

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Фурсов Владимир Алексеевич

Должность: И.о. директора Пятигорского института (филиал) Северо-Кавказского
федерального университета

Дата подписания: 08.06.2026 13:46:16

Уникальный программный ключ:

1c378726a41fd0143ae5bccc8ba81860b00daa62

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение
высшего образования**

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Пятигорский институт (филиал) СКФУ

Колледж Пятигорского института (филиал) СКФУ

**МДК. 01.01 ОСНОВЫ ПРОЕКТИРОВАНИЯ ЦИФРОВОЙ ТЕХНИКИ
МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ**

Специальности СПО

09.02.01 Компьютерные системы и комплексы

Квалификация специалист по компьютерным системам

Методические указания для практических занятий по МДК. 01.01 Основы проектирования цифровой техники составлены в соответствии с требованиями ФГОС СПО. Предназначены для студентов, обучающихся по специальности 09.02.01 Компьютерные системы и комплексы.

Оглавление

Практическая работа №1. Изучение видов и условных обозначений цифровых микросхем.	8
Практическое занятие №2. Изучение конструкторских и электрических параметров цифровых микросхем.	12
Практическая работа №3. Изучение форм сигналов и их параметров.	15
Практическая работа №4. Представление чисел в различных системах счисления. Перевод из одной системы счисления в другую.	19
Таблицы чисел в различных системах счисления	Таблица 1 19
Практическая работа №5. Схемная реализация функций алгебры логики (ДНФ)	24
Практическая работа №6. Схемная реализация функций алгебры логики (КНФ)	25
Практическая работа №7. Минимизация функций алгебры логики.	27
Практическая работа №8. Проектирование одновыходной комбинационной схемы.....	31
Практическая работа №9. Расчет динамических параметров схемы	33
Практическая работа № 10. Синтез дешифраторов	37
Практическая работа №11. Каскадное соединение дешифраторов.....	40
Практическая работа №12. Синтез преобразователя двоичных кодов.	45
Практическая работа №13. Преобразователи произвольных кодов.....	49
Практическая работа №14. Синтез мультиплексоров.	54
Практическая работа № 15. Построение функциональной схемы сумматора.	57
Практическая работа №16. Исследование работы АЛУ.....	59
Практическая работа №17. Цифровые компараторы.....	62
Практическая работа №18. Разработка и моделирование процессов передачи информации через драйверные схемы.....	65
Практическая работа №19. Исследование принципов построения и функционирования цифровых автоматов.	66
Практическая работа №20. Моделирование и исследование логики работы асинхронных RS-триггеров.	73
Практическая работа №21. Моделирование и исследование логики работы синхронных RS-триггеров.	75
Практическая работа №22. Моделирование и исследование логики работы D-триггеров.....	78
Практическая работа №23. Моделирование и исследование логики работы T-триггеров.	81
Практическая работа №24. Моделирование и исследование логики работы JK-триггеров.	83
Практическая работа №25. Реализация функций, выполняемых статическими регистрами.	85
Практическая работа №26. Синтез и исследование работы регистров.....	90
Практическая работа №27. Синтез регистра сдвига.	98
Практическая работа №28. Синтез суммирующего счетчика.	101
Практическая работа №29. Синтез вычитающего счетчика.	103
Практическая работа №30. Синтез реверсивного счетчика.	105
Практическая работа №31. Разработка принципиальной схемы генератора двоичных чисел на D-триггерах.	106
Практическая работа №32. Исследование двойного преобразования (АЦП-ЦАП).	109
Практическая работа №33. Исследование аналого-цифровых преобразователей.....	111
Практическая работа №34. Исследование цифроаналоговых преобразователей.	113
Практическая работа №35. Электронный усилитель. Биполярные транзисторы.	120
Практическая работа №36. Исследование параметров операционных усилителей	122
Практическая работа №37. Исследование статических свойств операционных усилителей.....	126
Практическая работа №38. Исследование динамических свойств операционных усилителей	128
Практическая работа №39. Расчет режимов работы транзисторного ключа.	131
Практическая работа №40. Исследование логического элемента ТТЛ.	134
Практическая работа №41. Микросхемы эмиттерно–связанной логики (ЭСЛ).	135
Практическая работа №42. Микросхемы на КМОП (КМДП) – транзисторах.....	136
Практическая работа №43. Разработка электронной схемы программируемой логической матрицы.	137
Практическая работа №44. Синтез схем на программируемых логических матрицах.	139
Практическое занятие №45. Запоминающие устройства	142
Практическое занятие №46. Исследование режимов работы статического ОЗУ	146

Практическое занятие №47. Изучение структуры и принципов работы постоянного запоминающего устройства (ПЗУ) с неизменяемой информацией.....	148
Практическое занятие №48. Исследование работы репрограммируемого постоянного запоминающего устройства.....	153
Практическое занятие №49. Диагностика и обслуживание FLASH накопителей.....	157
Практическое занятие №50. Принципы работы кэш-памяти.....	160

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Программа МДК. 01.01 Основы проектирования цифровой техники предусматривает изучение и классификацию основных видов микросхем.

При изучении предмета следует соблюдать единство терминологии и обозначения в соответствии с действующими стандартами, Международной системной единицы (СИ).

В результате изучения МДК. 01.01 Основы проектирования цифровой техники студенты *должны*

уметь:

- применять методы анализа требований;
- применять рекомендуемые нормативные и руководящие материалы на разрабатываемые цифровые системы;
- применять системы автоматизированного проектирования;
- осуществлять компьютерное моделирование цифровых устройств с использованием конструкторских систем автоматизированного проектирования;
- оформлять результаты тестирования цифровых устройств;
- применять рекомендуемые нормативные и руководящие материалы на разрабатываемую техническую документацию;
- пользоваться стандартным программным обеспечением при оформлении документации;
- разрабатывать рабочие чертежи в соответствии с требованиями стандартов организации, национальных стандартов и технических регламентов;
- применять имеющиеся шаблоны для составления технической документации;
- использовать прикладные программы для разработки конструкторской документации;
- работать в средах моделирования цифровых устройств и систем;
- выполнять тестирование прототипов.

знать:

- основные параметры и условия эксплуатации систем;
- особенности построения, применения и подключения основных типов цифровых устройств;
- электронные справочные системы и библиотеки: наименования, возможности и порядок работы в них;
- технические характеристики типовых цифровых устройств;
- особенностей применения и подключения основных типов цифровых устройств;
- основы электротехники и силовой электроники;
- полупроводниковой электроники;
- основы цифровой схемотехники;
- основы аналоговой схемотехники;
- основы микропроцессоров;
- основные понятия теории автоматического управления;
- номенклатуру основных радиоэлектронных компонентов: назначения, типы, характеристики;
- типы, основные характеристики, назначение радиоматериалов;

типы, основные характеристики, назначение материалов базовых несущих конструкций радиоэлектронных средств;
специальные пакеты прикладных программ для конструирования радиоэлектронных средств: наименования, возможности и порядок работы в них;
основные методы проведения электротехнических измерений и основы метрологии;
электронные справочные системы и библиотеки: наименования, возможности и порядок работы в них;
виды и содержание конструкторской документации на цифровые устройства;
основные требования Единой системы конструкторской документации (далее - ЕСКД);
правила оформления и внесения изменений в техническую и эксплуатационную документацию;
специальные пакеты прикладных программ для разработки конструкторской документации: наименования, возможности и порядок работы в них;
прикладные компьютерные программы для создания графических документов: наименования, возможности и порядок работы в них;
технические характеристики типовых цифровых устройств;
особенностей применения и подключения основных типов цифровых устройств;
среды моделирования цифровых устройств и систем;
методы построения компьютерных моделей цифровых устройств;
методы обеспечения качества на этапе проектирования;
требования охраны труда, пожарной, промышленной, экологической безопасности и электробезопасности.

Общие методические указания к выполнению практических работ

По МДК. 01.01 Основы проектирования цифровой техники практические работы содержат задачи и теоретические вопросы. Варианты для каждого обучающегося – индивидуальные.

Задачи и ответы на вопросы, выполненные не по своему варианту, не засчитываются.

Практическая работа выполняется в отдельной тетради. Условия задачи и формулировки вопросов переписываются полностью. Формулы, расчеты, ответы на вопросы пишутся ручкой, а чертежи, схемы и рисунки выполняются карандашом, на графиках и диаграммах указывается масштаб. Вначале задача решается в общем виде, затем делаются расчёты по условию задания. Решение задач обязательно ведется в Международной системе единиц (СИ).

При выполнении практической работы необходимо следовать методическим указаниям: повторить краткое содержание теории, запомнить основные формулы и законы, проанализировать пример выполнения аналогичного задания, затем преступить непосредственно к решению задачи. К зачету допускаются студенты, получившие положительные оценки по всем практическим работам.

Правила выполнения практических работ.

1. Студент должен прийти на практическое занятие подготовленным к выполнению практической работы.
2. Каждый студент после проведения работы должен представить отчет о проделанной работе с анализом полученных результатов и выводом по работе.
3. Таблицы и рисунки следует выполнять с помощью чертежных инструментов (линейки, циркуля, и т.д.) карандашом с соблюдением ЕСКД.
4. Расчет следует проводить с точностью до двух значащих цифр.
5. Исправления проводить на обратной стороне листа. При мелких исправлениях неправильное слово (буква, число и т.п.) аккуратно зачеркивается и над ним пишут правильное пропущенное слово (букву, число и т.п.).
6. Вспомогательные расчеты можно выполнять на отдельных листах, а при необходимости на листах отчета.
7. Если студент не выполнит практическую работу или часть работы, то он выполнит ее во внеурочное время, согласованное с преподавателем.
8. Оценку по практической работе студент получает с учетом срока выполнения работы, если;
 - расчеты выполнены правильно и в полном объеме;
 - сделан анализ проделанной работы и вывод по результатам работы;
 - студент может пояснить выполнение любого этапа работы;
 - отчет выполнен в соответствии с требованиями к выполнению работы.

Практическая работа №1. Изучение видов и условных обозначений цифровых микросхем.

Цель работы: изучение видов и условных обозначений цифровых микросхем, приобретение навыков определения правильности работы логического элемента.

Краткие теоретические сведения

Цифровые микросхемы предназначены для преобразования и обработки сигналов, изменяющихся по законам дискретной функции. Они применяются для построения ЦВМ, а также цифровых узлов измерительных приборов, аппаратуры автоматического управления, связи и т. д.

По конструктивно-технологическому исполнению все цифровые ИС делятся на группы. По характеру выполняемых функций в аппаратуре ИС подразделяются на подгруппы (к примеру, логические элементы, триггеры и т.д.) и виды внутри подгрупп (к примеру, триггеры с задержкой, триггеры универсальные и т.д.). Разделение цифровых ИС на подгруппы и виды по функциональному назначению приведены в таблице 1.1.

Таблица 1.1.

Подгруппы и виды ЦИС

Подгруппа и вид ИС	Обозначение
Схемы арифметических и дискретных устройств:	ИА
шифраторы	ИБ
дешифраторы	ИД
счетчики	ИЕ
комбинированные	ИК
полусумматоры	ИЛ
сумматоры	ИМ
прочие	ИП
регистры	ИР
Логические элементы	
И-НЕ	ЛА
И-НЕ/ИЛИ-НЕ	ЛБ
расширители	ЛД
ИЛИ-НЕ	ЛЕ
И	ЛИ
И-ИЛИ-НЕ/И-ИЛИ	ЛК
ИЛИ	ЛЛ
ИЛИ-НЕ/ИЛИ	ЛМ
НЕ	ЛН
прочие	ЛП
И-ИЛИ-НЕ	ЛР
И-ИЛИ	ЛС
Схемы запоминающих устройств (ЗУ)	
ассоциативные ЗУ	РА
матрицы постоянных ЗУ	РВ
матрицы оперативных ЗУ	РМ
постоянные ЗУ (масочные)	РЕ
прочие	РП

постоянные ЗУ с возможностью многократного электрического перепрограммирования	РТ
оперативные ЗУ	РУ
постоянные ЗУ с ультрафиолетовым стиранием и электрической записью информации	РФ
Триггеры	
универсальные (типа JK)	ТВ
динамические	ТД
комбинированные	ТК
Шмитта	ТЛ
с задержкой (типа D)	ТМ
прочие	ТП
с раздельным запуском (типа RS)	ТР
счетные (типа Т)	ТТ

Сведения о подгруппе и виде микросхемы содержатся в ее условном обозначении.

В соответствии с ГОСТ 17021—75 обозначение цифровых ИС должно состоять из четырех элементов. Первый из них — цифра (1, 5, 7), обозначающая группу ИС. Она определяется конструктивно-технологическим исполнением ИС. Второй элемент — две или три цифры (от 00 до 99 либо от 000 до 999), указывающие порядковый номер разработки серии ИС. Третий элемент — две буквы, обозначающие подгруппу и вид микросхемы, определяющие основные функциональные назначения ИС (таблица). Четвертый элемент — число, обозначающее порядковый номер разработки ИС по функциональному признаку в данной серии.

Два первых элемента обозначают серию ИС. Под серией понимают совокупность типов ИС, которые могут выполнять различные функции, имеют единое конструктивно-технологическое исполнение и предназначены для совместного применения.

Пример условного обозначения интегральной полупроводниковой логической микросхемы К155ЛА3, представляющей логический элемент И-НЕ с порядковым номером разработки серии — 55, порядковым номером разработки данной схемы в серии по функциональному признаку — 3 приведен ниже (рисунок 1)

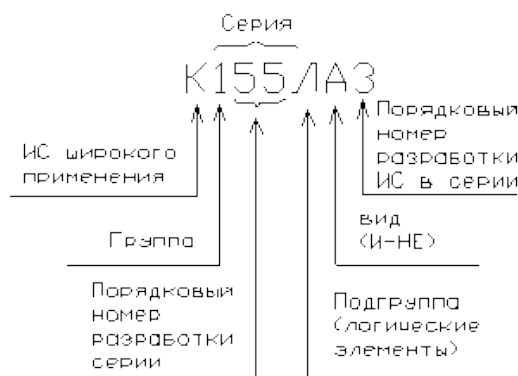


Рисунок 1.

При крайней важности разработчик ИС вправе после порядкового номера разработки ИС по функциональному признаку в данной серии дополнительно поместить букву (от А до Я), обозначающую отличие электрических параметров ИС одного типа (к примеру, 531ЛА1П). Конечная буква при маркировке должна быть заменена точкой. Цвет ее указывается в технических условиях (ТУ) на ИС конкретных типов. Для микросхем, используемых в устройствах широкого применения, в начале обозначения добавляется буква К (к примеру, К1533ЛА3). Как правило, ИС с буквой К отличаются от микросхем, не имеющих ее, условиями приемки на заводе-изготовителе, т. е. отличаются не только диапазоном температур, при которых они бывают использованы, но и численными значениями некоторых параметров.

В последнее время для некоторых ИС после буквы К ставится дополнительная буква, указывающая особенность конструктивного исполнения (к примеру, КР, КМ, КФ).

