Для n-разрядных слов выражение «на больше» следующее

$$F_{A>B} = a_{n-1} \cdot \bar{b}_{n-1} \vee r_{n-1} \cdot a_{n-2} \cdot \bar{b}_{n-2} \vee \dots \vee r_{n-1} \cdot r_{n-2} \dots r_1 \cdot a_0 \cdot \bar{b}_0 \quad (4.5)$$

Выражение для двухразрядного компаратора «на меньше»

$$F_{A<B} = \overline{\overline{A_1 B_1 A_0 B_1 B_0 A_1 A_0 B_0}} \quad (4.6)$$

Компараторы для слов большой разрядности получают путем наращивания с использованием нескольких интегральных микросхем.

4.3 Задание

1 Записать выражение в алгебре логики для выходной функции F по заданному варианту.

2 Записать таблицу истинности для заданной функции.

3 Построить принципиальную схему в заданном базисе.

4 Провести моделирование схемы в программной среде, используя светодиоды.

Таблица 4.1 – Варианты заданий

№ варианта	Задание
1	двухразрядный цифровой компаратор с тремя выходами в базисе И-НЕ
2	четырёхразрядный цифровой компаратор на равенство в базисе И-НЕ
3	трехразрядный цифровой компаратор на равенство на демультиплексоре и мультиплексоре
4	двухразрядный цифровой компаратор на равенство в базисе И-НЕ
5	четырёхразрядный цифровой компаратор с тремя выходами в базисе И-ИЛИ-НЕ
6	двухразрядный цифровой компаратор на меньше в базисе И-НЕ
7	трехразрядный цифровой компаратор на больше на демультиплексоре и мультиплексоре
8	четырёхразрядный цифровой компаратор на меньше в базисе И-ИЛИ-НЕ
9	двухразрядный цифровой компаратор с тремя выходами в базисе ИЛИ-НЕ
10	четырёхразрядный цифровой компаратор с тремя выходами в базисе И-НЕ
11	Трехразрядный цифровой компаратор с тремя выходами в базисе И-НЕ
12	двухразрядный цифровой компаратор на меньше в базисе ИЛИ-НЕ

Пример реализации схемы четырёхразрядного компаратора «на больше» в смешанном базисе приведен на рисунке 4.4.

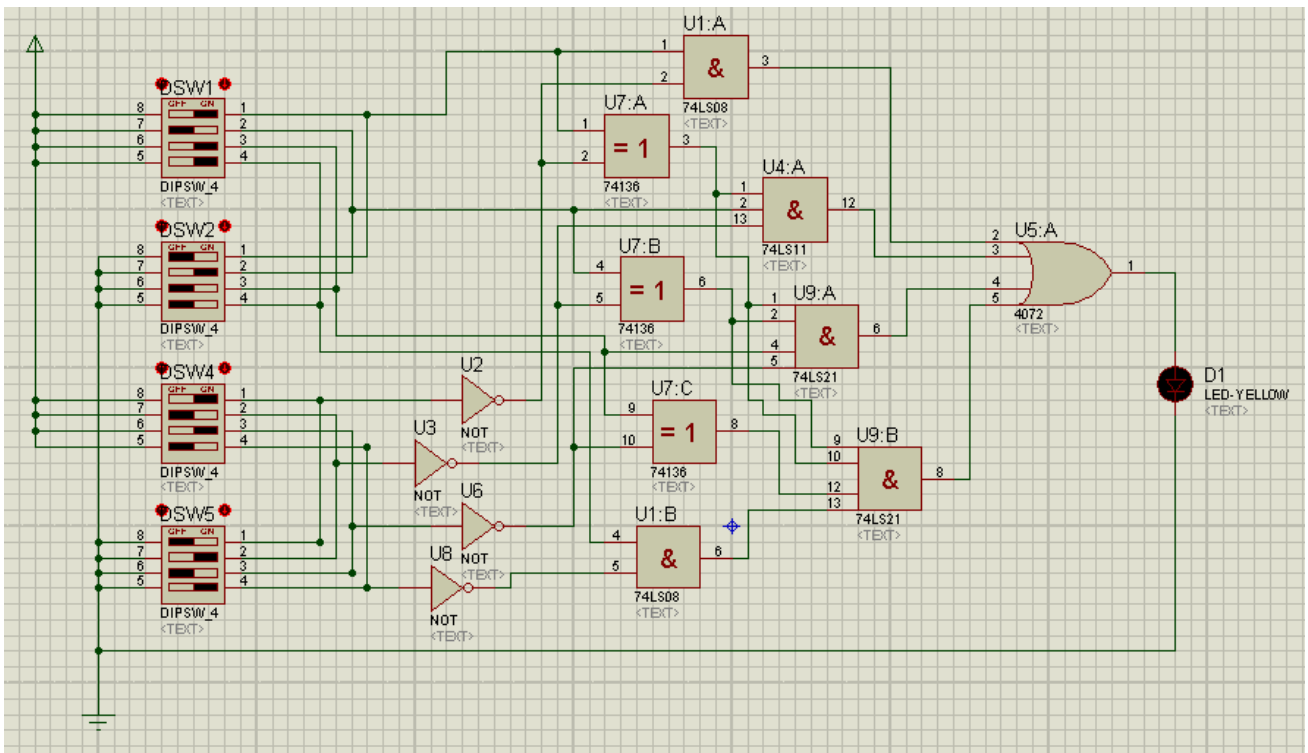


Рисунок 4.4 - Схема четырехразрядного компаратора «на больше»

4.4 Контрольные вопросы

- 1 Дать определение компаратора, привести условное графическое обозначение, пояснить принцип работы.
- 2 Записать выражение и привести схему реализации двухразрядного компаратора «на равенство» для базиса И-НЕ.
- 3 Записать выражение и привести схему реализации двухразрядного компаратора «на больше» в смешанном базисе.
- 4 Записать выражение и привести схему реализации одноразрядного компаратора «на меньше».
- 5 Привести схему наращивания разрядности компараторов, пояснить принцип ее работы
- 6 Привести схему реализации компаратора «на равенство» на демультиплексоре и мультиплексоре.
- 7 Записать выражение для n-разрядного компаратора на «больше».

5 Лабораторная работа № 5. Синтез сумматоров

5.1 Цель работы

- Изучить принцип работы сумматора.
- Освоить синтез сумматоров на логических схемах.

5.2 Теоретические сведения

Сумматоры выполняют арифметическое сложение и вычитание чисел. Сумматоры имеют самостоятельное значение, а также являются ядром арифметико-логических устройств (АЛУ), реализующих ряд разнообразных операций в составе процессоров.

Условное графическое обозначение двухразрядного сумматора показано на рисунке 5.1.

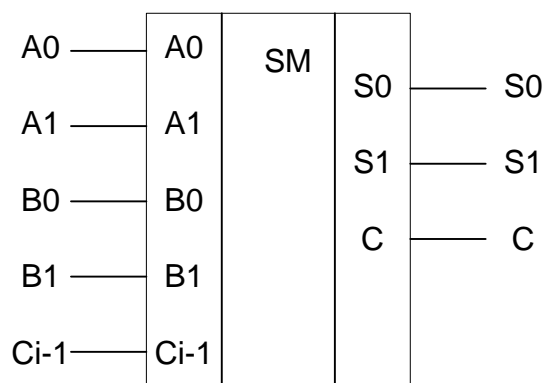


Рисунок 5.1 -Условное графическое обозначение двухразрядного сумматора

Аппаратная сложность и быстродействие сумматора являются очень важными параметрами, разработано множество вариантов сумматоров:

- одноразрядный сумматор;
- сумматор для последовательных операндов;
- сумматор для параллельных операндов с последовательным переносом;
- сумматор для параллельных операндов с параллельным переносом;
- сумматор групповой структуры с цепным переносом;

- сумматор с параллельным межгрупповым переносом;
- сумматор с условным переносом;
- накапливающий сумматор.

Наряду с сумматорами могут быть реализованы вычитатели, однако это почти никогда не делается, поскольку вычитание выполняется через сложение с применением дополнительных либо обратных кодов.