Для бескорпусных ИС перед цифровым обозначением серии добавляют букву Б, а после обозначения порядкового номера разработки ИС по функциональному признаку в данной серии (или после дополнительного буквенного обозначения) через дефис указывают цифру, характеризующую модификацию конструктивного исполнения (к примеру, Б133ЛА3-1).

Согласно ГОСТ 2.743-82 условное графическое обозначение (УГО) элемента цифровой логики имеет форму прямоугольника, к которому подводят линии выводов. УГО элементов может содержать три поля: основное и два дополнительных. Дополнительные поля располагают слева и справа от основного. Допускается дополнительные поля разделять на зоны, которые отделяют горизонтальной чертой. В первой строке основного поля помещают обозначение функции, выполняемой элементом. В дополнительных полях помещают информацию о функциональных назначениях выводов. Линии выводов характеризуются меткой и указателем. Метка - это наименование вывода. Указатель характеризует свойства вывода. Входы элемента изображают с левой стороны УГО, выходы – с правой стороны. Размеры УГО определяются по высоте:

- количеством линий выводов;
 - количеством интервалов;
 - количеством строк информации в основном и дополнительных полях;
 - размером шрифта;
- по ширине:
- наличием дополнительных полей;
 - количеством знаков, помещаемых в одной строке внутри УГО;
 - размером шрифта.

Расстояние между линиями выводов должно быть не менее и кратным величине С (минимальное $C = 5$ мм).

Расстояние между горизонтальной стороной УГО, границей зоны и линией вывода должно быть не менее и кратным $C/2$ (рисунок 2).

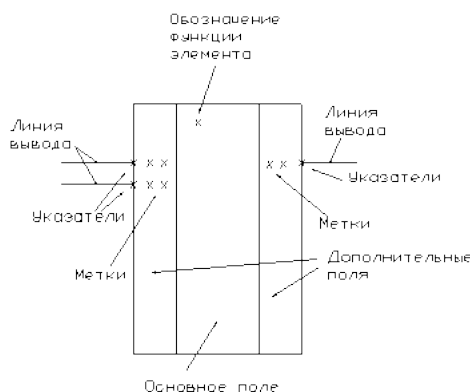


Рисунок 2

Начертание и размеры условных графических обозначений (далее для краткости — УГО) элементов должны быть такими, как указано в стандартах.

Составляя схему устройства, следует придерживаться общепринятого правила вход — слева, выход — справа.

УГО наиболее часто встречающихся в схемах элементов и их размеры в масштабе 1:1 приведены на рисунке. Возле каждого элемента (желательно сверху или справа) должно быть указано его позиционное обозначение (R1, R2, ..., C1, C2 и т.д.). Нумеровать элементы крайне важно слева направо — сверху вниз, к примеру, так:

R1 R4 R7 R9
R2 R5
R3 R6 R8 R10

Для упрощения схем нередко используют слияние линий электрической связи в одну так называемую групповую линию связи, которую изображают утолщенной линией. В непосредственной близости от мест входа в групповую линии обычно нумеруют. Вместо номеров

можно использовать буквенные обозначения сигналов, иногда это упрощает чтение схемы. Минимальное расстояние между соседними линиями, отходящими от групповой в разные стороны, должно быть не менее 2 мм (в масштабе 1:1). Линии, выходящие из конца линии групповой связи, изображают линиями нормальной толщины.

УГО микросхем цифровой и аналоговой техники построены на базе прямоугольников, называемых полями. УГО простейших устройств (к примеру, логических элементов) состоят только из основного поля, в более сложных к нему добавляют одно или два дополнительных, располагаемых слева и справа. В основном поле помещают надписи и знаки, обозначающие функциональное назначение элемента или микросхемы, в дополнительных – так называемые метки, поясняющие назначение выводов. Ширина полей определяется числом знаков (с учетом пробелов) Минимальная ширина основного поля – 10, дополнительных – 5 мм. Расстояние между выводами, а также между выводом и горизонтальной стороной УГО или границей зоны, отделяющей одни выводы от других, – 5 мм (все размеры в масштабе 1:1).

В местах присоединения линий-выводов изображают специальные знаки (указатели), характеризующие их особые свойства, небольшой кружок (инверсия), наклонную черточку ("/" – прямой, "\" – инверсный динамический вход), крестик (вывод, не несущий логической информации, к примеру, вывод питания).

В правом поле УГО цифровых микросхем иногда помещают знаки, построенные на базе ромбика. В случае если он снабжен черточкой сверху, это означает, что данный вывод соединен с коллектором р-п-р транзистора, эмиттером п-р-п транзистора, стоком полевого с р-каналом или истоком транзистора с п-каналом. В случае если же названные электроды принадлежат транзисторам противоположной структуры или приборам с каналом противоположного типа, черточку помещают снизу. Ромбиком с черточкой внутри обозначают вывод с так называемым состоянием высокого выходного сопротивления (Z-состоянием).

Чтобы не загромождать схему цепями питания цифровых микросхем, соответствующие выводы в их УГО обычно не изображают, а, чтобы было ясно, к каким выводам подводится питание, в местах, откуда оно поступает (выход источника питания, цепь, к которой подключается внешний источник), помещают стрелки с адресами, к примеру, "К выв. 14 DD1, DD2, выв. 10 DD3, DD4, выв. 16 DD5, DD6"

И, наконец, – об УГО, используемых в структурных и функциональных схемах. Их основа – квадрат, в котором указывается функциональное назначение устройства. В частности, символ генератора помимо буквы G, может содержать область частот (одна синусоида – низкие частоты, две – звуковые, три – высокие), конкретное значение частоты (к примеру, 500 кГц), форму колебаний в виде упрощенной осциллограммы, наличие стабилизации частоты и т. д.

Два или три символа синусоиды используют также для указания назначения фильтров, но здесь они обозначают полосы частот. К примеру, в УГО фильтров верхних (ФВЧ) и нижних частот (ФНЧ) две синусоиды символизируют колебания частот, лежащих выше и ниже частоты раздела (в первом случае зачеркнута нижняя синусоида, следовательно, устройство пропускает сигналы с частотой выше частоты среза, во втором – верхняя, что говорит о пропускании сигналов ниже этой частоты). В УГО полосового и режекторного фильтров – три синусоиды. Как и в предыдущем случае, пропускаются полосы частот, обозначенные не зачеркнутыми синусоидами: если зачеркнуты верхняя и нижняя, – фильтр полосовой, а если средняя, – режекторный.

Усилители обозначают либо квадратом с треугольником – символом усиления – внутри, либо равнобедренным треугольником (вершина с выводом выхода – направление передачи сигнала). Предпочтительно второе УГО: оно более наглядно и к тому же позволяет указать в нем, к примеру, число каскадов устройства (его вписывают в треугольник).

УГО линий задержки вместо символов сосредоточенных и распределенных параметров могут содержать численное значение времени задержки, а также знаки, обозначающие способ преобразования: пьезоэлектрический (в виде символа кварцевого резонатора), магнитострикционный (две горизонтально расположенные полуокружности).

Содержание отчета

Отчет по работе должен содержать:

- а) наименование и цель работы;
- б) технические данные цифровых микросхем;

в) выводы по работе.

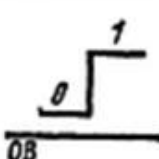
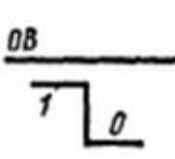
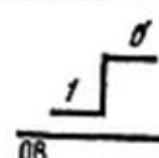
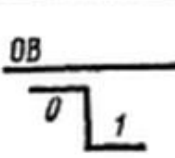
Практическое занятие №2. Изучение конструкторских и электрических параметров цифровых микросхем.

Цель работы: изучение терминов, определений, классификации и системы условных обозначений, применяемых в микроэлектронике, а также конструктивно-технологических параметров полупроводниковых микросхем.

Краткие теоретические сведения.

Большинство цифровых микросхем относятся к потенциальным микросхемам: сигнал на их входе и выходе представляется высоким и низким уровнем напряжения. Указанным двум состояниям сигнала ставятся в соответствие логические значения 1 и 0. В зависимости от кодирования состояния двоичного сигнала различают положительную и отрицательную логику (табл. 1).

Таблица 1

Вид логики	Полярность напряжения питания	
	Положительная	Отрицательная
Положительная		
Отрицательная		

Примечание. Состояния сигнала показаны относительно уровня нулевого потенциала.

Логические операции, выполняемые микросхемами, обычно указывают для положительной логики. Однако есть и исключения из этого правила, они в тексте будут оговорены.

Длительность потенциального сигнала определяется сменой информации: например, длительность сигнала на выходе микросхемы определяется временным интервалом между двумя входными сигналами. Иногда применительно к потенциальным микросхемам говорят, что они управляются положительными или отрицательными импульсами. В таких случаях речь идет о том, что для изменения состояния микросхемы необходимо на заданное время изменить уровень входного сигнала с 1 на 0 (отрицательный импульс) либо с 0 на 1 (положительный импульс).

Свойства цифровых микросхем характеризуют системой электрических параметров, которые для удобства рассмотрения разделим на статические и динамические.

Статические параметры характеризуют микросхему в статическом режиме. К ним относятся:

напряжение источника питания $U_{и.п.}$; входное $U^0_{вх}$ и выходное $U^0_{вых}$ напряжения логического 0; входное $U^1_{вх}$ и выходное $U^1_{вых}$ напряжения логической 1; входной $I^0_{вх}$, $I^1_{вх}$ и выходной $I^0_{вых}$, $I^1_{вых}$ токи логического 0 и логической 1;

коэффициент разветвления по выходу Краз, определяющий число входов микросхем — нагрузок, которые можно одновременно подключить к выходу данной микросхемы; в этом смысле часто употребляют термин «нагрузочная способность» микросхемы;

коэффициент объединения по входу Коб, определяющий число входов микросхемы, по которым реализуется логическая функция; допустимое напряжение статической помехи $U_{\text{пом}}$;

средняя потребляемая мощность $P_{\text{пот.ср}}$.

Последние два параметра нуждаются в кратком пояснении.

Допустимое напряжение статической помехи характеризует статическую помехоустойчивость микросхемы, т. е. ее способность противостоять воздействию мешающего сигнала, длительность которого значительно превосходит время переключения микросхемы. Такая помеха и названа статической. Напряжение допустимой статической помехи обычно определяется как разность выходного и входного напряжений, соответствующих уровню логической 1 либо уровню логического 0 (в расчет принимается наименьшее значение U_a): $U_{1n} = U_{1BbIX} - U_{1vx}$; $U_{0n} = U_{0vx} - U_{0вых}$.

Средняя потребляемая мощность определяется выражением

$$P_{\text{пот.ср}} = (P_{0\text{пот}} + P_{1\text{пот}}) / 2,$$

где $P_{0\text{пот}}$, $P_{1\text{пот}}$ — потребляемая микросхемой мощность в состоянии соответственно 0 и 1 на выходе.

Общепринятое усреднение потребляемой мощности оправдано тем, что обычно во время работы в составе цифрового устройства логические микросхемы половину времени находятся в открытом состоянии, а другую половину времени — в закрытом.

Средняя потребляемая мощность тесно связана с быстродействием микросхемы (ее временем переключения или рабочей частотой переключения); чем больше средняя потребляемая мощность, тем с большей частотой может переключаться микросхема.

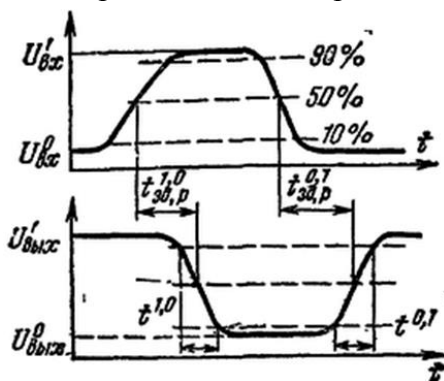


Рис. 1. Временные диаграммы напряжений на входе и выходе логической микросхемы

Для многих типов микросхем характерно заметное увеличение потребляемой мощности с ростом частоты переключения, что связано с увеличением потребления мощности в процессе переключения по сравнению со статическим режимом. Учитывая это, следует при расчетах реального энергопотребления цифрового устройства ориентироваться на мощность, потребляемую микросхемами в режиме переключения с заданной частотой, т. е. на мощность, потребляемую в динамическом режиме.

Динамические параметры характеризуют свойства микросхемы в режиме переключения. В основном это временные параметры микросхемы:

- время перехода из состояния логического 0 в состояние логической 1 $t_{0,1}$;
- время задержки распространения сигнала при выключении микросхемы $t_{0,1зд,р}$;
- время перехода из состояния логической 1 в состояние логического 0 $t_{1,0}$;
- время задержки распространения сигнала при включении микросхемы $t_{1,0зд,р}$;
- среднее время задержки распространения сигнала $t_{зд,р,ср}$.

Динамические параметры определяют при сравнении сигналов на входе и выходе логического элемента. На рис. 1 приведены временные диаграммы входного и выходного сигналов и показаны уровни отсчета, относительно которых определяют динамические параметры.

Среднее время задержки служит усредненным параметром быстродействия и определяется как полусумма задержек $t_{0,1зд,р}$ и $t_{1,0зд,р}$.

Этот параметр часто является основным при расчете рабочей частоты сложных логических устройств.

Среднее время задержки зависит от многих факторов: принципа построения логических элементов, наличия или отсутствия режима насыщения у входящих в схему транзисторов, величины переключающих токов и т. д. Кроме того, на среднее время задержки оказывают существенное влияние и условия работы микросхемы: температура окружающей среды, изменения питающих напряжений, емкость нагрузки и т. д.

Стремление обеспечить высокую надежность аппаратуры заставляет принимать в расчет те значения параметров логических элементов, в том числе и среднего времени задержки, которые соответствуют наихудшим условиям их работы.



Рис. 2. Характеристика динамической помехоустойчивости логической микросхемы

При использовании в расчетах справочных данных необходимо обращать внимание на то, для каких условий приведены эти данные и при необходимости перерасчитывать параметры с учетом реальных условий работы микросхем.

Например, расчеты, уточняющие среднее время задержки, можно производить с помощью коэффициентов, отражающих влияние на значение среднего времени задержки температуры и емкости нагрузки K_c :

$$t_{зд, \text{ р, ср (при } T^*_{\text{н}})} = t_{зд, \text{ р, ср (при } T^*_{\text{н}})} + \alpha_{\theta} (T^*_{\text{н}} - T^*_{\text{н}}),$$

$$t_{зд, \text{ р, ср (при } C_{\text{нн}})} = t_{зд, \text{ р, ср (при } C_{\text{нн}})} + K_c (C_{\text{нн}} - C_{\text{нн}}).$$

При этом предполагается линейная зависимость среднего времени задержки от указанных факторов.

К числу динамических параметров следует отнести также динамическую помехоустойчивость, характеризующую способность микросхемы противостоять воздействию импульсной помехи, длительность которой соизмерима со средним временем задержки передачи сигнала через микросхему.

Количественно динамическая помехоустойчивость определяется амплитудой и длительностью импульса помехи, но чаще с помощью характеристики (рис. 2), отражающей зависимость допустимой амплитуды импульса помехи от длительности этого импульса. Из рисунка видно, что по мере увеличения длительности импульса помехи допустимая амплитуда помехи снижается до уровня максимально-допустимого напряжения статической помехи.

Заметим, что указанные параметры широко используют для характеристики как микросхемы в целом, так и отдельных ее элементов: логических элементов, триггеров и др.

Эксплуатационные параметры характеризуют работоспособность интегральных микросхем в условиях воздействия окружающей среды. К ним относятся: диапазон рабочих температур, допустимые механические нагрузки (вибрации, удары, линейные ускорения), границы допустимого изменения атмосферного давления, наибольшая влажность и некоторые другие.

Ход работы.

1. Для представленных ИМС по маркировке определить тип микросхемы и ее функциональное назначение. Результаты занести в таблицу

Таблица 1

№п/п	Обозначение ИМС	Тип ИМС	Выполняемая функция ИМС
------	-----------------	---------	-------------------------

--	--	--	--

2. Дать классификацию представленных корпусов по форме проекции корпуса на плоскость основания и расположению выводов корпуса.

3. Классифицировать представленные корпуса по конструктивно -технологическому исполнению.

4. Результаты занести в таблицу 2.

Таблица 2

№п/п	Наименование ИМС	Тип корпуса по форме проекции и расположению выводов	Конструктивно-технологическое исполнение корпусов	Область применения

5. Сделать вывод о работе.

Практическая работа №3. Изучение форм сигналов и их параметров.

Цель: изучить основные виды сигналов, их формы; изучить основные параметры импульсного сигнала: низкий и высокий логические уровни, частота повторения, фронт, срез.

Основные теоретические сведения

Форма сигналов.

Сигналом называют физический процесс, несущий информацию. Сигналы могут быть звуковыми, световыми, электрическими.

Информация сосредоточена в изменениях параметров физического процесса. Если параметры процесса не меняются, то не является сигналом. Так, неизменные звук, световой поток, синусоидальное электрическое колебание никакого сообщения не содержат. Наоборот, в изменениях громкости и тона звука, яркости и цвета светового излучения, амплитуды, частоты и фазы электрического колебания запечатлена информация. Информацией является также появление или окончание, например, звукового сигнала, т.е. его изменения.

Сигналы на выходе микрофона, предающей телекамеры, различного рода датчиков аналогичны по своему «рисунку» воздействиям на эти устройства – звуковому давлению, распределению освещенности, температуре и т.п. поэтому подобные сигналы называются *аналоговыми*. Между минимальным и максимальным значениями аналоговый сигнал может иметь любое значение. Обычно аналоговые сигналы являются непрерывными. Устройства, в которых действуют такие сигналы, называют *аналоговыми*.

Прерывистые кратковременные электрические колебания используются в импульсной технике. ЭВМ созданы на основе импульсной техники. Эта техника также является составной частью радиоэлектроники, телевидения, многоканальной связи. Причем различная аппаратура содержит узлы, вырабатывающие импульсы различной формы. Кроме этого, прерывистость импульсных колебаний дает возможность осуществить многоканальную связь. Используя один канал, импульсы, передающие одно сообщение, размещаются в паузах между импульсами, передающими другое сообщение. Устройства, в которых действуют электрические импульсы, называют *импульсными*.

Электрическим импульсом называют отклонение напряжения или тока от некоторого постоянного уровня (в частности, от нулевого), наблюдаемое в течение времени, которое меньше или сравнимо с длительностью переходных процессов в схеме.

Существуют два вида импульсов:

- видеоимпульсы (получившие свое название из телевидения, где они широко используются);
- радиоимпульсы.

Видеоимпульсы получают при коммутации цепи постоянного тока. Наиболее часто используют видеоимпульсы прямоугольной (рис.1а), трапецеидальной (рис.1б), экспоненциальной (остроконечной) (рис.1в), пилообразной (рис.1г) и треугольной (рис.1д) форм.

Различают видеоимпульсы положительной (рис.1,а,б,г,д) и отрицательной (рис.1,в) полярности, а также двусторонние – разнополярные импульсы (рис.1,е). Следует иметь в виду, что реальные импульсы не имеют формы, строго соответствующей названию.

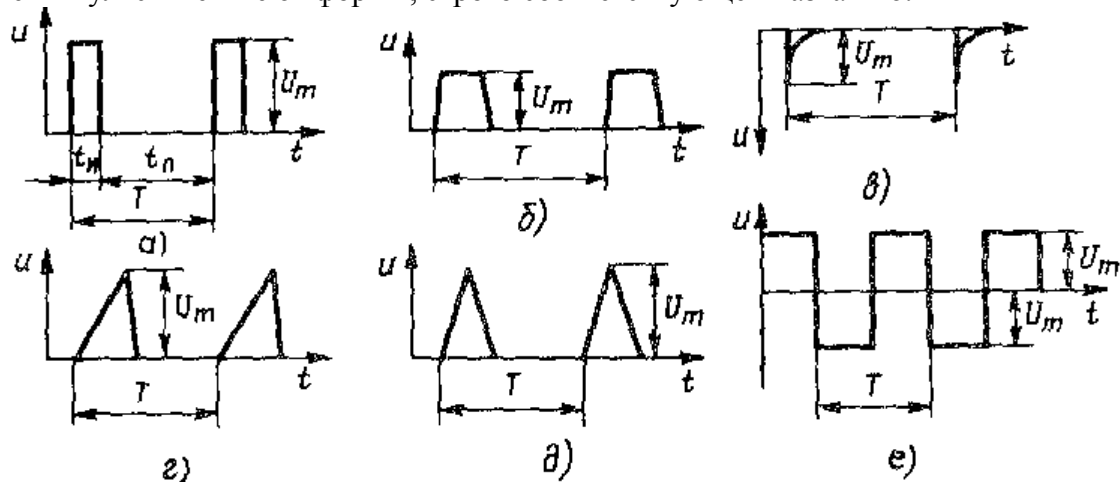


Рисунок 1.

Радиоимпульсы (рис.2) представляют собой кратковременные посылки синусоидального напряжения или тока.

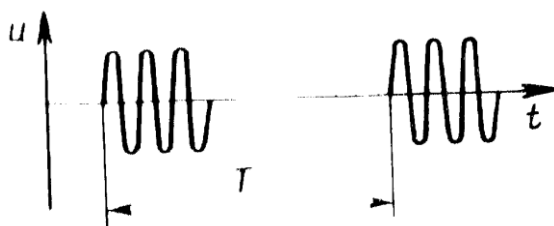


Рисунок 2.

Они снимаются с выхода высокочастотного генератора, который управляется (моделируется) видеоимпульсами. Поэтому форма огибающей радиоимпульсов соответствует форме модулирующих видеоимпульсов. Радиоимпульсы – результат модуляции амплитуды высокочастотного колебания прямоугольными видеоимпульсами.

Импульсное колебание, параметры которого изменяются в соответствии с передаваемой информацией, является сигналом. Такой сигнал относят к аналоговым, т.к. в диапазоне своих изменений он может принимать любое значение. Импульсный сигнал называют *дискретным*, так как он составляется элементами – импульсами, действующими в отдельные (дискретные) моменты времени. Эти импульсы являются выборками (отсчетами) непрерывного сигнала. Процесс взятия отсчетов называется *дискретизацией* непрерывного сигнала, а их период – *период дискретизации*. Вместо передачи непрерывного сигнала можно передавать соответствующий ему дискретный сигнал. В основе этого утверждения лежит теорема отсчетов, сформулированная и доказанная советским академиком В.А. Котельниковым.

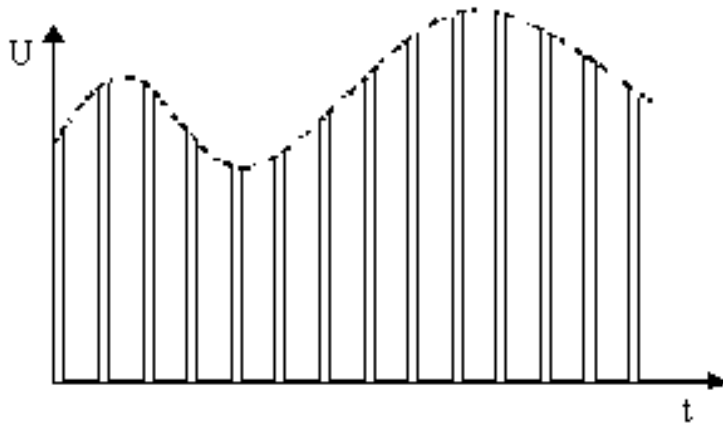


Рисунок 3.

Параметры сигнала: низкий и высокий логические уровни, частота повторения, фронт, срез.

Введем понятие об основных параметрах сигналов (импульсов) на примере реального прямоугольного импульса:

Длительность.

За активную длительность импульса t_{ia} принимают промежуток времени, измеренный на уровне, соответствующем половине амплитуды.

Иногда длительность сигналов определяют на уровне $0,1U_m$ ($0,1I_m$) или по основанию импульса. В дальнейшем, длительность импульса будем определять по основанию и обозначать t_u (см. рис. 1). Длительность импульса выражается в единицах времени: секундах (с), миллисекундах ($мс=10^{-3}с$), микросекундах ($мкс=10^{-6}с$) и наносекундах ($нс=10^{-9}с$).

Амплитуда.

Наибольшее значение напряжения или тока импульса данной формы является амплитудой U_m (I_m).

Амплитуда импульса U_m (I_m) выражается в вольтах (В), киловольтах ($кВ=10^3В$), милливольтмах ($мВ=10^{-3}В$) и микровольтах ($мкВ=10^{-6}В$) (или амперах (А), миллиамперах (мА) и микроамперах (мкА)).

Длительность и крутизна фронта импульса.

Импульс имеет *передний фронт* и *срез*, последний также называют *задним фронтом*.

Длительность переднего фронта импульса определяется временем нарастания импульса, а длительность среза – временем спада импульса.

Наиболее часто пользуются понятием *активной длительности фронта* t_f , за которую принимают время нарастания импульса от $0,1U_m$ до $0,9U_m$; аналогично, длительность среза t_c – время спада импульса от $0,9U_m$ до $0,1U_m$.

Обычно длительность t_f и t_c составляет единицы процентов от длительности импульса. Чем меньше t_f и t_c по сравнению с t_u , тем больше форма импульса приближается к прямоугольной.

Иногда вместо t_f и t_c фронты импульса характеризуют *крутизной фронта (среза)* и выражают в вольтах в секунду (В/с), киловольтах в секунду (кВ/с). Участок импульса между фронтами называют *плоской вершиной*. На рисунке показан спад плоской вершины (ΔU), а также отрицательный выброс.

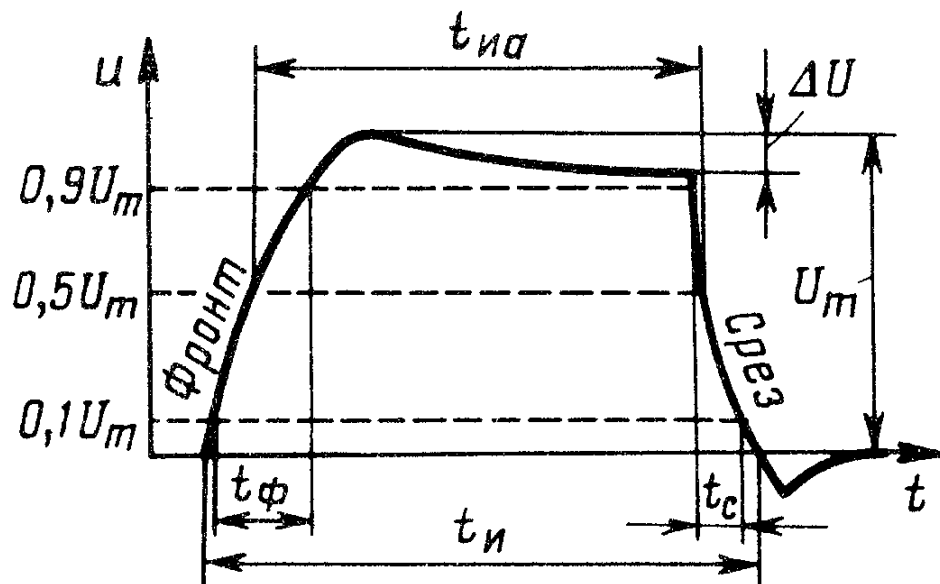


Рисунок 4.

Период повторения импульсов. Импульсы, повторяющиеся через разные промежутки времени, образуют периодическую последовательность.

Промежуток времени между началом двух соседних однополярных импульсов (рис.1а) называют периодом повторения (следования) импульсов T .

Он выражается в единицах времени: с, мс, мкс.

Величину, обратную периоду повторения, называют частотой повторения (следования) импульсов f .

Она определяет количество периодов в течение 1с и выражается в герцах (Гц), килогерцах (кГц) и т.д.

Указания по выполнению работы.

1. Ознакомиться с основными теоретическими положениями.
2. Составить классификацию видов сигналов:
3. Составить классификационную таблицу «Виды и формы импульсов»:

характеристика		
форма		
полярность		

4. Зарисовать рисунок реального прямоугольного импульса, охарактеризовать его основные параметры.

5. Составить отчет о работе. Сделать выводы. Ответить на контрольные вопросы.

Контрольные вопросы.

1. Что называется сигналом? Привести примеры.
2. Какой процесс может называться информационным?
3. Какие устройства называются аналоговыми?
4. Что такое электрический импульс?
5. На какие виды делят электрические импульсы, чем они отличаются друг от друга?
6. Почему импульсный сигнал называется дискретным?

Практическая работа №4. Представление чисел в различных системах счисления. Перевод из одной системы счисления в другую.

Цель работы. Изучение методов перевода чисел из одной системы счисления в другую.
Изучение способов представления числовой информации в компьютере

План

1. Изучить методы перевода целых чисел из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления
2. Изучить методы перевода правильных дробей из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления
3. Изучить методы перевода вещественных чисел из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления
4. Изучить методы перевода чисел из двоичной, восьмеричной и шестнадцатеричной систем счисления в десятичную систему счисления
5. Изучить методы перевода чисел из двоичной системы счисления в восьмеричную и шестнадцатеричную системы счисления и наоборот
6. Изучить представление целых и вещественных чисел в компьютере
7. Ответить на контрольные вопросы

Краткие сведения

В двоичной системе счисления все числа записываются с помощью двух цифр 0 или 1, основание (базис) двоичной системы счисления $q=2$.

В восьмеричной системе счисления все числа записываются с помощью восьми цифр 0, 1, 2, 3, 4, 5, 6, 7, основание восьмеричной системы счисления $q=8$.