Одноразрядный сумматор. Одноразрядный сумматор имеет три входа (два слагаемых и перенос из предыдущего разряда) и два выхода (сумма). Таблица истинности одноразрядного сумматора показана в таблице 5.1.

Таблица 5.1 - Таблица истинности одноразрядного сумматора

a_i	b_i	c_{i-1}	S_i	C_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Аналитические выражения для функций суммы S и переноса C

$$S_i = \bar{a}_i \bar{b}_i c_{i-1} \vee \bar{a}_i b_i \bar{c}_{i-1} \vee a_i \bar{b}_i \bar{c}_{i-1} \vee a_i b_i c_{i-1}$$

$$C_i = a_i b_i \vee a_i c_{i-1} \vee b_i c_{i-1}$$

Можно записать эти выражения в базисе И-НЕ

$$S_i = \overline{\bar{a}_i \bar{b}_i c_{i-1} * \bar{a}_i b_i \bar{c}_{i-1} * a_i \bar{b}_i \bar{c}_{i-1} * a_i b_i c_{i-1}}$$

$$C_i = \overline{a_i b_i * a_i c_{i-1} * b_i c_{i-1}}$$

Полусумматор – это комбинационная логическая схема, имеющая два входа и два выхода (двухразрядный сумматор, бинарный сумматор). Полусумматор позволяет вычислять сумму $A+B$, где A и B — это разряды (биты) обычно двоичного числа, при этом результатом будут два бита S и C , где S — это бит суммы по модулю 2, а C — бит переноса. Существуют сумматоры и полусумматоры, работающие не в двоичной логике.

Отличается от полного сумматора тем, что не имеет входа переноса из предыдущего разряда. Для построения полного сумматора необходимо иметь дополнительный вход переноса из предыдущего разряда, таким образом, полный сумматор имеет 3 входа. Двоичный полный сумматор строится из двух полусумматоров и логического элемента 2ИЛИ, именно поэтому рассматриваемая схема называется полусумматором. Полусумматоры используются для построения полных сумматоров.

Последовательные сумматоры.

Последовательные сумматоры преобразуют последовательные коды слагаемых в последовательный код суммы этих слагаемых. Сложение начинается с младшего разряда и выполняется поразрядно последовательно за столько тактов, сколько разрядов содержится в числе. В состав многоразрядного сумматора последовательного действия, кроме комбинационного одноразрядного сумматора, применяют три сдвигающих регистра для двух слагаемых A и B и для результата S , триггер запоминания переноса и схемы управления вводом и выводом чисел. Регистры и триггер тактируются синхроимпульсами.

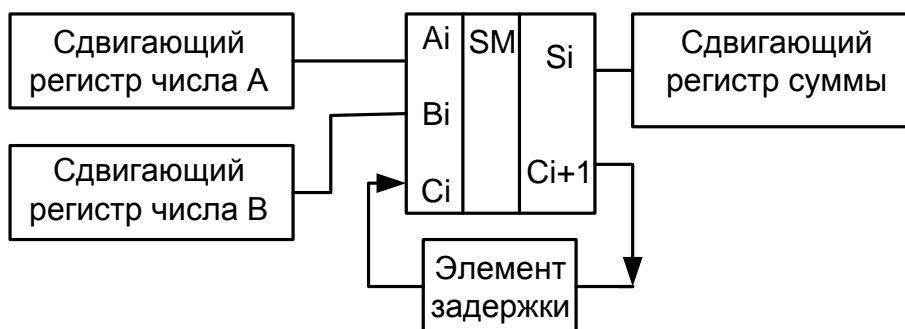


Рисунок 5.2 – Схема последовательного сумматора

Последовательный сумматор состоит из одноразрядного сумматора, на входы которого из сдвигающих регистров, хранящих слагаемые A и B , подаются по тактам разряд за разрядом коды этих чисел, начиная с младшего разряда. Если время элемента задержки равно одному такту (например, использован D -триггер), то сигнал переноса от предыдущего разряда, сформированный в предыдущем такте, поступит на нижний вход сумматора только в следующем такте, когда на входы A и B будут поданы значения следующего разряда слагаемых. В результате на выходе S разряд за разрядом будет формироваться двоичное число, равное сумме чисел A и B , которое будет восприниматься сдвигающим регистром суммы.

Параллельные сумматоры.

Недостатком последовательного сумматора является то, что выполнение операции сложения растягивается на множество тактов, которое тем больше, чем больше разрядность чисел. Значительно меньшее время выполнения операции имеет параллельный сумматор. В этом устройстве операция сложения производится одновременно во всех разрядах чисел A и B , поступающих в параллельном коде. Для этого схему составляют из n одноразрядных сумматоров, соединяя выход переноса i -го разряда со входом переноса соседнего $(i + 1)$ -го разряда. Схема показана на рисунке 5.3. Такой сумматор называют сумматором с последовательным переносом. После того как сформируется результат на выходах всех одноразрядных сумматоров, он запоминается в параллельном регистре.

Для того чтобы на выходах одноразрядных сумматоров сформировался результат, необходимо, чтобы на входах присутствовали все три сигнала – a_i , b_i , и c_i . Но сигнал переноса из разряда в разряд формируется предыдущими одноразрядными сумматорами, причем каждый последующий сумматор вынужден "ждать", пока не сформируется результат во всех предыдущих. В крайнем случае возможна ситуация, когда сигнал переноса будет последовательно передаваться через всю цепочку от самого младшего разряда до самого старшего, например, при сложении чисел $A = 11...11$ и $B = 00...01$. Поэтому время формирования результата в наихудшем случае включает в себя временные задержки, вносимые всеми одноразрядными сумматорами.

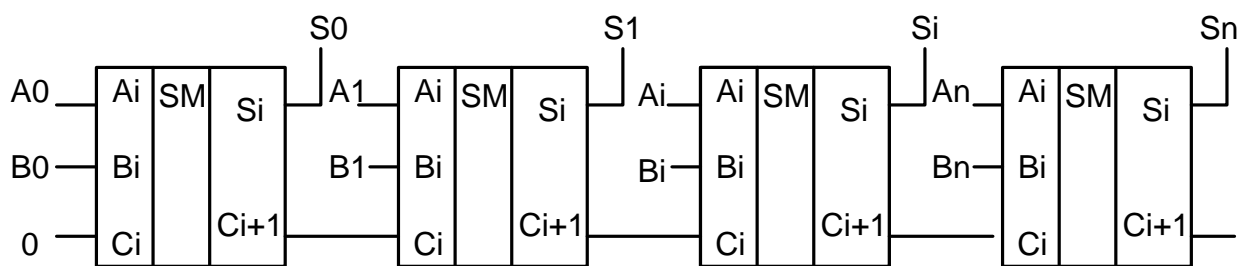


Рисунок 5.3 – Схема параллельного сумматора с последовательным переносом

5.3 Задание

- 1 Записать аналитическое выражение для выходной функции F в соответствии с заданным вариантом.
- 2 Построить таблицу истинности для выходной функции F в соответствии с заданным вариантом.
- 3 Построить электрическую принципиальную схему в заданном базисе.
- 4 Провести моделирование схемы в программной среде, используя сигнализирующие светодиоды.

Таблица 5.2 – Варианты заданий

№ варианта	Задание
1	Одноразрядный сумматор (на входе a и b ; выход сумма). Нельзя использовать XOR.
2	Одноразрядный полусумматор (на входе a и b ; выход сумма и перенос) в базисе И-ИЛИ-НЕ
3	Одноразрядный полный сумматор (на входе a , b и перенос; выход сумма и перенос) в базисе И-НЕ
4	4-разрядный параллельный сумматор. Предусмотреть вход переноса.
5	2-разрядный последовательный сумматор.

Продолжение таблицы 5.2

6	2-разрядный параллельный сумматор с параллельным переносом. Предусмотреть вход переноса
7	Параллельный сумматор с последовательным переносом для 2-х разрядных чисел
8	2-разрядный двоичный сумматор– вычитатель и выполнить следующие арифметические операции $A+B$ и $C-D$
9	2-разрядный полусумматор в базисе И-ИЛИ-НЕ
10	2-разрядный полусумматор в базисе И-НЕ
11	4- разрядный полусумматор в базисе И-ИЛИ-НЕ
12	4- разрядный полусумматор в базисе И-НЕ

5.4 Контрольные вопросы

1 Дать определение сумматора, привести условное графическое обозначение сумматора и пояснить принцип работы.