В десятичной системе счисления все числа записываются с помощью десяти цифр 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

В шестнадцатеричной системе счисления все числа записываются с помощью шестнадцати цифр 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, А (количественный эквивалент числа 10), В (11), С (12), D(13), Е (14), F (15), базис шестнадцатеричной системы счисления $q=16$. Рассмотрим соотношение цифр и чисел в различных системах счисления (Таблица 1).

Таблицы чисел в различных системах счисления

Таблица 1

10-я	2-я	8-я	16-я
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9

10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

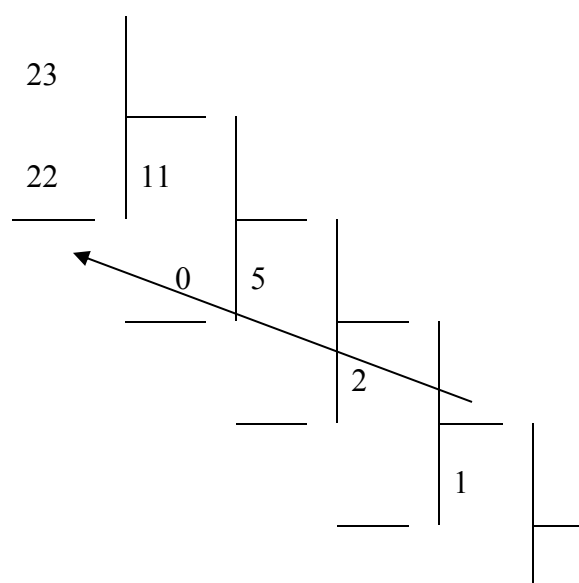
Для того чтобы перевести целую часть числа из десятичной системы счисления в двоичную, восьмеричную, шестнадцатеричную, необходимо:

1. Последовательно делить данное число и получаемые при делении целые частные на основание q новой системы счисления (2, 8 или 16), выражая промежуточный результат цифрами исходной системы, до тех пор, пока частное не станет равным нулю.

2. Полученные остатки, фактически являющиеся цифровым выражением числа в новой системе, привести в соответствие с алфавитом этой системы счисления (имеются в виду системы счисления с основанием больше 10).

3. Записать число в новой системе счисления, начиная с последнего остатка.

ПРИМЕР 1. ПЕРЕВЕСТИ ЧИСЛО 23_{10} В ДВОИЧНУЮ СИСТЕМУ СЧИСЛЕНИЯ.



ПОЛУЧАЕМ $23_{10}=10111_2$

Для того, чтобы выполнить перевод правильных дробей из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную необходимо последовательно умножать правильную дробь и получаемые дробные части произведений на основание системы q до тех пор, пока дробная часть произведения не станет равной нулю или не будет достигнута требуемая точность представления числа.

Полученные целые части произведений, являющиеся цифрами числа в новой системе счисления, привести в соответствие с алфавитом новой системы счисления. Составить дробную часть числа в новой системе счисления, начиная с целой части первого произведения.

Пример 2. Перевести число $0,65625_{10}$ в восьмеричную систему счисления.

$$\begin{array}{r}
 6 \\
 5625 \times \\
 \hline
 2 \\
 5000 \\
 \hline
 0
 \end{array}$$

Получаем: $0,65625_{10} = 0,52_8$

Перевод вещественных чисел, т.е. чисел, содержащих целую и дробную части, осуществляется в два этапа. Отдельно переводится целая часть, отдельно — дробная. В итоговой записи полученного числа целая часть отделяется от дробной запятой.

Пример 3. Перевести число $156,356_{10}$ в шестнадцатеричную систему счисления. Представим число в виде суммы целой и дробной части $156,356_{10}=156_{10}+0,356_{10}$

Переводим целую часть 156_{10} в шестнадцатеричную систему счисления:

	1	5
156	6	
144	9	6
2		

C)

Переводим дробную часть $0,356_{10}$ в шестнадцатеричную систему счисления:

0	356
x 16	
5	6
96	
1	1
1, 36	
2	1
76	

Получаем: $0,356_{10} \approx 0,5B2_{16}$.

Получаем $156_{10}=9C_{16}$

Таким образом, $9C_{16}+0,5B_{216}=9C,5B_{216}$. Получаем: $156,356_{10} \approx 9C,5B_{16}$

При переводе чисел из системы счисления с основанием q в десятичную систему счисления необходимо пронумеровать разряды целой части справа налево, начиная с нулевого, и дробной части, начиная с разряда сразу после запятой, слева направо – начальный номер -1. Затем вычислить сумму произведений соответствующих значений разрядов на основание q системы счисления в степени, равной номеру разряда

Пример 4. Перевести число $10110110111,101_2$ в десятичную систему счисления

$$1_2 = 1 \cdot 2^9 + 0 \cdot 2^8 + 1 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} = 512 + 0 + 128 + 64 + 0 + 16 + 8 + 0 + 2 + 1 + 0,5 + 0 + 0,125 = 731,625_{10}$$

Если необходимо перевести число из двоичной системы счисления в систему счисления, основанием которой является степень двойки, достаточно объединить цифры двоичного числа в группы по столько цифр, каков показатель степени. При этом в целой части числа группировка производится справа налево, а в дробной слева направо. Если в последней группе недостает цифр, дописываются нули: в целой части - слева, в дробной – справа. Затем каждая группа заменяется цифрой новой системы счисления (Таблица 1).

При переводе чисел из системы счисления, основанием которой является степень двойки, в двоичную систему счисления необходимо каждую цифру заменить группой по столько цифр двоичной системы счисления, каков показатель степени. Затем записать цифры слева направо.

Пример 5. Перевести число $1011010010111,1011_2$ в восьмеричную систему счисления

Разбиваем число на тройки цифр и заменяем каждую триаду восьмеричной цифрой:

001 011 010 010 111,101 100₂

1 3 2 2 7 5 4

Получим 1011010010111,101100₂=13227,54₈

Пример 6. Перевести число 801A9E,3F₁₆ в двоичную систему счисления

Заменим каждую цифру четверкой двоичных цифр: 8 0 1 A 9 E, 3 F₁₆

1000 0000 0001 1010 1001 1110 0011 1111

Получим 801A9E,3F₁₆=100000000001101010011110,00111111₂

Задания к практической работе

1) Выполнить перевод целых чисел из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления:

1. 26 ₁₀	2. 27 ₁₀	3. 28 ₁₀	4. 29 ₁₀	5. 30 ₁₀	6. 31 ₁₀	7. 42 ₁₀	8. 43 ₁₀	9. 44 ₁₀	10. 45 ₁₀
11. 46 ₁₀	12. 47 ₁₀	13. 58 ₁₀	14. 59 ₁₀	15. 60 ₁₀	16. 61 ₁₀	17. 62 ₁₀	18. 63 ₁₀	19. 74 ₁₀	20. 75 ₁₀
21. 76 ₁₀	22. 77 ₁₀	23. 78 ₁₀	24. 79 ₁₀	25. 90 ₁₀	26. 91 ₁₀	27. 92 ₁₀	28. 93 ₁₀	29. 94 ₁₀	30. 95 ₁₀

2) Выполнить перевод правильных дробей из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления:

1) 0,25 ₁₀	2) 0,125 ₁₀	3) 0,34 ₁₀	4) 0,37 ₁₀	5) 0,39 ₁₀	6) 0,41 ₁₀	7) 0,44 ₁₀	8) 0,49 ₁₀
9) 0,52 ₁₀	10) 0,56 ₁₀	11) 0,59 ₁₀	12) 0,61 ₁₀	13) 0,62 ₁₀	14) 0,63 ₁₀	15) 0,68 ₁₀	16) 0,69 ₁₀
17) 0,73 ₁₀	18) 0,76 ₁₀	19) 0,79 ₁₀	20) 0,82 ₁₀	21) 0,84 ₁₀	22) 0,85 ₁₀	23) 0,86 ₁₀	24) 0,89 ₁₀
25) 0,91 ₁₀	26) 0,93 ₁₀	27) 0,94 ₁₀	28) 0,95 ₁₀	29) 0,96 ₁₀	30) 0,99 ₁₀		

3) Выполнить перевод вещественных чисел из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления:

1. 106, 125 ₁₀	2. 107,456 ₁₀	3. 108,375 ₁₀	4. 109,377 ₁₀	5. 110,378 ₁₀
6. 111,379 ₁₀	7. 122,381 ₁₀	8. 123,382 ₁₀	9. 124,383 ₁₀	10. 125,384 ₁₀
11. 126,385 ₁₀	12. 127,386 ₁₀	13. 138,387 ₁₀	14. 139,388 ₁₀	15. 140,389 ₁₀
16. 141,391 ₁₀	17. 142,393 ₁₀	18. 143,394 ₁₀	19. 154,395 ₁₀	20. 155,396 ₁₀
21. 156,391 ₁₀	22. 157,392 ₁₀	23. 158,393 ₁₀	24. 159,394 ₁₀	25. 170,395 ₁₀
26. 171,396 ₁₀	27. 172,397 ₁₀	28. 173,398 ₁₀	29. 174,399 ₁₀	30. 175,401 ₁₀

4) Выполнить перевод чисел из двоичной, восьмеричной и шестнадцатеричной систем счисления в десятичную систему счисления

а) Выполнить перевод числа из двоичной системы счисления в десятичную:

1) 1001010,1 ₂	2) 1100111,01 ₂	3) 1111000,1 ₂	4) 1001101,01 ₂
5) 1100011,1 ₂	6) 1011111,01 ₂	7) 1100010,1 ₂	8) 1011010,01 ₂
9) 1010101,01 ₂	10) 1011001,1 ₂	11) 10110101,01 ₂	12) 1010100010,1 ₂
13) 10100011,01 ₂	14) 110001001,01 ₂	15) 110101011,1 ₂	16) 11011001,01 ₂
17) 10101001,01 ₂	18) 100011101,1 ₂	19) 1101000,01 ₂	20) 100111,01 ₂
21) 1110101,1 ₂	22) 1010111,01 ₂	23) 1001101,1 ₂	24) 10101111,01 ₂
25) 1111101,1 ₂	26) 10000011,01 ₂	27) 10101011,1 ₂	28) 11001101,01 ₂
29) 1001100111,1 ₂	30) 101011010,01 ₂		

б) Выполнить перевод числа из восьмеричной системы счисления в десятичную:

1. 21, 7 ₈	2. 63, 1 ₈	3. 36, 2 ₈	4. 23,4 8	5. 40, 3 ₈	6. 14, 2 ₈	7. 44, 7 ₈	8. 15, 4 ₈
9. 24, 7 ₈	10. 20, 4 ₈	11. 25, 1 ₈	12. 52,6 8	13. 23, 5 ₈	14. 54, 1 ₈	15. 15, 6 ₈	16. 16, 3 ₈

17.	61,	18.	16,	19.	26,	20.	36,7	21.	35,	22.	31,	23.	37,	24.	32,
5 ₈		4 ₈		3 ₈		4 ₈		1 ₈		7 ₈		2 ₈		5 ₈	
25.	47,	26.	46,	27.	41,	28.	72,1	29.	71,	30.	27,				
2 ₈		1 ₈		5 ₈		8		2 ₈		3 ₈					

в) Выполнить перевод числа из шестнадцатеричной системы счисления в десятичную:

1)	3A,8 ₁₆	2)	F1,A ₁₆	3)	33,A ₁₆	4)	D4,7 ₁₆	5)	FE,1 ₁₆	6)	9B,4 ₁₆
7)	D5,6 ₁₆	8)	F0,9 ₁₆	9)	B5,C ₁₆	10)	B4,2 ₁₆	11)	A3,2 ₁₆	12)	A1,1 ₁₆
13)	1D,4 ₁₆	14)	E9,2 ₁₆	15)	C1,4 ₁₆	16)	2D,A ₁₆	17)	7E,4 ₁₆	18)	7D,6 ₁₆
19)	6C,5 ₁₆	20)	3B,45D ₁₆	21)	7A,2 ₁₆	22)	F6,9 ₁₆	23)	6E,4 ₁₆	24)	CF,5 ₁₆
25)	2B,6 ₁₆	26)	2A,4 ₁₆	27)	CF,9 ₁₆	28)	5B,6 ₁₆	29)	6E,1 ₁₆	30)	28,D ₁₆

5) Выполнить перевод чисел из двоичной системы счисления в восьмеричную и шестнадцатеричную системы счисления и наоборот.

а) Выполнить перевод чисел из двоичной системы счисления в восьмеричную и шестнадцатеричную:

1)	11001101,1011 ₂	2)	1001101,0111 ₂	3)	1001110000,001 ₂	4)	101001010,0101 ₂
5)	1100010010,0111 ₂	6)	1110111100,011 ₂	7)	1100000011,0111 ₂	8)	111010101,101 ₂
9)	110100101011,1 ₂	10)	1011101001,11011 ₂	11)	101101001,01 ₂	12)	1000101110,1001 ₂
13)	1100110101,1 ₂	14)	1011100011,01 ₂	15)	10000001001,0101 ₂	16)	1010000110,01 ₂
17)	11010000000,01 ₂	18)	1001011010,011 ₂	19)	111101110,1011 ₂	20)	1111011110,1 ₂
21)	11111111010,01 ₂	22)	1000110010,0101 ₂	23)	1010001010,1011 ₂	24)	1101010100,011 ₂
25)	1111011010,011 ₂	26)	1011100111,01 ₂	27)	1010011110,101 ₂	28)	1101001010,011 ₂
29)	1111110100,01 ₂	30)	111000011,101 ₂				

б) Выполнить перевод чисел из восьмеричной системы счисления в двоичную:

1) 221,7 ₈	2) 630,61 ₈	3) 736,1 ₈	4) 237,2 ₈	5) 140,31 ₈	6) 14,02 ₈	7) 144,7 ₈	8) 145,4 ₈
9) 24,07 ₈	10) 250,4 ₈	11) 25,31 ₈	12) 252,06 ₈	13) 25,153 ₈	14) 254,7 ₈	15) 15,06 ₈	16) 160,37 ₈
17) 161,54 ₈	18) 162,04 ₈	19) 263,02 ₈	20) 36,74 ₈	21) 305,61 ₈	22) 31,307 ₈	23) 370,27 ₈	24) 321,5 ₈
25) 47,302 ₈	26) 47,613 ₈	27) 414,57 ₈	28) 72,015 ₈	29) 716,25 ₈	30) 271,03 ₈	31)	32)

в) Выполнить перевод чисел из шестнадцатеричной системы счисления в двоичную:

1) 3C9A,8 ₁₆	2) 3BF1,A ₁₆	3) 3F13,A ₁₆	4) 9D4,7A ₁₆	5) 2FE3,61 ₁₆	6) 39B1,4A ₁₆
7) 24D5,67B ₁₆	8) 2F0,519 ₁₆	9) 2B45,0C ₁₆	10) BB4,22C ₁₆	11) 4A3F,29 ₁₆	12) 8A21,10D ₁₆
13) 180D,48F ₁₆	14) E029,72 ₁₆	15) C10,4A7 ₁₆	16) 22D,A17 ₁₆	17) 1EE3,694 ₁₆	18) C7D1,6A ₁₆
19) 6C23,0A ₁₆	20) 3B06,45D ₁₆	21) 7A58,0B ₁₆	22) 1F16,09 ₁₆	23) 16E,0D4 ₁₆	24) 2CF,A35 ₁₆
25) 3F2B,61A ₁₆	26) 24A,14B ₁₆	27) 3CF,D59 ₁₆	28) 3FB,64 ₁₆	29) 6ED0,19C ₁₆	30) 283,D5F ₁₆

Контрольные вопросы

Обязательная часть

1. Чем отличаются позиционные системы счисления от непозиционных?
2. Почему в компьютере используется двоичная система счисления?
3. Какие формы записи применяются в компьютерной технике для кодирования целых чисел со знаком?
4. В чем заключается преимущество экспоненциальной формы числа?

Дополнительная часть

16. Запишите числа в прямом коде (формат 1 байт): 31; 65; -63; -128
17. Запишите числа в обратном коде (формат 1 байт): -9; -127; -15; -128

18. Запишите числа в дополнительном коде (формат 1 байт): -9; -127; -15; -128
19. Переведите пары чисел в двоичную систему счисления и произведите арифметические операции: 36 и 4; 75 и 5; 12 и 4. Ответы проверьте.

Практическая работа №5. Схемная реализация функций алгебры логики (ДНФ)

Цель работы: разработать цифровое устройство, реализующее заданную таблицей истинности функцию алгебры логики (ФАЛ) в форме СДНФ.

Теоретические сведения

1. Аналитические формы записи булевых функций

Рассмотрим аналитическую запись булевых функций и метод перехода от таблицы задания функции к ее аналитическому представлению в форме СДНФ. Он состоит в следующем:

1. Выбрать в таблице задания функций $F(X_1, X_2, \dots, X_n)$ все наборы аргументов, на которых функция обращается в единицу.

2. Для каждого из выбранных наборов выписывается элементарная конъюнкция аргументов данной функции. При этом если аргумент X_i входит в данный набор как единица, то он включается в конъюнкцию без изменения, в противном случае в конъюнкцию включается его отрицание $\bar{x}_i$. Например, имеется функция от трех переменных $F(X_1, X_2, X_3)$, принимающая значение «единица» на наборе $X_1 = 0; X_2 = 1; X_3 = 1$. Этой ситуации будет соответствовать элементарная конъюнкция $\bar{x}_1 \cdot x_2 \cdot x_3$.

3. Все выписанные конъюнкции соединяются знаками дизъюнкции, при этом получаем аналитическую запись исходной функции F . Такую аналитическую запись называют «совершенной дизъюнктивной нормальной формой» (СДНФ) представления функции.

На рис.1 приводится пример получения СДНФ для функции $F(X_1, X_2, X_3)$, заданной таблично. Буквами a, b, c и d обозначены наборы, на которых $F = 1$ и соответствующие им элементарные конъюнкции.

X_1	X_2	X_3	F	
0	0	0	0	i
0	0	1	1	a
0	1	0	0	j
0	1	1	1	b
1	0	0	1	c
1	0	1	1	d
1	1	0	0	k
1	1	1	0	r

$$F = \underbrace{\bar{X}_1 \cdot \bar{X}_2 \cdot X_3}_a \cup \underbrace{\bar{X}_1 \cdot X_2 \cdot X_3}_b \cup \underbrace{X_1 \cdot \bar{X}_2 \cdot \bar{X}_3}_c \cup \underbrace{X_1 \cdot \bar{X}_2 \cdot X_3}_d$$

Рис. 1. Пример получения СДНФ.

Однако представление функции в виде СДНФ является в общем случае не оптимальным с точки зрения ее дальнейшей схемной реализации. Как правило, используя законы склеивания и поглощения, аналитическое выражение для F , заданное в виде СДНФ, можно упростить, а это, в свою очередь, приведет к снижению затрат при физической реализации КС. Упростим, например, СДНФ, полученную на рис.1:

$$F = \bar{X}_1 \cdot \bar{X}_2 \cdot X_3 \cup \bar{X}_1 \cdot X_2 \cdot X_3 \cup X_1 \cdot \bar{X}_2 \cdot \bar{X}_3 \cup X_1 \cdot \bar{X}_2 \cdot X_3 = \bar{X}_1 \cdot X_3 \cdot (\bar{X}_2 \cup X_2) \cup X_1 \cdot \bar{X}_2 \cdot (\bar{X}_3 \cup X_3) = \bar{X}_1 \cdot X_3 \cup X_1 \cdot \bar{X}_2$$

Реализация этого выражения на схемах И и ИЛИ показана на рис.2.

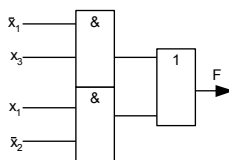


Рис.2. Пример упрощенной КС.

Результаты сравнения показывают, что схема на рис.2 содержит два двухвходовых элемента И и один двухвходовой элемент ИЛИ, а схема на рис.1, реализующая ту же функцию F, содержит четыре трехвходовых элемента И и один четырехвходовой элемент ИЛИ. Следовательно, схема на рис.2 является более экономичной как по количеству элементов, так и по числу входов у этих элементов.

Таким образом, можно сделать вывод о том, что СДНФ не является, как правило, единственно возможной дизъюнктивной формой представления какой-либо булевой функции F. Обычно для этой же функции существуют другие дизъюнктивные формы представления называют: «дизъюнктивные нормальные формы» (ДНФ).

Порядок выполнения работы

1. Для заданной таблицы истинности записать логическую функцию в форме СДНФ и разработать схему устройства.
2. Проверить схему, для этого собрать устройство в NI Multisim и измерить его таблицу истинности.

Варианты заданий к практической работе

X ₁	X ₂	X ₃	X ₄	Номер варианта											
				1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	1	0	1	1	0	0	0	0	0	0	1	1
0	0	0	1	0	1	1	1	0	1	1	1	1	1	1	0
0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	1
0	0	1	1	1	0	0	0	1	1	1	0	1	1	0	1
0	1	0	0	1	1	1	1	1	0	0	1	0	1	1	1
0	1	0	1	1	1	0	0	0	0	0	1	1	1	0	1
0	1	1	0	0	1	1	0	1	1	0	1	0	1	0	0
0	1	1	1	1	0	0	0	0	1	1	1	1	1	0	1
1	0	0	0	1	1	1	0	1	0	0	0	1	0	1	1
1	0	0	1	0	1	1	0	1	0	0	1	0	0	0	0
1	0	1	0	0	0	0	0	1	1	1	0	0	1	1	0
1	0	1	1	1	0	0	0	0	1	1	0	0	1	0	1
1	1	0	0	1	1	1	1	1	0	1	0	1	0	1	1
1	1	0	1	0	1	0	1	0	1	1	1	0	0	0	0
1	1	1	0	0	0	0	0	1	0	0	1	1	0	0	1
1	1	1	1	1	0	1	1	0	1	1	1	0	1	1	1

Требования к отчету по практической работе

Отчет по практической работе должен содержать:

- заданную таблицу истинности и записанную по ней логическую функцию;
- графическое изображение полученной схемы.

Контрольные вопросы

1. Что называется комбинационной схемой?
2. Какие методы перехода от таблицы задания функции к ее аналитической форме вам известны?

Практическая работа №6. Схемная реализация функций алгебры логики (КНФ)

Цель работы: разработать цифровое устройство, реализующее заданную таблицей истинности функцию алгебры логики (ФАЛ) в форме СКНФ.

Теоретические сведения

1. Аналитические формы записи булевых функций

Рассмотрим аналитическую запись булевых функций и метод перехода от таблицы задания функции к ее аналитическому представлению в форме СКНФ. Он состоит в следующем:

1. Выбрать в таблице задания функций $F(X_1, X_2, \dots, X_n)$ все наборы аргументов, на которых функция обращается в нуль.

2. Для каждого из выбранных наборов выписываются элементарные дизъюнкции аргументов данной функции. При этом если аргумент X_i входит в данный набор как нуль, то он включается в конъюнкцию без изменения, в противном случае в конъюнкцию включается его отрицание $\bar{x}_i$.

3. Все выписанные элементарные дизъюнкции берутся в скобки и объединяются знаками конъюнкции. При этом получаем СКНФ заданной функции.

В качестве примера построим СКНФ функции F :

$$F = (X_1 \cup X_2 \cup X_3)(X_1 \cup \bar{X}_2 \cup X_3)(\bar{X}_1 \cup \bar{X}_2 \cup X_3)(\bar{X}_1 \cup \bar{X}_2 \cup \bar{X}_3).$$

i
j
k
r

Любая булева функция описывает некоторую комбинационную схему (КС) с одним выходом, которую можно реализовать физически на интегральных микросхемах. При этом переход от табличного описания КС к ее аналитическому описанию в виде СКНФ является весьма важным, так как последнее непосредственно задает функциональную схему КС. Например, для нашей СКНФ КС выглядит следующим образом (рис.1):

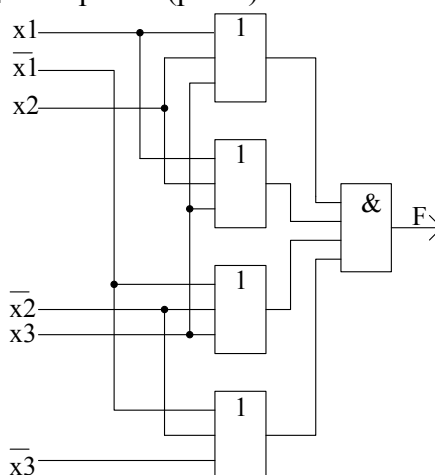


Рис.1. Пример комбинационной схемы.

Однако представление функции в виде СКНФ является в общем случае не оптимальным с точки зрения ее дальнейшей схемной реализации. Как правило, используя законы склеивания и поглощения, аналитическое выражение для F , заданное в виде СКНФ, можно упростить, а это, в свою очередь, приведет к снижению затрат при физической реализации КС. Упростим, например, СКНФ, полученную на рис.1:

$$F = (X_1 \cup X_2)(\bar{X}_1 \cup \bar{X}_3)$$

Реализация этого выражения на схемах ИЛИ и И показана на рис.2.

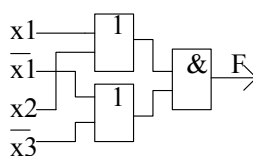


Рис.2. Пример упрощенной КС.

Результаты сравнения показывают, что схема на рис.2 содержит два двухвходовых элемента ИЛИ и один двухвходовой элемент И, а схема на рис.1 реализующая ту же функцию F , содержит четыре трехвходовых элемента ИЛИ и один четырехвходовой элемент И. Следовательно, схема на рис.2 является более экономичной как по количеству элементов, так и по числу входов у этих элементов.

Таким образом, можно сделать вывод о том, что СКНФ не является, как правило, единственно возможной дизъюнктивной (конъюнктивной) формой представления какой-либо булевой функции F . Обычно для этой же функции существуют другие конъюнктивные формы представления, содержащие меньшее число букв, чем СКНФ. Такие формы представления называют «конъюнктивные нормальные формы» (КНФ).

Порядок выполнения работы

1. Для заданной таблицы истинности записать логическую функцию и разработать схему устройства.
2. Проверить схему, для этого собрать устройство в Multisim и измерить его таблицу истинности.

Варианты заданий к практической работе

X_1	X_2	X_3	X_4	Номер варианта											
				1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0
0	0	0	1	1	1	1	1	1	1	0	1	0	0	0	1
0	0	1	0	1	1	0	1	1	1	1	0	1	0	1	1
0	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1
0	1	0	0	0	0	0	1	1	1	0	0	0	1	1	0
0	1	0	1	1	0	0	1	0	1	0	1	1	1	1	0
0	1	1	0	0	0	0	0	1	1	1	0	1	1	1	1
0	1	1	1	1	1	1	0	0	1	1	1	0	1	1	1
1	0	0	0	0	0	0	1	1	0	0	1	0	1	1	0
1	0	0	1	1	1	1	1	1	1	0	1	0	1	1	1
1	0	1	0	1	1	0	1	0	1	1	0	1	0	1	1
1	0	1	1	1	1	1	0	0	1	1	0	1	0	1	1
1	1	0	0	0	0	0	1	0	0	1	1	0	1	0	0
1	1	0	1	1	0	0	1	0	1	1	1	1	1	0	1
1	1	1	0	0	0	0	1	0	1	0	0	1	1	1	1
1	1	1	1	1	1	1	0	0	1	0	1	1	1	1	1

Требования к отчету по практической работе

Отчет по практической работе должен содержать:

- заданную таблицу истинности и записанную по ней логическую функцию;
- графическое изображение полученной схемы.

Контрольные вопросы

1. Что называется комбинационной схемой?
2. Какие методы перехода от таблицы задания функции к ее аналитической форме вам известны?
3. Что называется минимизацией?

Практическая работа №7. Минимизация функций алгебры логики.

Цель работы: разработать цифровые устройства, реализующие заданную таблицей истинности функцию алгебры логики (ФАЛ) в форме МДНФ и МКНФ.

Теоретическая часть

Среди всех возможных для данной функции ДНФ можно найти одну (или несколько), содержащую минимальное число букв. Такая ДНФ называется минимальной (МДНФ).

Таким образом, при синтезе КС (с одним входом) ставится задача получения не просто аналитического выражения для описывающей данную КС булевой функции, а выражения, содержащего минимальное число букв. Как правило, при этом речь идет о МДНФ, поскольку разработанные методы минимизации предназначены для получения именно этих форм. Затем по полученной МДНФ строится требуемая КС с приблизительно минимальным числом элементов. Слово «приблизительно» указывает на то, что имеющийся в нашем распоряжении набор интегральных логически элементов может не совпадать с набором элементов, вытекающим из МДНФ, по числу входов и (или) другим параметрам. В частности, МДНФ дают схему на элементах И, а многие промышленные интегральные серии базируются на элементах И-НЕ.

1. Минимизация булевых функций в форме МДНФ.

Процесс получения МДНФ называют минимизацией. Разработано несколько десятков методов минимизации, позволяющих получить МДНФ либо из ее аналитического выражения для функции, либо из ее таблицы. Для функции от небольшого числа аргументов ($n \leq 6$) наибольшее распространение получил метод минимизации с помощью карт Карно. Рассмотрим далее этот метод минимизации.

Карты Карно для функций от n переменных представляют собой таблицу, состоящую из 2^n клеток. Каждой клетке ставится в соответствие один из возможных наборов аргументов. При этом различным клеткам должны соответствовать различные наборы. В то же время карта строится таким образом, чтобы склеивающиеся между собой элементарные конъюнкции располагались в соседних клетках карты. На рис.3 приведены карты Карно (один из возможных вариантов) соответственно для функций от трех и четырех аргументов.