2 Привести схему и таблицу истинности 2-разрядного сумматора.

3 Дать определение полусумматора, привести условное графическое обозначение, пояснить принцип работы.

4 Дать определение и объяснить принцип работы сумматора для последовательных операндов. Привести схему, пояснить принцип работы.

5 Укажите достоинства и недостатки двоичных сумматоров с последовательным переносом.

6 Как оценивается быстродействие сумматоров?

7 Что такое параллельный сумматор, назовите его виды. Поясните принцип работы.

8 Что такое сумматор с групповой структурой? Привести схему и пояснить принцип работы.

6 Практическая работа № 6. Триггеры и их преобразования

6.1 Цель работы

Изучение типов, схем построения и принципов работы триггеров

6.2 Теоретические сведения

Триггеры - цифровое устройство последовательного типа, имеющее два устойчивых состояния и способное под действием внешних сигналов переключаться из одного состояния в другое. При этом напряжение на его выходе изменяется скачкообразно. Триггеры — элементарные автоматы, содержащие собственно элемент памяти (фиксатор) и схему управления. Фиксатор строится на двух инверторах, связанных друг с другом «накрест», так что выход одного соединен с входом другого. Такое соединение дает цепь с двумя устойчивыми состояниями. Действительно, если на выходе инвертора 1 имеется логический ноль, то он обеспечивает на выходе инвертора 2 логическую единицу, благодаря которой сам и существует. То же согласование сигналов имеет место и для второго состояния, когда инвертор 1 находится в единице, а инвертор 2 — в нуле. Любое из двух состояний может существовать неограниченно долго.

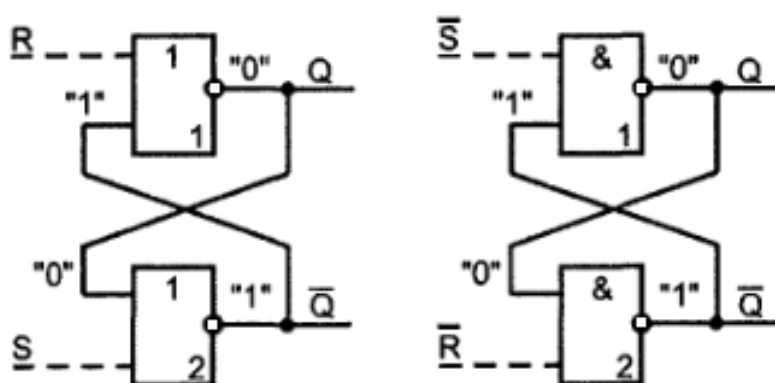


Рисунок 6.1 – Схема RS - триггера

Переходное состояние, в котором инверторы активны, неустойчиво. Это можно показать, имея в виду, что напряжения в любой цепи не являются идеально постоянными, а всегда имеют место флуктуации. Флуктуации обязательно при-

ведут фиксатор в одно из двух стабильных состояний, т. к. из-за наличия в схеме петли положительной обратной связи любое изменение режима вызывает продолжение в том же направлении, пока фиксатор не перейдет в устойчивое состояние, когда петля обратной связи как бы разрывается вследствие потери инверторами усилительных свойств (переход в режимы отсечки и насыщения, свойственные устойчивым состояниям).

Буквой R латинского алфавита (от Reset) обозначен сигнал установки триггера в ноль (сброса), а буквой S (от Set) — сигнал установки в состояние логической единицы (установки). Состояние триггера считывается по значению прямого выхода, обозначаемого как Q. Чаще всего триггер имеет и второй выход с инверсным сигналом \bar{Q} . Для фиксатора на элементах ИЛИ-НЕ установочным сигналом является единичный, поскольку только он приводит логический элемент в нулевое состояние независимо от сигналов на других входах элемента. Для фиксатора на элементах И-НЕ установочным сигналом является нулевой, как обладающий тем же свойством однозначно задавать состояние элемента независимо от состояний других входов. Практически все серии цифровых ИС содержат готовые триггеры, и поэтому задача проектировщика — правильное использование имеющихся триггеров. Отсюда важное значение приобретают классификация триггеров, изучение их параметров и особенностей функционирования.

Триггер типа RS имеет два входа — установки в единицу (S) и установки в ноль (R). Одновременная подача сигналов установки S и сброса R не допускается, эта комбинация сигналов называется запрещенной. Условное графическое обозначение синхронного RS-триггера представлено на рисунке 6.2.

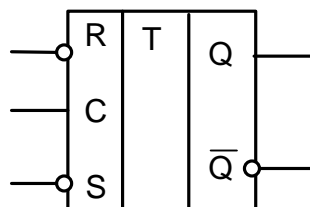


Рисунок 6.2 - Условное графическое обозначение синхронного RS-триггера

Триггер типа D (от слова Delay — задержка) имеет один вход. Его состояние повторяет входной сигнал, но с задержкой, определяемой тактовым сигналом.

Триггер типа T изменяет свое состояние каждый раз при поступлении входного сигнала. Имеет один вход, называется триггером со счетным входом или счетным триггером.

Триггер типа JK универсален, имеет входы установки (J) и сброса (K), подобные входам триггера RS. В отличие от последнего, допускает ситуацию с одновременной подачей сигналов на оба эти входа ($J = K = 1$). В этом режиме работает как счетный триггер относительно третьего (тактового) входа. В комбинированных триггерах совмещаются несколько режимов. Например, триггер типа RST — счетный триггер, имеющий также входы установки и сброса.

По способу записи информации различают асинхронные (нетактируемые) и синхронные (тактируемые) триггеры. В нетактируемых переход в новое состояние вызывается непосредственно изменениями входных информационных сигналов. В тактируемых, имеющих специальный вход C, переход происходит только при подаче на этот вход тактовых сигналов. Тактовые сигналы называют также синхронизирующими, исполнительными, командными и т. д.

По способу восприятия тактовых сигналов триггеры делятся на управляемые уровнем и управляемые фронтом. Управление уровнем означает, что при одном уровне тактового сигнала триггер воспринимает входные сигналы и реагирует на них, а при другом не воспринимает и остается в неизменном состоянии. При управлении фронтом разрешение на переключение дается только в момент перепада тактового сигнала. В остальное время независимо от уровня тактового сигнала триггер не воспринимает входные сигналы и остается в неизменном состоянии. Триггеры, управляемые фронтом, называют также триггерами с динамическим управлением. Динамический вход может быть прямым или инверсным. Прямое динамическое управление означает разрешение на переключение при изменении тактового сигнала с нулевого значения на единичное, инверсное — при изменении тактового сигнала с единичного значения на нулевое.

По характеру процесса переключения триггеры делятся на одноступенчатые и двухступенчатые. Двухступенчатые триггеры состоят из входной и выходной ступеней. Переход в новое состояние происходит в обеих ступенях поочередно. Один из уровней тактового сигнала разрешает прием информации во входную ступень при неизменном состоянии выходной ступени. Другой уровень тактового сигнала разрешает передачу нового состояния из входной ступени в выходную.

Взаимные преобразования триггеров.

JK-триггер преобразуется в динамический D-триггер подключением инвертора ко входу К. Схема представлена на рисунке 6.3.

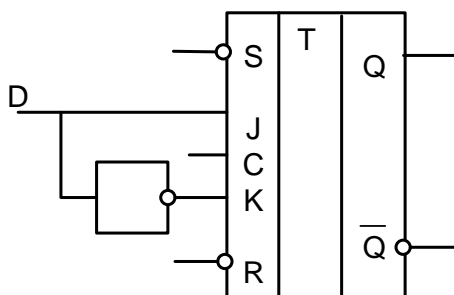


Рисунок 6.3 – Схема преобразования JK-триггер в D-триггер

При этом из четырех комбинаций сигналов: $J=K=0$, $J=K=1$, $J=0$ $K=1$, $J=1$ $K=0$ осуществлены будут две последних, т.е. синхронные установка и сброс. Если необходим прямой синхровход, ко входу С подключается еще один инвертор. На рисунке 6.4 показана схема преобразования JK-триггера Т – триггер со счетным входом ($J=K=1$).

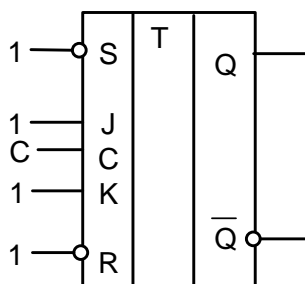


Рисунок 6.4 – Схема преобразования JK-триггера в Т-триггер

D-триггер с динамическим управлением также преобразуется в Т-триггер, путем введения обратной связи с инверсного выхода на вход D. Схема преобразования приведена на рисунке 6.5. Тогда $Q(t+dt) = D$, но D в свою очередь равно $D =$

$\sim Q_t$ и, следовательно $Q(t+dt) = \sim Q_t$, т.е. новое значение на выходе триггера является инверсией старого состояния с каждым поступлением положительного перепада тактового импульса C .

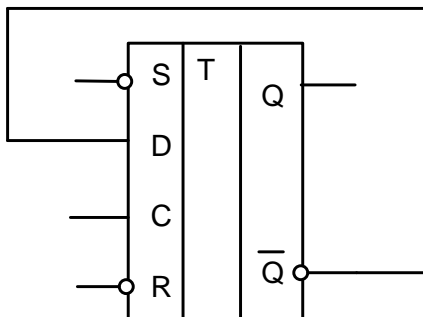


Рисунок 6.5 – Схема преобразования D-триггера в T-триггер

Любой из перечисленных триггеров может быть использован в качестве асинхронного RS-триггера с инверсными входами, невзирая на остальные сигналы, что объясняется наивысшим приоритетом входов $\sim S$ и $\sim R$.

6.3 Задание

6.3.1 Построить таблицу истинности для триггера в соответствии с заданным вариантом в таблице 6.1.

6.3.2 Записать логическое выражение синтеза триггера в заданном базисе.

6.3.3 Построить электрическую схему в заданном базисе.

6.3.4 Построить схему в программной среде и провести моделирование.

Таблица 6.1 – Варианты задания

№ вар.	Задание
1	Синтезировать RS- триггер в базисе И-НЕ
2	Синтезировать D- триггер в базисе ИЛИ-НЕ.
3	Синтезировать T- триггер в базисе И-НЕ

Продолжение таблицы 6.1

4	Синтезировать синхронный JK- триггер и в базисе ИЛИ-НЕ.
5	Синтезировать RS- триггер в базисе ИЛИ-НЕ.
6	Синтезировать D- триггер в базисе И-НЕ.
7	Синтезировать T- триггер в базисе ИЛИ-НЕ
8	Синтезировать синхронный JK- триггер в базисе И-НЕ.
9	Синтезировать двухступенчатый RS- триггер в базисе И-НЕ
10	Синтезировать на логических элементах динамический D-триггер
11	Преобразовать JK- триггер в динамический D - триггер
12	Преобразовать JK- триггер в RS- триггер
13	Преобразовать JK- триггер в D- триггер
14	Преобразовать D- триггер в T -триггер
15	Преобразовать JK - триггер в T- триггер

6.4 Контрольные вопросы

1 Дать определение триггера. Привести условное графическое обозначение, пояснить назначение выводов.

2 Классификация триггеров, охарактеризовать все классы триггеров.

3 Пояснить принцип работы RS- и JK-триггеров с помощью таблиц истинности и временных диаграмм.

4 Пояснить принцип работы D- и T-триггеров с помощью таблиц истинности и временных диаграмм.

5 Привести схемы взаимного преобразования триггеров, пояснить принцип работы.

6 Дать характеристику области применения триггеров в цифровой электронной технике.

7 Лабораторная работа № 7. Синтез регистров

7.1 Цель работы

- Освоить навыки построения функциональных схем регистров.
- Изучить принцип работы регистров.

7.2 Теоретические сведения

Регистр – последовательностное устройство, осуществляющее запись, хранение, сдвиг и выдачу чисел в виде многоразрядного двоичного кода (от англ. Register – регистрировать, фиксировать)

Запоминающим элементом в регистре служат триггеры. Число триггеров равно числу разрядов хранимых чисел.

Схема регистра служит для ввода/вывода хранимых чисел, преобразования их кодов, сдвига кодов на определённое число разрядов.

В регистрах выполняются следующие операции:

- ввод и вывод информации;
- хранение информации;
- сдвиг вправо или влево на определенное число разрядов;
- преобразование кода числа из последовательного в параллельный.

По способу приема информации регистры подразделяются:

— параллельные, информация записывается и считывается только в параллельной форме (статические – построены на триггерах и могут хранить информацию долго при наличии $U_{пит}$);

— последовательные (сдвиговые) – запись и считывание происходит только в последовательной форме;

— последовательно-параллельные, используются для преобразования кода из параллельного в последовательный и наоборот.

По числу каналов передачи информации:

- однофазного типа;

— парафазного типа.

В однофазных ввод (вывод) может производиться только в прямом или только в обратном коде.

В парафазных возможен ввод и вывод как в прямом так и в обратном кодах. Вид осуществляемого ввода (вывода) определяется сигналами управления.

По способу тактирования:

— одноктактные, синхронизируемые одной управляющей последовательностью;

— двухтактные;

— многотактные, управляемые несколькими последовательностями импульсов.

Регистры с параллельной записью называются, также регистрами памяти. В них могут использоваться, как прозрачные "защелки", так и триггеры с динамическим управлением. На рисунке 7.1 приведена схема 8-ми разрядного регистра памяти с общим входом управления записью информацией и ее условное обозначение. Высокий уровень на входе С переписывает информацию с входа на выход ($Q_i=D_i$), а низкий уровень - защелкивает данные.

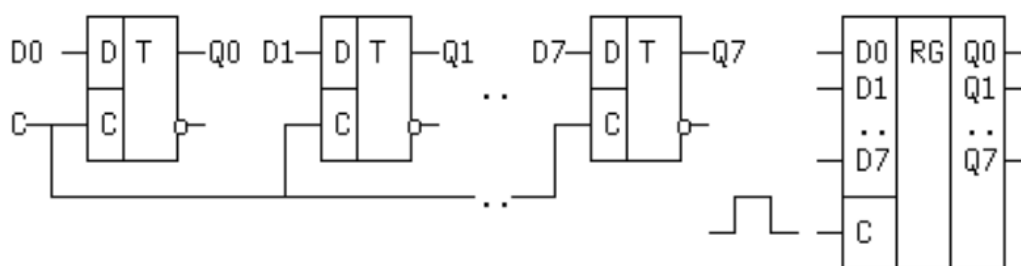


Рисунок 7.1 - Схема 8-ми разрядного регистра памяти с общим входом управления записью

Последовательные регистры или, как их еще называют, регистры сдвига выполняются на основе триггеров с динамическим синхривходом (справедливо для регистров с одним тактирующим сигналом). В двухтактных можно использовать и прозрачные регистры - "защелки". Функция записи в n-разрядном регистре сдвига на D-триггерах задается в виде условий:

$$D_0 = DS = x, D_i = Q_{(i-1)},$$

где $i=1,2,\dots,n-1$. DS - вход для последовательной записи. Схема последовательного регистра приведена на рисунке 7.2.