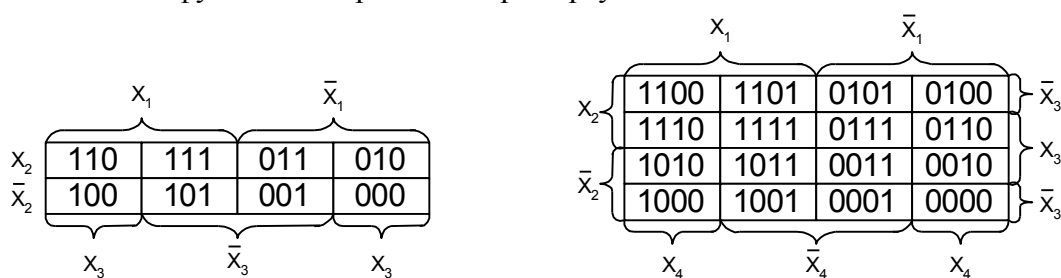


Рис.3. Примеры карт Карно.

Минимизация осуществляется по следующему алгоритму:

1. Значение функции переносится из таблицы в соответствующие клетки карты Карно. При этом, если функция задана в виде ДНФ, ее в начале нужно «развернуть» в СДНФ.
2. Все единицы на карте охватываются контурами. При этом должны соблюдаться следующие правила:

А) Любой контур представляет собой правильный прямоугольник и может охватывать только соседние единицы. Единицы, расположенные на противоположных краях карты, также являются соседними. Одна и та же единица может одновременно входить в различные контуры.

Б) контур может охватывать только 2^n (n - целое) единиц, то есть одну, две, четыре, восемь и т.д. единиц.

В) Контуры проводятся с таким расчетом, чтобы их было как можно меньше и в то же время каждый контур охватывал возможно большее число единиц.

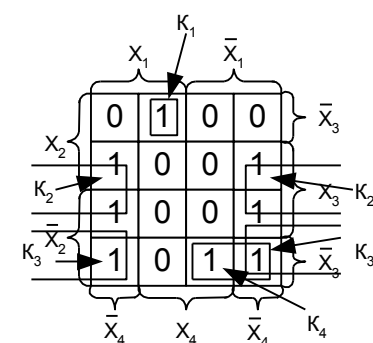
3. Для каждого контура выписывается элементарная конъюнкция, которая составляется по следующему правилу:

Назовем все клетки карты, которые относятся к переменной $\bar{x}_i$, областью определения этой переменной. Тогда переменная $\bar{x}_i$ входит в составляемую конъюнкцию, если данный контур полностью расположен в области ее определения, и не входит в конъюнкцию, если в этой области располагается только половина рассматриваемого контура.

4. Все выписанные конъюнкции объединяются знаками дизъюнкций. При этом получаем ДНФ функции. Если контуры проведены оптимальным образом, то полученная ДНФ представляет собой МДНФ функции.

На рис.4 приведен пример получения с помощью карты Карно МДНФ функции с заданной таблицей.

X ₁	X ₂	X ₃	X ₄	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0



$$F_{\text{МДНФ}} = X_1 X_2 \bar{X}_3 X_4 \cup X_3 \bar{X}_4 \cup \bar{X}_2 \bar{X}_4 \cup \bar{X}_1 \bar{X}_2 \bar{X}_3$$

Рис.4. Пример получения МДНФ функции.

В заключение отметим, что рассмотренные методы минимизации, как и другие существующие, предназначены для разработки КС с одним выходом. Если у схемы несколько выходов, то можно провести минимизацию для каждого выхода в отдельности. Однако в этом случае минимизация оказывается не оптимальной. Разработаны также методы совместной минимизации, предназначенные для КС со многими выходами, однако они слабо формализованы. Соответственно конечный результат при их использовании в основном зависит от опыта и сноровки человека.

Среди всех возможных для данной функции КНФ можно найти одну (или несколько), содержащую минимальное число букв. Такая КНФ называется минимальной (МКНФ).

Таким образом, при синтезе КС (с одним входом) ставится задача получения не просто аналитического выражения для описывающей данную КС булевой функции, а выражения, содержащего минимальное число букв. Как правило, при этом речь идет об МКНФ, поскольку разработанные методы минимизации предназначены для получения именно этих форм. Затем по полученной МКНФ строится требуемая КС с приблизительно минимальным числом элементов. Слово «приблизительно» указывает на то, что имеющийся в нашем распоряжении набор интегральных логических элементов может не совпадать с набором элементов, вытекающим из МКНФ, по числу входов и (или) другим параметрам. В частности, МКНФ дают схему на элементах ИЛИ, а многие промышленные интегральные серии базируются на элементах ИЛИ-НЕ.

2. Минимизация булевых функций в форме МКНФ.

Процесс получения МКНФ называют минимизацией. Разработано несколько десятков методов минимизации, позволяющих получить МКНФ либо из ее аналитического выражения для функции, либо из ее таблицы. Для функции от небольшого числа аргументов ($n \leq 6$) наибольшее распространение получил метод минимизации с помощью карт Карно. Рассмотрим далее этот метод минимизации.

Минимизация осуществляется по следующему алгоритму:

1. Значение функции переносится из таблицы в соответствующие клетки карты Карно. При этом, если функция задана в виде ДНФ (КНФ), ее в начале нужно «развернуть» в СКНФ.

2. Все нули на карте охватываются контурами. При этом должны соблюдаться следующие правила:

А) Любой контур представляет собой правильный прямоугольник и может охватывать только соседние нули. Нули, расположенные на противоположных краях карты, также являются соседними. Один и тот же ноль может одновременно входить в различные контуры.

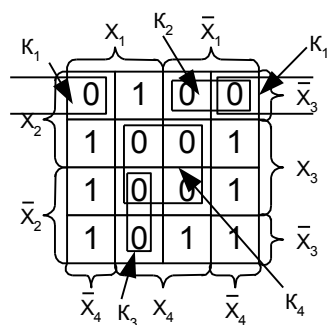
Б) контур может охватывать только 2^n (n - целое) нулей, то есть один, два, четыре, восемь и т.д. нулей.

В) Контур проводится с таким расчетом, чтобы их было как можно меньше и в то же время каждый контур охватывал возможно большее число нулей.

3. Для каждого контура выписывается элементарная дизъюнкция, которая составляется по следующему правилу: если данный контур входит полностью в область определения переменной $\bar{X}_i$, в дизъюнкцию включается инверсия этой переменной, в противном случае инверсия переменной $\bar{X}_i$ в рассматриваемую дизъюнкцию не входит.

4. Все выписанные дизъюнкции объединяются знаками конъюнкций. При этом получаем КНФ функции. Если контуры проведены оптимальным образом, то полученная КНФ представляет собой МкНФ функции.

На рис.3 приведен пример получения с помощью карты Карно МкНФ функции с заданной таблицей.



$$F_{\text{МкНФ}} = (\bar{X}_2 \cup X_3 \cup X_4) \cdot (X_1 \cup \bar{X}_2 \cup X_3) \cdot (\bar{X}_1 \cup X_2 \cup \bar{X}_4) \cdot (\bar{X}_3 \bar{X}_4)$$

Рис.3. Пример получения МкНФ функции.

В заключение отметим, что рассмотренные методы минимизации, как и другие существующие, предназначены для разработки КС с одним выходом. Если у схемы несколько выходов, то можно провести минимизацию для каждого выхода в отдельности. Однако в этом случае минимизация оказывается не оптимальной. Разработаны также методы совместной минимизации, предназначенные для КС со многими выходами, однако они слабо формализованы. Соответственно конечный результат при их использовании в основном зависит от опыта и сноровки человека.

2. Задание на практическую работу

1. Выполнить минимизацию логической функции.
2. Проверить результаты минимизации, для этого собрать устройство в Multisim и измерить его таблицу истинности.

X ₁	X ₂	X ₃	X ₄	Номер варианта											
				1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	0
0	0	0	1	1	1	1	1	1	1	0	1	0	0	0	1
0	0	1	0	1	1	0	1	1	1	1	0	1	0	1	1
0	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1
0	1	0	0	0	0	0	1	1	1	0	0	0	1	1	0
0	1	0	1	1	0	0	1	0	1	0	1	1	1	1	0
0	1	1	0	0	0	0	0	1	1	1	0	1	1	1	1
0	1	1	1	1	1	1	0	0	1	1	1	0	1	1	1
1	0	0	0	0	0	0	1	1	0	0	1	0	1	1	0
1	0	0	1	1	1	1	1	1	1	0	1	0	1	1	1
1	0	1	0	1	1	0	1	0	1	1	0	1	0	1	1
1	0	1	1	1	1	1	0	0	1	1	0	1	0	1	1
1	1	0	0	0	0	0	1	0	0	1	1	0	1	0	0
1	1	0	1	1	0	0	1	0	1	1	1	1	1	0	1

1	1	1	0	0	0	0	1	0	1	0	0	1	1	1	1
1	1	1	1	1	1	1	0	0	1	0	1	1	1	1	1

Практическая работа №8. Проектирование одновыходной комбинационной схемы

Цель работы: ознакомиться с методикой и получить практические навыки синтеза комбинационных схем на интегральных микросхемах малой степени интеграции.

1. Минимизация булевых функций в базисах И – НЕ и ИЛИ – НЕ

Как уже отмечалось выше, не всегда в нашем распоряжении имеются микросхемы, соответствующие логическим функциям И и ИЛИ, задаваемым МДНФ (МКНФ). В частности, в данной практической работе используются микросхемы серий 155, 1533 (см. рис.1), выполненные в базисе И – НЕ, ИЛИ – НЕ.

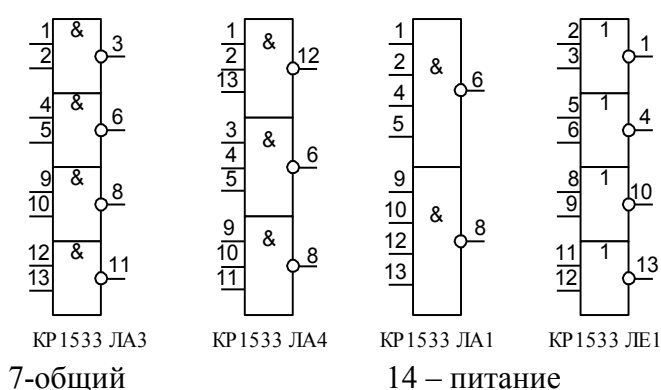


Рис. 1. Некоторые микросхемы серии КР1533

Возникает проблема перехода от базиса И, ИЛИ к базису И-НЕ. Решение этой проблемы основывается на хорошо известном законе Де-Моргана:

$$X \cup Y \cup \dots \cup Z = \overline{\overline{X} \cdot \overline{Y} \cdot \dots \cdot \overline{Z}}$$

или

$$X \cdot Y \cdot \dots \cdot Z = \overline{\overline{X} \cup \overline{Y} \cup \dots \cup \overline{Z}}$$

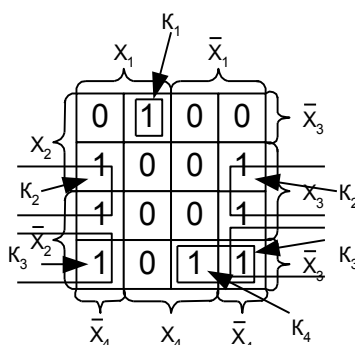
Исходя из этого закона можно предложить два простых, похожих друг на друга алгоритма получения минимизированного аналитического выражения, пригодного для реализации в базисах И-НЕ и ИЛИ-НЕ.

Первый из них заключается в следующем

1. Находим МКНФ функции с помощью карты Карно.
2. В МДНФ над всеми элементарными конъюнкциями ставят знаки инверсии.
3. Заменяем в полученном выражении все знаки дизъюнкции на знаки конъюнкции.
4. Ставим знак инверсии над всем полученным выражением. В итоге получаем выражение, удобное для реализации на схемах И-НЕ.

Например, воспользуемся этим алгоритмом для следующей функции:

X ₁	X ₂	X ₃	X ₄	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0



1	0	1	0	1	$F_{\text{мднф}} = X_1 X_2 \bar{X}_3 X_4 \cup X_3 \bar{X}_4 \cup \bar{X}_2 \bar{X}_4 \cup \bar{X}_1 \bar{X}_2 \bar{X}_3$
1	0	1	1	0	
1	1	0	0	0	
1	1	0	1	1	
1	1	1	0	1	
1	1	1	1	0	

1 пункт: $F_{\text{мднф}} = X_1 X_2 \bar{X}_3 X_4 \cup X_3 \bar{X}_4 \cup \bar{X}_2 \bar{X}_4 \cup \bar{X}_1 \bar{X}_2 \bar{X}_3$

2 пункт: $\overline{X_1 X_2 \bar{X}_3 X_4} \cup \overline{X_3 \bar{X}_4} \cup \overline{\bar{X}_2 \bar{X}_4} \cup \overline{\bar{X}_1 \bar{X}_2 \bar{X}_3}$

3 пункт: $\overline{X_1 X_2 \bar{X}_3 X_4} \cdot \overline{X_3 \bar{X}_4} \cdot \overline{\bar{X}_2 \bar{X}_4} \cdot \overline{\bar{X}_1 \bar{X}_2 \bar{X}_3}$

4 пункт: $\overline{X_1 X_2 \bar{X}_3 X_4} \cdot \overline{X_3 \bar{X}_4} \cdot \overline{\bar{X}_2 \bar{X}_4} \cdot \overline{\bar{X}_1 \bar{X}_2 \bar{X}_3} = F_{\text{И-НЕ}}$

Схемная реализация функции $F_{\text{И-НЕ}}$ показана на рис.2.

Как видно из рис.2, для реализации потребовалась одна микросхема ЛА4 и одна микросхема ЛА1. Можно было бы конечно, как вытекает из реализуемого выражения, взять в качестве элементов 3 и 4 схемы 2И-НЕ, однако при этом нам очевидно пришлось бы использовать третью микросхему (ЛА3), что было бы не оптимально.

Второй алгоритм заключается в следующем:

1. Находим МКНФ функции;
2. В МКНФ над всеми элементарными дизъюнкциями ставим знаки инверсии.
3. Заменяем в полученном выражении все знаки конъюнкции на знаки дизъюнкции.
4. Ставим знак инверсии над всем полученным выражением. В итоге получаем выражение, удобное для реализации в базисе ИЛИ-НЕ.

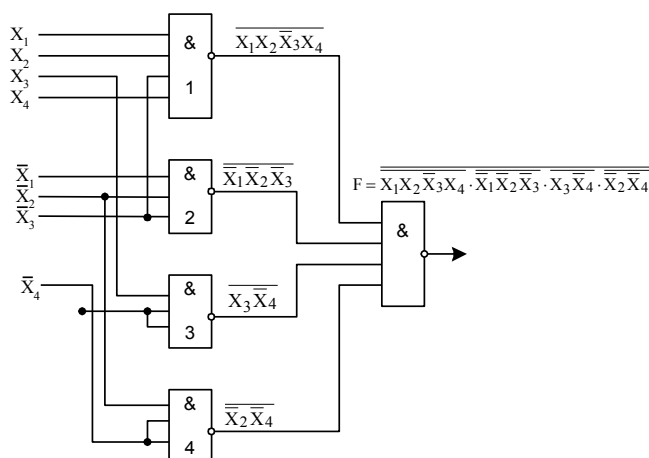
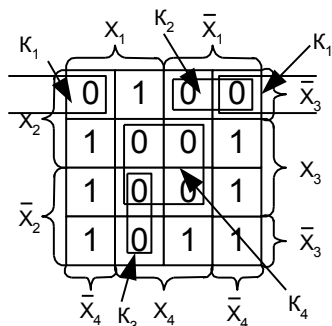


Рис. 2. Схемная реализация функции $F_{\text{И-НЕ}}$

Возьмём для примера МКНФ:



$$F_{\text{МКНФ}} = (\bar{X}_2 \cup X_3 \cup X_4) \cdot (X_1 \cup \bar{X}_2 \cup X_3) \cdot (\bar{X}_1 \cup X_2 \cup \bar{X}_4) \cdot (\bar{X}_3 \bar{X}_4)$$

- 1 пункт: $F_{\text{МКНФ}} = (\bar{X}_2 \cup X_3 \cup X_4) \cdot (X_1 \cup \bar{X}_2 \cup X_3) \cdot (\bar{X}_1 \cup X_2 \cup \bar{X}_4) \cdot (\bar{X}_3 \bar{X}_4)$
- 2 пункт: $\overline{\bar{X}_2 \cup X_3 \cup X_4} \cdot \overline{X_1 \cup \bar{X}_2 \cup X_3} \cdot \overline{\bar{X}_1 \cup X_2 \cup \bar{X}_4} \cdot \overline{\bar{X}_3 \bar{X}_4}$
- 3 пункт: $\overline{\bar{X}_2 \cup X_3 \cup X_4 \cup X_1 \cup \bar{X}_2 \cup X_3 \cup \bar{X}_1 \cup X_2 \cup \bar{X}_4 \cup \bar{X}_3 \bar{X}_4}$
- 4 пункт: $\overline{\overline{\bar{X}_2 \cup X_3 \cup X_4 \cup X_1 \cup \bar{X}_2 \cup X_3 \cup \bar{X}_1 \cup X_2 \cup \bar{X}_4 \cup \bar{X}_3 \bar{X}_4}} = F_{\text{ИЛИ-НЕ}}$

Из выражения видно, что его непосредственная реализация на имеющихся в нашем распоряжении микросхемах (рис. 1) не представляется возможной. Действительно, для его реализации нужны элементы 3ИЛИ-НЕ и 4ИЛИ-НЕ, а у нас есть только элементы 2ИЛИ-НЕ. Используя закон Де-Моргана, мы можем преобразовать полученное выражение к приемлемой для нас форме. Например учитывая, что мы можем реализовать это выражение на элементе 3И-НЕ (ЛА4) и т.д.

2. Задание на практическую работу

Минимизировать заданную функцию, используя метод диаграмм Вейча, реализовать функцию на заданных микросхемах.

X ₁	X ₂	X ₃	X ₄	Номер варианта											
				1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	1	0	1	1	0	0	0	0	0	0	1	1
0	0	0	1	0	1	1	0	0	1	1	1	1	1	1	0
0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	1
0	0	1	1	1	0	0	0	1	1	1	0	1	1	0	0
0	1	0	0	1	1	1	1	1	0	0	1	0	0	1	1
0	1	0	1	1	1	0	0	0	0	1	1	1	1	0	1
0	1	1	0	0	1	1	0	1	1	0	1	0	1	0	0
0	1	1	1	1	1	0	0	0	1	1	1	1	1	0	1
1	0	0	0	0	1	1	0	1	0	0	0	1	0	1	1
1	0	0	1	0	1	1	0	1	0	0	1	0	0	0	1
1	0	1	0	0	0	0	0	1	1	1	0	0	1	1	0
1	0	1	1	1	0	0	0	0	1	1	0	0	1	0	1
1	1	0	0	1	1	1	1	1	1	1	0	1	0	1	1
1	1	0	1	0	1	0	1	0	1	1	1	0	0	0	0
1	1	1	0	0	0	0	0	1	0	0	1	1	0	0	1
1	1	1	1	1	0	1	1	0	1	1	1	0	1	1	1

3. Требования к отчету по практической работе

Отчет по практической работе должен содержать:

- исходные данные задания;
- синтез комбинационной схемы;
- графическое изображение полученной схемы.;

4. Контрольные вопросы

1. Что называется комбинационной схемой?
2. Какие методы перехода от таблицы задания функции к ее аналитической форме вам известны?
3. Что называется минимизацией?
4. Как осуществляется минимизация в базисах И-НЕ и ИЛИ-НЕ?

Практическая работа №9. Расчет динамических параметров схемы

Цель работы: получить практические навыки синтеза комбинационных схем на интегральных микросхемах малой степени интеграции, научиться рассчитывать динамические параметры.

1. Проектирование одновыходной комбинационной схемы на элементах И-НЕ

Проектирование одновыходной комбинационной схемы на элементах И-НЕ состоит из следующих шагов:

- получение минимальной ДНФ переключательной функции,
- представление полученной минимальной формы в базисе Шеффера,
- построение по полученному выражению комбинационной схемы.

Пусть задана функция F_1 от четырех переменных в виде десятичных номеров конstituент совершенной дизъюнктивной нормальной формы, т.е. тех наборов, на которых функция равна единице:

$$F_1(x_3, x_2, x_1, x_0) = \Sigma(0, 2, 4, 6, 10, 12, 14, 15).$$

Эту функцию необходимо реализовать на элементах И-НЕ. Запишем эту функцию в виде логического выражения:

$$F_1(x_3, x_2, x_1, x_0) = (\bar{x}_3 \bar{x}_2 \bar{x}_1 \bar{x}_0) \vee (\bar{x}_3 \bar{x}_2 x_1 \bar{x}_0) \vee (\bar{x}_3 x_2 \bar{x}_1 \bar{x}_0) \vee (\bar{x}_3 x_2 x_1 \bar{x}_0) \vee (x_3 \bar{x}_2 x_1 \bar{x}_0) \vee (x_3 x_2 \bar{x}_1 \bar{x}_0) \vee (x_3 x_2 x_1 \bar{x}_0) \vee (x_3 x_2 x_1 x_0).$$

Произведем минимизацию заданной функции, используя метод диаграмм Вейча (рис.1).

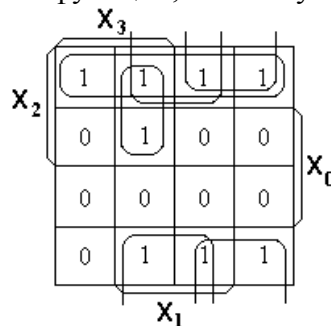


Рис. 1. Минимизация функции при помощи диаграмм Вейча.

Получим следующее минимальное выражение функции:

$$F_{1, \min}(x_3, x_2, x_1, x_0) = x_2 \bar{x}_0 \vee x_1 \bar{x}_0 \vee \bar{x}_3 \bar{x}_0 \vee x_3 x_2 x_1. \quad (1)$$

Преобразуем полученное выражение (1) к виду, удобному для реализации на элементах И-НЕ, т.е. запишем его в базисе Шеффера.

Преобразование выполняют по следующим правилам:

- все знаки конъюнкции и дизъюнкции заменяют на знаки «штрих Шеффера»;
- переменные, входящие в одну импликанту, заключают в скобки;
- однобуквенные импликанты инвертируют;
- исходное выражение, состоящее только из одной импликанты с числом букв 2 и более, после замены конъюнкции на «штрих Шеффера» инвертируют.

Примечание. Обычно легко запоминают первые два пункта правила перехода от ДНФ к функции Шеффера (это наиболее типичный случай) и забывают о частных случаях, отраженных в двух последних пунктах. Проиллюстрируем эти случаи.

Пример 1. Пусть полученное минимальное выражение:

$$F(a, b, c) = \bar{a} \vee b \bar{c}.$$

Тогда преобразованное выражение будет выглядеть так:

$$F(a, b, c) = \bar{\bar{a}} | (b | \bar{c}) = a | (b | \bar{c}).$$

Пример 2. Допустим, исходное выражение имеет вид:

$$F(a, b, c) = \bar{a} b \bar{c}.$$

После преобразования выражение будет выглядеть следующим образом:

$$F(a, b, c) = \bar{\bar{a}} | b | \bar{\bar{c}} = a | b | \bar{c}.$$

В нашем примере получим:

$$F_1(x_3 x_2 x_1 x_0) = (x_2 | \bar{x}_0) | (x_1 | \bar{x}_0) | (\bar{x}_3 | \bar{x}_0) | (x_3 | x_2 | x_1). \quad (2)$$

Построение комбинационной схемы выполняют в следующей последовательности:

- вначале с помощью элементов НЕ получают инверсные значения переменных;
- затем, используя элементы И-НЕ, реализуют члены логической функции, заключенные в скобки;
- наконец, выходы элементов И-НЕ, использованных в п. б), подают на входы результирующего элемента И-НЕ.

Приведенная последовательность построения комбинационной схемы по полученному выражению (2) показана на рис. 2. Вначале с помощью инверторов $D1$ и $D2$ получают инверсные значения переменных x_0 и x_3 . Затем, используя элементы $D3 - D6$, реализуют члены логической функции, заключенные в скобки. Наконец, с помощью элемента $D7$ завершают построение схемы.

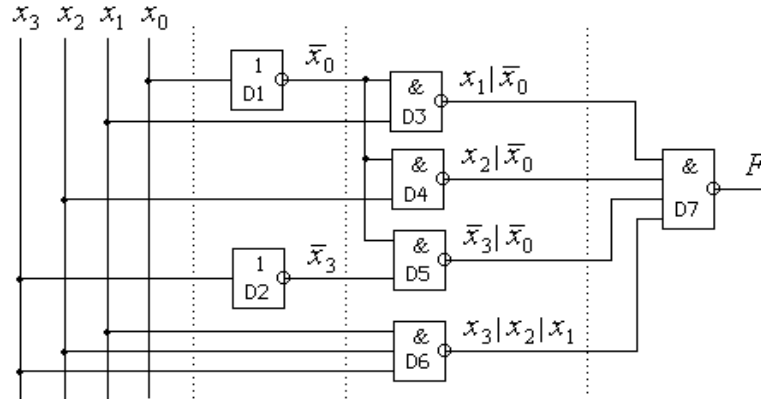


Рис. 2. Реализация комбинационной схемы на элементах И-НЕ

Временная диаграмма работы спроектированной комбинационной схемы приведена на рис.

3.

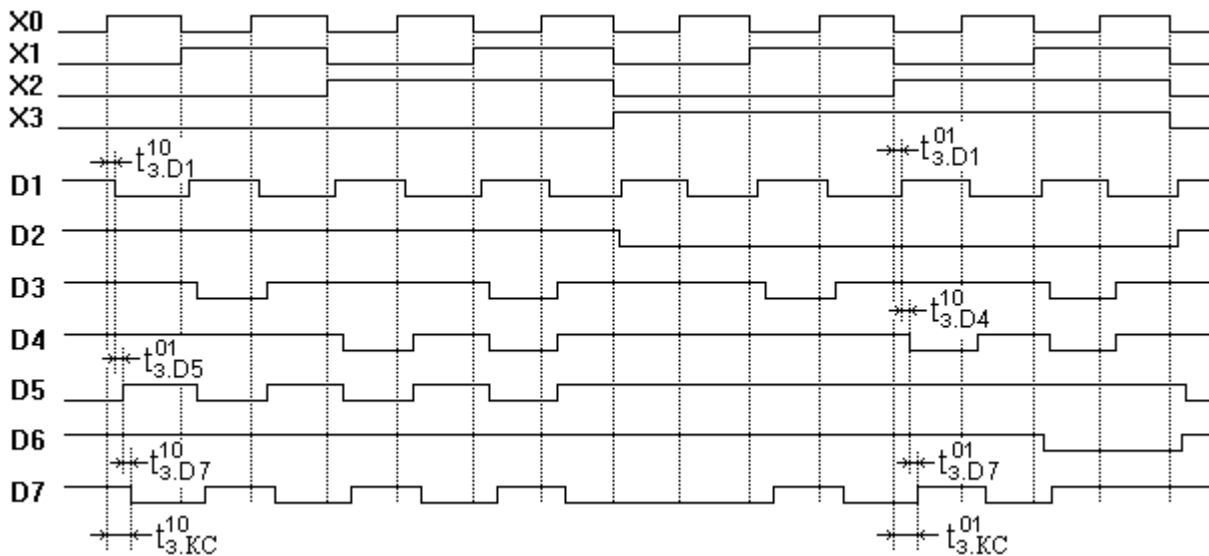


Рис. 3. Временная диаграмма работы комбинационной схемы

Из ее рассмотрения можно сделать вывод о соответствии функционирования схемы заданному закону. На вход схемы поступает последовательность сигналов, задающая все 16 наборов, на которых определена функция. Из рис. 3 видно, что выход схемы принимает значение 1 на наборах 0, 2, 4, 6, 10, 12, 14 и 15, т.е. спроектированная схема правильно реализует заданную функцию.

Динамические параметры комбинационной схемы

К динамическим параметрам комбинационной схемы относят задержки переключения выходного сигнала из 0 в 1 и из 1 в 0.

Данные задержки определяют по временной диаграмме в такой последовательности:

- предварительно на схеме находят наиболее длинный путь (или пути) прохождения сигнала от входа до выхода (длина пути определяется числом *последовательно* соединенных элементов);
- затем на временной диаграмме определяют входные состояния, переход к которым вызывает последовательное переключение всех элементов в найденном пути. Переход к одному из этих

состояний должен вызывать переключение выходного сигнала из 0 в 1, переход к другому - переключение выходного сигнала из 1 в 0;

- после чего записывают выражения для подсчета задержек переключения выходного сигнала из 0 в 1 и из 1 в 0. Выражения представляют собой сумму соответствующих задержек логических элементов, входящих в путь;
- наконец, определяют числовое значение задержек комбинационной схемы, подставляя в найденные выражения значения задержек логических элементов.

Для спроектированной комбинационной схемы (см. рис. 2) наиболее длинный путь содержит три элемента. Таких путей в схеме несколько: $D1 \rightarrow D3 \rightarrow D7$, $D1 \rightarrow D4 \rightarrow D7$, $D1 \rightarrow D5 \rightarrow D7$ и $D2 \rightarrow D6 \rightarrow D7$.

На временной диаграмме существует несколько состояний, переход к которым активизирует некоторые из этих путей. Можно выбрать любые из них для оценки задержек переключения комбинационной схемы. Два перехода с обозначением задержек элементов показаны на рис. 3.

Непосредственно из рис. 3 можно записать выражения для задержек переключения выходного сигнала комбинационной схемы:

$$t_{3.KC}^{01} = t_{3.D1}^{01} + t_{3.D4}^{10} + t_{3.D7}^{01},$$

$$t_{3.KC}^{10} = t_{3.D1}^{10} + t_{3.D5}^{01} + t_{3.D7}^{10}.$$

Подставляя в найденные выражения паспортные значения задержек логических элементов, получим числовое значение задержек комбинационной схемы.