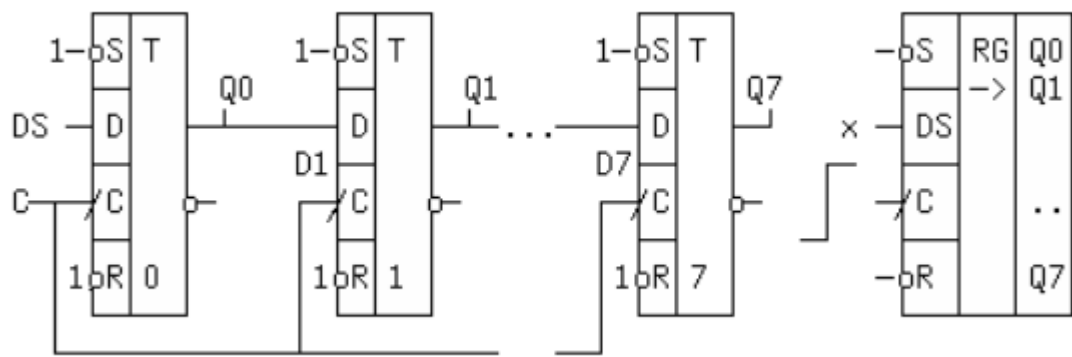


Рисунок 7.2 - Схема последовательного регистра

С приходом очередного положительного фронта синхроимпульса C , сигнал с входа i -го триггера через время задержки распространения сигнала $t_{зд.р.}$ окажется на его выходе и поступит на вход следующего $(i+1)$ -го триггера. Однако на его выход эта информация не переписется, т.к. длительность активного фронта $t_{0,1}$ меньше $t_{зд.р.}$. На этом процесс сдвига данных на один разряд закончится до прихода следующего положительного фронта тактового сигнала. Отсюда понятно, почему нельзя использовать триггеры со статическим управлением. Каждый раз при $C = 1$ вся цепочка окажется прозрачной от входа DS до выхода Q7 и значение $DS = x$ будет записано во все триггеры.

В обозначениях регистров сдвига направление стрелки, указывающей сдвиг, условно. В разных справочниках, ее направление различно. Условно принимается, что сдвиг производится от младшего разряда к старшему. Практические схемы регистров дополняются схемами, подключаемыми к каждому триггеру и имеющими вход параллельной записи D_i , общий вход разрешения записи L и общий асинхронный вход сброса $\sim R$ всех триггеров. Эти схемы подключаются к незадействованным входам $\sim R_i, \sim S_i$ триггеров. Данному описанию соответствует таблица истинности, показанная на рисунке 7.3.

Входы			Выходы		Название режима
L	Di	~R	~Si	~Ri	
X	X	0	1	0	Установка в "0" триггеров регистра
1	Di	1	~Di	Di	Параллельная запись инф-ии в триггеры
0	X	1	1	1	Режим хранения или сдвига

Рисунок 7.3 – Таблица истинности регистра сдвига

Di и X - могут принимать любые значения, но Di в пределах одной строки, неизменно. Минимизируя логические функции $\sim Ri$ и $\sim Si$ с помощью таблиц Карно, получим:

$$\sim Si = \sim(L * Di * \sim R) \text{ и } \sim Ri = \sim R * (\sim L + Di)$$

Этой паре уравнений соответствует схема синтеза регистра сдвига, показанная на рисунке 7.4, где приведено также условное обозначение регистра сдвига с параллельной записью.

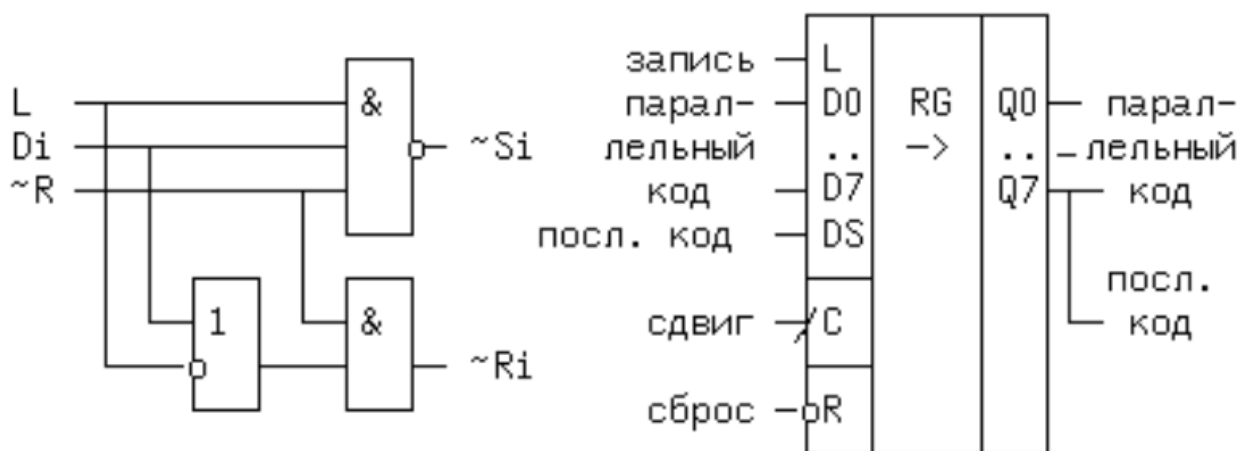


Рисунок 7.4 – Схема синтеза регистра сдвига и условное обозначение регистра сдвига с параллельной записью

Вход $\sim R$ обладает наивысшим приоритетом, если $\sim R=0$, то $\sim Ri=0$, а $\sim Si=1$ и все триггеры обнуляются, независимо от сигналов L, Di и C. Меньшим приоритетом обладают входы L и Di. Если $\sim R = 1$, то при $L = 1$, производится параллельная запись информации и $Qi = Di$ независимо от сигнала C. Если на входах $\sim R$ и L

пассивные уровни, то $\sim R_i = \sim S_i = 1$, тоже пассивный уровень и регистр хранит информацию, либо производит ее сдвиг.

Одно из применений регистров сдвига с параллельной загрузкой кода заключается в преобразовании параллельного формата данных в последовательный, передаче этих данных по однопроводной линии связи (вторая линия должна быть, как минимум "землей" или экраном) и обратном преобразовании последовательной информации в параллельную. Схема регистров сдвига с параллельной загрузкой представлена на рисунке 7.5.

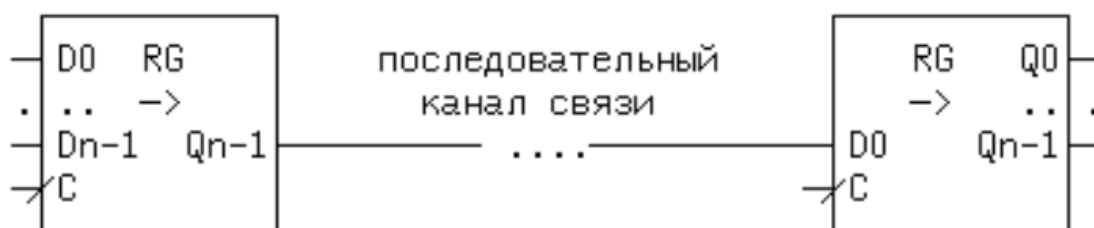


Рисунок 7.5 – Схема регистров сдвига с параллельной загрузкой

Реверсивный регистр сдвига. Название указывает, что сдвиг данных от разряда к разряду может производиться, как в одну сторону, так и в другую. Схема представлена на рисунке 7.6.

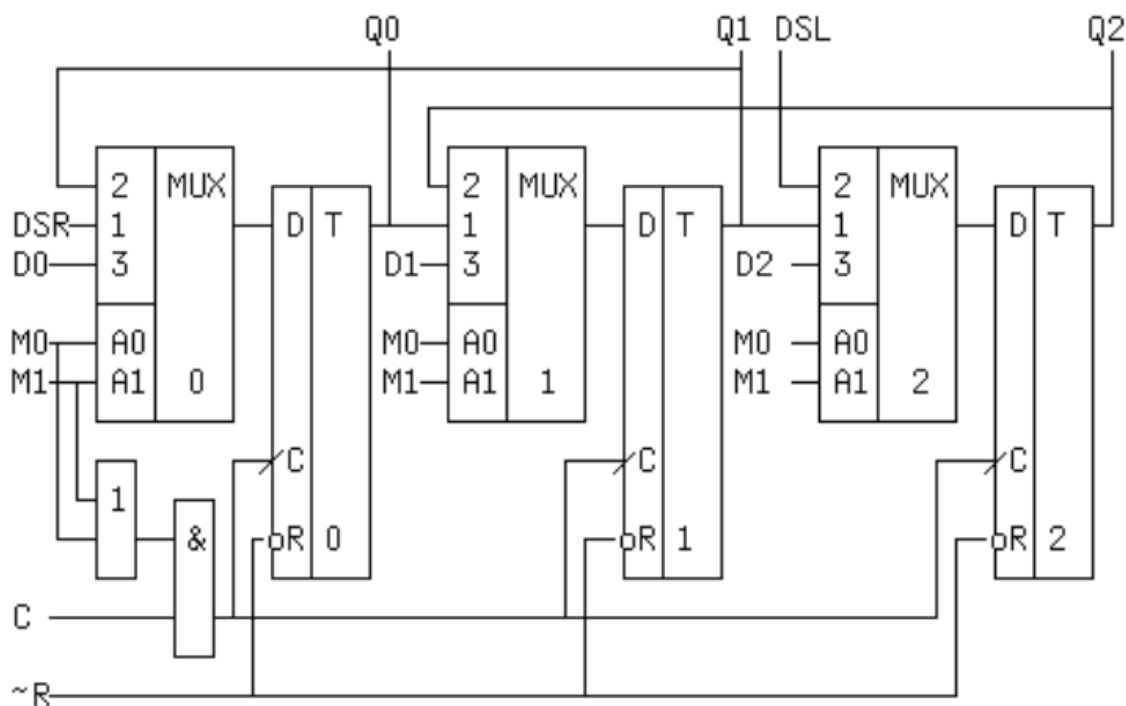


Рисунок 7.6 – Схема реверсивного регистра сдвига

Это схема трехразрядного реверсивного регистра с двумя последовательными информационными входами, для сдвига информации влево DSL, и вправо DSR, с параллельными входами записи (D0..D2), синхровходом С, входом $\sim R$ установки в 0 всех триггеров и двумя входами выбора режима M1, M0.

К D-входу любого разряда, за исключением крайних, подключены, через входы 1 и 2 мультиплексора, выходы и левого и правого соседних триггеров. Если M1=0, а M0=1, то к входам D подключены первые входы мультиплексоров и информация в каждый триггер, кроме нулевого записывается от левого соседа (происходит сдвиг вправо). Вход DSR служит в этом режиме для последовательного ввода информации. Если M1=1, M0=0, то к входам D подключены вторые входы мультиплексоров и информация в каждый триггер, кроме последнего записывается от правого соседа (происходит сдвиг влево). Для последовательного ввода данных в этом режиме используется вход DSL. При M1=M0=1 происходит параллельная запись $Q_i=D_i$ положительным фронтом тактового сигнала. Для хранения информации необходимо подать комбинацию M1=M0=0. В этом случае к D-входам триггеров подключаются нулевые входы мультиплексоров (на схеме не показаны). Чтобы не произошло несанкционированной записи схема ИЛИ-И запрещает в этом режиме прохождение синхроимпульсов. По такой схеме выполнен 8-разрядный регистр 1533ИР13. Реверсивный регистр может использоваться для быстрого (всего за n тактов) деления и умножения двоичных чисел на 2^n , где n - число сдвигов.

Синхронный и асинхронный способы загрузки параллельного кода.

Устройства (в том числе регистры), в которых для записи входного параллельного кода D_i используется сигнал разрешения записи L, а тактовый сигнал С не используется, называются *устройствами с асинхронной параллельной записью кода*. Регистры, в которых для записи входного параллельного кода D_i необходим, во-первых, сигнал разрешения записи L и, во-вторых, перепад синхросигнала на тактовом входе С - называются *устройствами с синхронной параллельной записью кода*.

7.3 Задание

7.3.1 Построить таблицу истинности для регистра в соответствии с заданным вариантом в таблице 6.1.

7.3.2 Записать логическое выражение синтеза регистра.

7.3.3 Построить электрическую схему в заданном базисе.

7.3.4 Построить схему в программной среде и провести моделирование.

Таблица 1.1 – Варианты задания

№ вар.	Задание
1	Синтезировать 4-разрядный регистр сдвига на D-триггерах
2	Синтезировать реверсивный 8-разрядный регистр сдвига на D триггерах
3	Синтезировать последовательный 8-разрядный регистр
4	Синтезировать 4- разрядный регистр памяти с общим входом управления записью
5	Синтезировать последовательный 4-разрядный регистр
6	Синтезировать параллельно- последовательный 4-разрядный регистр
7	Синтезировать параллельный 8-разрядный регистр на D-триггерах
8	Синтезировать 8-разрядный регистр с синхронной параллельной записью
9	Синтезировать 8-разрядный устройство с асинхронной параллельной записью кода
10	Синтезировать 8- разрядный регистр памяти с общим входом управления записью
11	Синтезировать 4-разрядный регистр с синхронной параллельной записью кода
12	Синтезировать 4-разрядный регистр с асинхронной параллельной записью кода

Продолжение таблицы 7.1

13	Синтезировать реверсивный 4-разрядный регистр сдвига на D триггерах
14	Синтезировать параллельно- последовательный 4-разрядный регистр на JK триггерах
15	Синтезировать последовательный 4-разрядный регистр на RS триггерах
16	Синтезировать параллельный 8-разрядный регистр

7.4 Контрольные вопросы

1 Дать определение регистра, указать виды регистров. Привести условное графическое обозначение.

2 Дать характеристику классификации регистров по различным признакам.

3 Параллельный регистр, условное графическое обозначение, принцип работы, разновидности.

4 Последовательный регистр, условное графическое обозначение, принцип работы, виды.

5 Схема синтеза регистра сдвига и условное обозначение регистра сдвига с параллельной записью, принцип работы.

6 Синхронный и асинхронный способы загрузки параллельного кода.

7 Реверсивный регистр сдвига. Условное графическое обозначение, принцип работы.

8 Синтез параллельного 8-разрядного регистра на D-триггерах. Привести схему, пояснить принцип работы.

9 Синтез параллельно- последовательный 4-разрядного регистра. Привести схему, пояснить принцип работы.

10 Области применения различных видов регистров.

8 Практическая работа № 8. Синтез счетчиков

8.1 Цель работы

- Изучить принцип работы счетчика.
- Освоить синтез счетчиков на логических схемах.

8.2 Теоретические сведения

Счетчиком называется последовательностное устройство, предназначенное для счета входных импульсов и фиксации их числа. Счетчики – это цифровые автоматы, внутренние состояния которых определяются только количеством сигналов «1», пришедших на вход. Сигналы «0» не изменяют их внутренние состояния.

Триггер Т-типа является простейшим счетчиком, который считает до двух. Счетчик, образованный цепочкой из m триггеров, сможет подсчитывать в двоичном коде 2^m входных импульсов. Каждый из триггеров в этой цепочке называют разрядом счетчика. Основное функциональное назначение счетчиков:

- счет импульсов, поступивших на вход;
- деление частоты.

В счетчиках выполняются следующие операции:

- установка в нулевое состояние (сброс);
- запись входной информации в параллельной форме – начального кода, с которого начинается счет;
- хранение записанной информации;
- выдача хранимой информации в параллельной форме;
- инкремент – увеличение хранящегося числа на единицу;
- декремент – уменьшение хранящегося числа на единицу.

Основная характеристика счетчика – модуль счета, или емкость счетчика $K_{сч}$. Это количество поступивших входных сигналов, которое возвращает счетчик в исходное состояние. Количество триггеров, необходимое для реализации счетчика, равно $m = \log_2 K_{сч}$, где m – ближайшее большее целое число.

Классификация счетчиков. Цифровые счетчики классифицируются следующим образом:

— по модулю счета: двоичные, двоично-десятичные или с другим основанием счета, недвоичные с постоянным модулем счета, с переменным модулем счета;

— по направлению счета: суммирующие, вычитающие, реверсивные;

— по способу организации внутренних связей: с последовательным переносом, с параллельным переносом, с комбинированным переносом, кольцевые.

Классификационные признаки независимы и могут встречаться в различных сочетаниях: например, суммирующие счетчики бывают как с последовательным, так и с параллельным переносом и могут иметь двоичный, десятичный и иной модуль счета.

В суммирующем счетчике каждый входной импульс увеличивает число, записанное в счетчик, на единицу (для счетчиков с естественным порядком счета) и на единицу и более для счетчиков с произвольным порядком счета.

Вычитающий счетчик действует обратным образом: двоичное число, хранящееся в счетчике, с каждым поступающим импульсом уменьшается. Переполнение счетчика наступает при поступлении на его вход количества импульсов большего Ксч.

Реверсивный счетчик может работать в качестве суммирующего и вычитающего. Эти счетчики имеют дополнительные входы для задания направления счета.

По способу кодирования различают счетчики Джонсона, счетчики с кодом «1 из N», счетчики в коде Грея и т.д. Наиболее распространены двоичные счетчики, все другие могут быть получены из них путем установления дополнительных связей между разрядами.

По способу переключения триггеров во время счета счетчики делятся на *асинхронные* и *синхронные*. Первые называются еще счетчиками с последовательным переносом, т.к. переход каждого триггера из одного состояния в противоположное происходит последовательно во времени. Входной переключающий сиг-

нал непосредственно воздействует лишь на первый триггер, и каждый триггер вырабатывает переключающий сигнал для следующего соседнего триггера.

Последовательные счетчики. Рассмотрим работу суммирующего двоичного счетчика ($K \text{ сч.} = 2^m$) с естественным порядком счета и с $K \text{ сч.} = 8$. Для его построения необходимо $m = \log_2 8 = 3$ триггера, что соответствует трем разрядам двоичного числа.

Таблица состояний такого счетчика имеет вид, показанный на рисунке 8.1. Причем входной сигнал x^n обозначим через 1, Q_3^n – старший разряд, Q_1^n – младший разряд.

x^n	Q_3^n	Q_2^n	Q_1^n	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}
1	0	0	0	0	0	1
1	0	0	1	0	1	0
1	0	1	0	0	1	1
1	0	1	1	1	0	0
1	1	0	0	1	0	1
1	1	0	1	1	1	0
1	1	1	0	1	1	1
1	1	1	1	0	0	0

Рисунок 8.1 - Таблица состояний счетчика

Из анализа таблицы видно:

— триггер младшего разряда Q_1 переключается от каждого входного сигнала;

— второй разряд Q_2 переключается через два входных сигнала;

— третий разряд Q_3 переключается через четыре входных сигнала.

Таким образом, частота переключения каждого следующего триггера уменьшается вдвое. Следовательно, последовательный счетчик можно построить как цепочку последовательно включенных счетных триггеров.

Схема последовательного счетчика на JK-триггерах, работающих в счетном режиме, показана на рисунке 8.2.

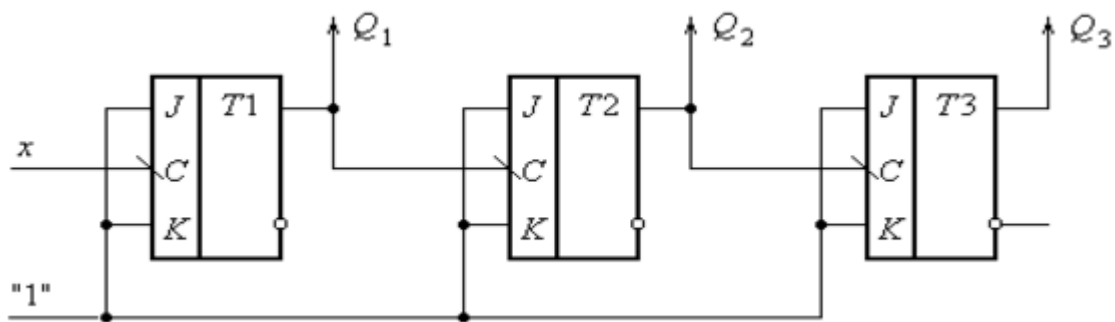


Рисунок 8.2 - Схема последовательного счетчика на JK-триггерах

Данный счетчик может работать как вычитающий. Для этого необходимо сигналы на входы последующих разрядов подавать с инверсных выходов триггеров предыдущих разрядов. Так как полученный счетчик – асинхронный, то каждый его триггер переключается с задержкой относительно входного сигнала. Поэтому по мере продвижения сигнала от младшего разряда к старшему эта задержка суммируется и может произойти искажение информации, в виде несоответствия числа уже поступивших в счетчик импульсов и кода на его выходах. В общем случае суммарная задержка пропорциональна числу триггеров и для устранения ее влияния на работоспособность счетчика приходится снижать частоту поступления входных импульсов, что снижает, в целом, быстродействие счетчика. Временная диаграмма работы последовательного счетчика приведена на рисунке 8.3.

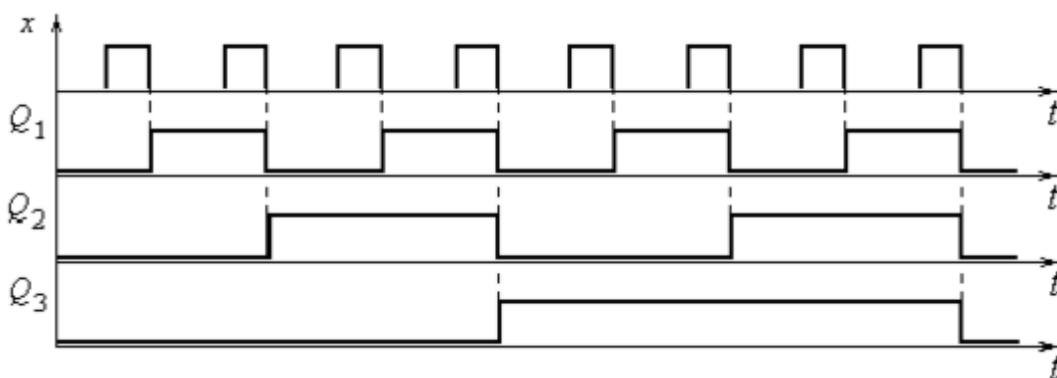


Рисунок 8.3 - Временная диаграмма работы последовательного счетчика

Счетчики с параллельным переносом. Для повышения быстродействия счетчики выполняются с параллельным (сквозным) переносом. Их особенность заключается в том, что выходы всех предшествующих разрядов счетчика соединяются с входами триггера последующего разряда, поэтому длительность переходного процесса определяется только длительностью переходного процесса одного разряда и не зависит от количества триггеров. Отсюда следует, что параллельные счетчики – синхронные. В результате получаются следующие функции входов триггеров счетчика

$$\begin{aligned}
 J_1 &= 1 & K_1 &= 1 \\
 J_2 &= Q_1 & K_2 &= Q_1 \\
 J_3 &= Q_1 Q_2 & K_3 &= Q_1 Q_2
 \end{aligned}$$

Схема параллельного суммирующего счетчика приведена на рисунке 8.4.

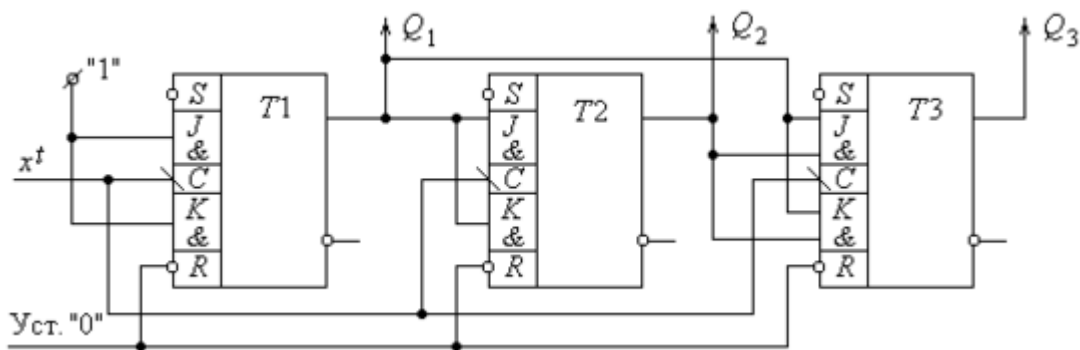


Рисунок 8.4 – Схема параллельного суммирующего счетчика

Реверсивный счетчик. Такой счетчик должен, в зависимости от сигналов управления, обеспечивать или режим суммирования, или режим вычитания входных сигналов. Из сравнения функций входов, полученных ранее для суммирующего и вычитающего параллельных счетчиков с К сч. = 8, следует, что сами функции имеют один и тот же вид, только в случае вычитающего счетчика берутся инверсные значения переменных. Следовательно, реверсивный счетчик должен содержать схему управления, обеспечивающую подключение либо прямых, либо инверсных выходов ко входам последующих разрядов, в зависимости от сигналов

управления направлением счета T. Функция входов для реверсивного счетчика будет иметь вид:

$$K_1 = J_1 = 1,$$

$$J_2 = K_2 = TQ_1 \vee \overline{TQ_1},$$

$$J_3 = K_3 = TQ_1 Q_2 \vee \overline{TQ_1} \overline{Q_2},$$

Схема реверсивного двоичного параллельного счетчика с Ксч = 8 показана на рисунке 8.5.

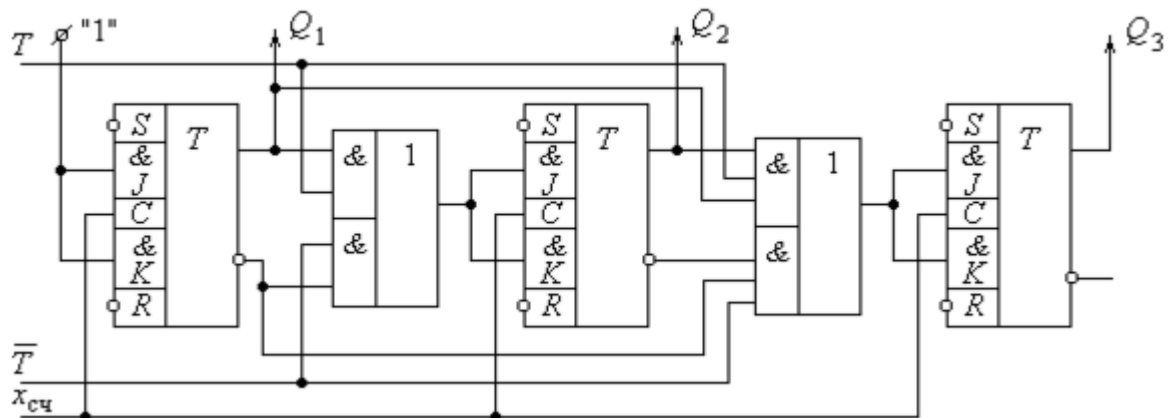


Рисунок 8.5 - Схема реверсивного параллельного счетчика с Ксч = 8

Условное графическое обозначение реверсивного счетчика приведено на рисунке 8.6.

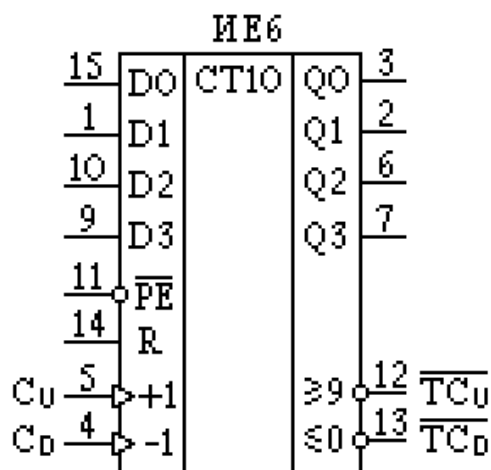


Рисунок 8.6 - Условное графическое обозначение реверсивного счетчика

8.3 Задание

1. Записать таблицу истинности (переходов) счетчика в соответствии с заданным вариантом.
2. Составить карты переходов триггеров каждого разряда. Карта переходов размечается также как карта Карно, строится по таблице состояний и отображает переход триггера из предыдущего состояния в последующее.
3. Разработать электрическую принципиальную схему на заданном виде триггеров, используя светодиоды.
4. Провести моделирование работы счетчика в программной среде.

Таблица 8.1 – Варианты заданий

№ вар.	Задание
1	Синтезировать 4-разрядный суммирующий счетчик на JK-триггерах
2	Синтезировать 4-разрядный вычитающий счетчик на JK-триггерах
3	Синтезировать 3-разрядный синхронный суммирующий счетчик на D-триггерах
4	Из 3-разрядных синхронных суммирующих счетчиков, построенных на D-триггерах, получить 9-разрядный счетчик
5	Синтезировать 3-разрядный счетчик на триггерах с изменяемым направлением счета
6	Синтезировать на триггерах 3-разрядный счетчик с предварительной установкой и сбросом
7	Синтезировать 4-разрядный счетчик, который при одном состоянии управляющего сигнала считает от 0 до 9, а при другом по коду Грея
8	Синтезировать 4-разрядный счетчик на триггерах с изменяемым направлением счета
9	Синтезировать двоично-десятичный суммирующий счетчик, работающий в прямом коде

Продолжение таблицы 8.1

10	Синтезировать на Т-триггерах счетчик с коэффициентом пересчета равным 5
11	Синтезировать вычитающий счетчик с коэффициентом пересчета 8
12	Синтезировать реверсивный счетчик с коэффициентом пересчета, равным 8
13	Синтезировать суммирующий последовательный 3-х разрядный счётчик
14	Синтезировать параллельный суммирующий 3-х разрядный счётчик
15	Синтезировать реверсивный последовательный 3-х разрядный счётчик
16	Синтезировать вычитающий последовательный 3-х разрядный счётчик

8.4 Контрольные вопросы

- 1 Дать определение счетчика. Привести его условное графическое изображение, указать основные функции.
- 2 Описать классификацию счетчиков, пояснить назначение каждого класса.
- 3 Пояснить работу схемы последовательного суммирующего двоичного счетчика.
- 4 Пояснить работу схемы параллельного суммирующего счетчика.
- 5 Дать определение и пояснить принцип работы реверсивного счетчика.
- 6 Что такое таблица состояний счетчика? Пояснить на примере последовательного счетчика.
- 7 Пояснить принцип синтеза счетчиков на D-триггерах.
- 8 Привести области применения счетчиков.
- 9 Пояснить принцип работы счетчика с изменяемым направлением счета.
- 10 Пояснить различия синхронных и асинхронных счетчиков на примере.
- 11 Дать характеристику счетчиков с разными методами кодирования.

Список использованных источников

- 1 Бурькова Е.В. Аналоговая и импульсная электроника: учебное пособие / Е.В. Бурькова. – Оренбург, ГОУ ОГУ, 2007. – 174 с.
- 2 Гусев, В. Г. Электроника и микропроцессорная техника: учеб. для вузов / В. Г. Гусев, Ю. М. Гусев .- 5-е изд., стер. - М: Высш. шк., 2008. - 798 с
- 3 Лачин, В.И. Электроника: учебное пособие / В.И. Лачин, Н.С. Савелов. - Ростов-на-Дону: изд. «Феникс», 2000. – 448 с.
- 4 Новиков, Ю. В. Основы цифровой схемотехники. Базовые элементы и схемы. / Ю. В. Новиков. - М. : Мир, 2001.-379 с.
- 5 Опадчий, Ю.Ф. Аналоговая и цифровая электроника / Ю.Ф. Опадчий, О.П. Глудкин. – М: Горячая линия -Телеком, 2000. – 768 с.
- 6 Угрюмов, Е. П. Цифровая схемотехника: учеб. пособие / Е. П. Угрюмов .- 2-е изд., перераб. и доп. - СПб. : БХВ-Петербург, 2007. - 800 с.
- 7 Прянишников, В.А. Электроника: курс лекций. / В.А. Прянишников. – СПб.: КОРОНА, 2000. – 416с.
- 8 Фрике, К. Вводный курс цифровой электроники: учеб. пособие для студентов: пер. с нем. / К. Фрике. - М. : Техносфера, 2003. - 432 с.