Задание на практическую работу

Минимизировать заданную функцию, используя метод диаграмм Вейча. Записать полученное выражение в базисе Шеффера, реализовать функцию на заданных микросхемах и построить временную диаграмму работы комбинационной схемы. Рассчитать динамические параметры схемы.

X ₁	X ₂	X ₃	X ₄	Номер варианта											
				1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	1	0	1	1	0	0	0	0	0	0	1	1
0	0	0	1	0	1	1	0	0	1	1	1	1	1	1	0
0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	1
0	0	1	1	1	0	0	0	1	1	1	0	1	1	0	0
0	1	0	0	1	1	1	1	1	0	0	1	0	0	1	1
0	1	0	1	1	1	0	0	0	0	1	1	1	1	0	1
0	1	1	0	0	1	1	0	1	1	0	1	0	1	0	0
0	1	1	1	1	1	0	0	0	1	1	1	1	1	0	1
1	0	0	0	0	1	1	0	1	0	0	0	1	0	1	1
1	0	0	1	0	1	1	0	1	0	0	1	0	0	0	1
1	0	1	0	0	0	0	0	1	1	1	0	0	1	1	0
1	0	1	1	1	0	0	0	0	1	1	0	0	1	0	1
1	1	0	0	1	1	1	1	1	1	1	0	1	0	1	1
1	1	0	1	0	1	0	1	0	1	1	1	0	0	0	0
1	1	1	0	0	0	0	0	1	0	0	1	1	0	0	1
1	1	1	1	1	0	1	1	0	1	1	1	0	1	1	1

4. Требования к отчету по практической работе

Отчет по практической работе должен содержать:

- исходные данные задания;
- синтез комбинационной схемы;
- графическое изображение полученной схемы;
- временную диаграмму работы схемы;
- расчет динамических параметров схемы.

Контрольные вопросы

1. Что называется комбинационной схемой?
2. Какие методы перехода от таблицы задания функции к ее аналитической форме вам известны?
3. Что называется минимизацией?
4. Как осуществляется минимизация в базисах И-НЕ и ИЛИ-НЕ?

Практическая работа № 10. Синтез дешифраторов

Цель работы: ознакомиться с методикой и получить практические навыки синтеза дешифраторов на интегральных микросхемах малой степени интеграции.

Краткие теоретические сведения и методические указания

В вычислительной технике многие часто используемые комбинационные схемы выпускаются в виде отдельных микросхем. К таким схемам относятся дешифраторы.

Дешифратором называется комбинационная схема, имеющая в общем случае n информационных входов и 2^n информационных выходов. Работу такого дешифратора можно описать системой уравнений (здесь X – входы, а Y – выходы)

$$Y_0 = \bar{X}_0 \& \bar{X}_1 \& \dots \& \bar{X}_{n-2} \& \bar{X}_{n-1}$$

$$Y_1 = X_0 \& \bar{X}_1 \& \dots \& \bar{X}_{n-2} \& \bar{X}_{n-1}$$

$$Y_2 = \bar{X}_0 \& X_1 \& \dots \& \bar{X}_{n-2} \& \bar{X}_{n-1}$$

$$\dots \dots \dots$$

$$Y_{2^n-1} = X_0 \& X_1 \& \dots \& X_{n-2} \& X_{n-1}$$

Иначе говоря, при подаче на входы дешифратора любой комбинации сигналов на одном из выходов дешифратора (соответствующем этой комбинации) появится активный сигнал. На рис.1 приводится условное обозначение дешифратора на два входа и таблица, описывающая его работу.

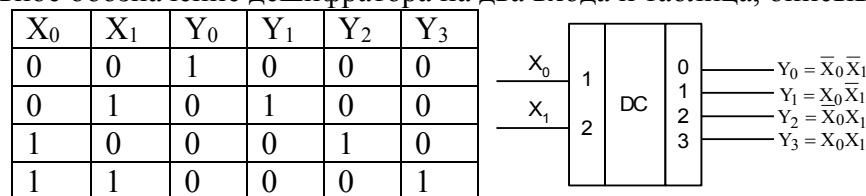


Рис.1. Пример функциональной схемы дешифратора.

Недостатком рассмотренного дешифратора является то, что в любой момент времени на одном из выходов стоит активный сигнал. Чтобы устранить этот недостаток, в состав дешифратора вводится управляющий вход E . Если сигнал на этом входе активен, то работа дешифратора совпадает с описанной ранее. В противном случае все выходы дешифратора находятся в пассивном состоянии. Таким образом, вход E осуществляет функцию стробирования дешифратора. В промышленно выпускаемых дешифраторах обычно имеется сразу несколько управляющих входов, объединенных по схеме И. Это представляет разработчику дополнительные удобства.

На рис.2 показан пример увеличения размерности дешифратора с использованием двух управляющих входов E .

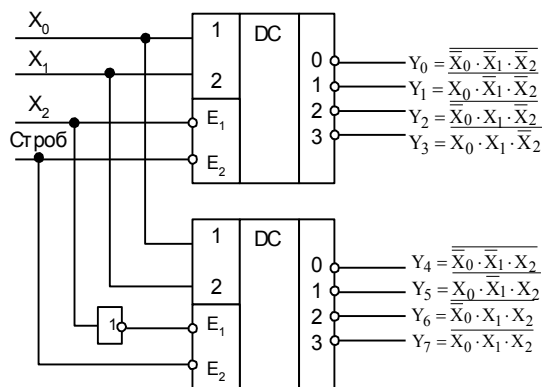


Рис.2. Пример увеличения размерности дешифратора.

На основе дешифраторов на n входов можно легко реализовать произвольные функции алгебры логики для n и менее переменных. Например, на рис.3 показана реализация функции, заданной таблицей.

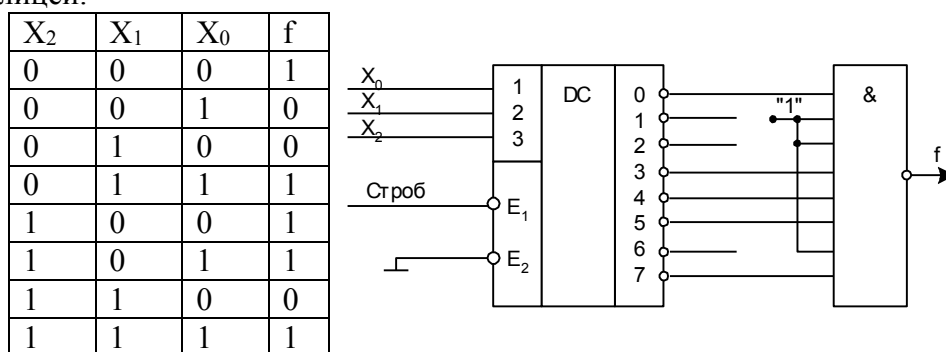


Рис.3. Пример реализации логической функции с использованием дешифратора.

Строблируемые дешифраторы можно использовать в качестве селекторов (демультиплексоров). Селектором в вычислительной технике называется комбинационная схема, осуществляющая передачу информации с единственного информационного входа на один из нескольких выходов. Селекторы не выпускаются в качестве самостоятельных микросхем, так как их функции легко реализуются дешифраторами. Управляющий вход (или один из них) дешифратора в этом случае рассматривается как информационный, а информационные входы задают номер выхода, на который и выдается информация, т.е. в этом случае информационные входы используются в качестве управляющих. На рис.4 показано использование дешифратора на 2 входа в качестве селектора.

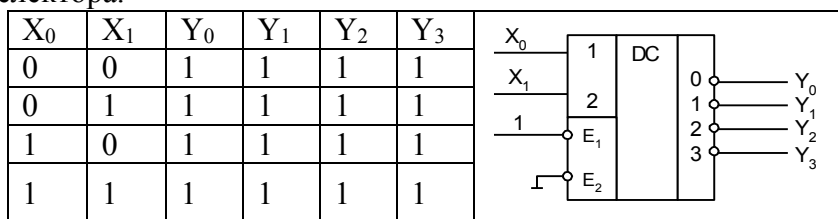


Рис.4. Пример использования дешифратора в качестве селектора.

В практической работе предполагается использовать дешифратор КР1533ИД4 (рис.5). Это двойной дешифратор, т.е. в корпусе реализуются два отдельных дешифратора, имеющих независимые выходы ($Y_{0÷3}$ и $Z_{0÷3}$) и независимые управляющие входы ($E_{1÷2}$ и $G_{1÷2}$). Однако оба дешифратора имеют одни и те же информационные входы $X_{0÷1}$.

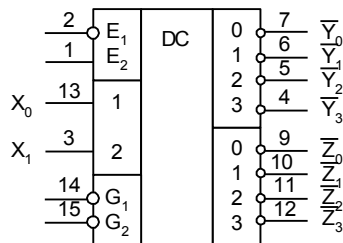


Рис.5. Микросхема дешифратора КР1533ИД4.

Варианты заданий на практическую работу

В соответствии с номером варианта согласно таблице 1 построить комбинационную схему на ИМС дешифратора.

Таблица 1

№ вар	Задание
1.	Реализовать на ИД4 и дополнительной логике функцию $F = \overline{X_1} \cup \overline{X_2} \oplus X_3 \cup X_2 \overline{X_4}$
2.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \cdot \overline{X_2} \cdot (X_2 \oplus \overline{X_3}) \cup (\overline{X_1} X_2 X_3)$
3.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \cup X_2 \cup X_3 \cdot X_4$
4.	Реализовать на ИД4 и дополнительной логике функцию $F = \overline{X_1} X_2 \cup \overline{X_2} X_3 \cup X_1 X_2 \overline{X_4}$
5.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 X_2 \cup (X_1 \oplus X_2 \oplus X_3)$
6.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \oplus \overline{X_2} \cup \overline{X_3} X_4$
7.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \overline{X_2} (X_2 \oplus \overline{X_3}) \cup \overline{X_1} X_2 X_3$
8.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 X_2 \cup X_2 X_3 \cup X_3 X_4 X_5$
9.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \oplus X_2 \oplus X_3 \oplus X_4$
10.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \cdot \overline{X_4} \cdot (X_2 \cup \overline{X_3}) \oplus (X_1 \overline{X_2} X_3)$
11.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \cdot X_2 \cdot X_3 \cup \overline{X_4} \overline{X_5}$
12.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 X_2 X_3 \oplus \overline{X_1} \overline{X_2} \overline{X_3}$
13.	Реализовать на ИД4 и дополнительной логике функцию $F = \overline{X_1} \cup X_3 \cdot (X_2 \cup \overline{X_4}) \cup (X_1 X_2 \oplus \overline{X_3})$
14.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \oplus X_2 \cup X_2 \oplus \overline{X_3} \cup X_3 X_4$
15.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \cup X_2 \oplus X_3 \cup \overline{X_4}$
16.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 X_2 \cup X_3 \oplus \overline{X_1} X_2 \overline{X_3}$
17.	Реализовать на ИД4 и дополнительной логике функцию $F = \overline{X_1} X_2 \oplus (X_1 X_2 \cup \overline{X_3})$
18.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \overline{X_2} \oplus \overline{X_3} \cup X_4$
19.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \cup X_2 X_3 \oplus \overline{X_1} X_3$
20.	Реализовать на ИД4 и дополнительной логике функцию $F = X_1 \cdot X_2 \oplus X_3 \cup \overline{X_2} X_4$

Порядок выполнения работы.

1. Выбрать из таблицы задание, согласно варианту.
2. Составить таблицу истинности для заданной функции.
3. Реализовать заданную функцию на дешифраторе.
4. Проверить правильность функционирования.

Содержание отчета

Отчет по практической работе должен содержать:

1. Исходное задание.
2. Реализацию задания в виде функциональной схемы.
3. Краткое описание применяемой микросхемы.

Контрольные вопросы

1. Что представляет собой дешифратор?
2. Как можно увеличить размерность дешифратора?
3. Что представляет собой селектор?

Практическая работа №11. Каскадное соединение дешифраторов

Цель работы: ознакомиться с методикой и получить практические навыки синтеза дешифраторов на интегральных микросхемах малой степени интеграции.

Краткие теоретические сведения и методические указания

Необходимость каскадного соединения нескольких дешифраторов возникает в том случае, когда разрядность одной ИС оказывается недостаточной для адресации большого количества различных устройств. Например, в нашем распоряжении схемы дешифратора на 2 входа и, соответственно, 4 выхода. А необходимо организовать возможность обращения к 16 цифровым устройствам. Очевидно, что для построения такой схемы понадобятся 4 ИС дешифратора указанной разрядности (выходная часть схемы на рис. 1). Первый из них будет обеспечивать выдачу сигналов $Q_0 \dots Q_3$, второй – $Q_4 \dots Q_7$, третий – $Q_8 \dots Q_{11}$ и последний, четвёртый – $Q_{12} \dots Q_{15}$. Для обеспечения выбора этих 16 выходов необходимы 4 входных сигнала – a_8, a_4, a_2 и a_1 . Два младших из них (a_2 и a_1) подаются на все дешифраторы одновременно. Два старших (a_8 и a_4) подаются на пятый дешифратор, служащий для обеспечения выбора одного из четырёх дешифраторов выходной части схемы.

Принцип работы каскада тот же, что и у отдельно взятого дешифратора - он выдаёт активный сигнал только на одном выходе. Номер этого выхода соответствует двоичному коду, поданному на входные линии. Например, при подаче кода 1110_2 (на рис. 1 показано красным цветом) будет работать только четвёртый дешифратор выходной очереди каскада, на разрешающие входы остальных дешифраторов подаётся логический ноль. Следовательно, на выходах схемы $Q_1 \dots Q_{11}$ будут сформированы логические нули. И только на активном, четвёртом дешифраторе формируется унарный код - на выходе "2" логическая единица соответствующая коду $10_2 = 2_{10}$ на входе этого дешифратора. Таким образом, на выходе каскада Q_{14} будет логическая единица, соответствующая коду $1110_2 = 14_{10}$.

На рис. 1 каскад построен на одинаковых дешифраторах. Но возможен синтез схем, когда выбирающий дешифратор на входе схемы будет другой разрядности. Рассмотрим каскад также на 16 выходов, но построенный на базе дешифраторов на 3 входа. В этом случае выходная очередь каскада содержит два дешифратора (рис. 2), на входы которых подаются сигналы с трёх младших входных линий a_4, a_2 и a_1 . Сигнал со старшей линии a_8 подключается к единственной входной линии управляющего дешифратора на входе схемы. Он определяет, какой из двух дешифраторов будет активен. Так, если подан тот же код 1110_2 (показан на рис. 2 красным цветом), что и в предыдущем примере, то выбирается нижний, второй дешифратор. Следовательно, на выходах верхнего дешифратора (выходы каскада $Q_0 \dots Q_7$ будут логические нули. Активный нижний дешифратор сформирует активный сигнал логической единицы только на одном выходе "6", т.е. на выходе каскада Q_{14} будет логическая единица, соответствующая входному коду $1110_2 = 14_{10}$.

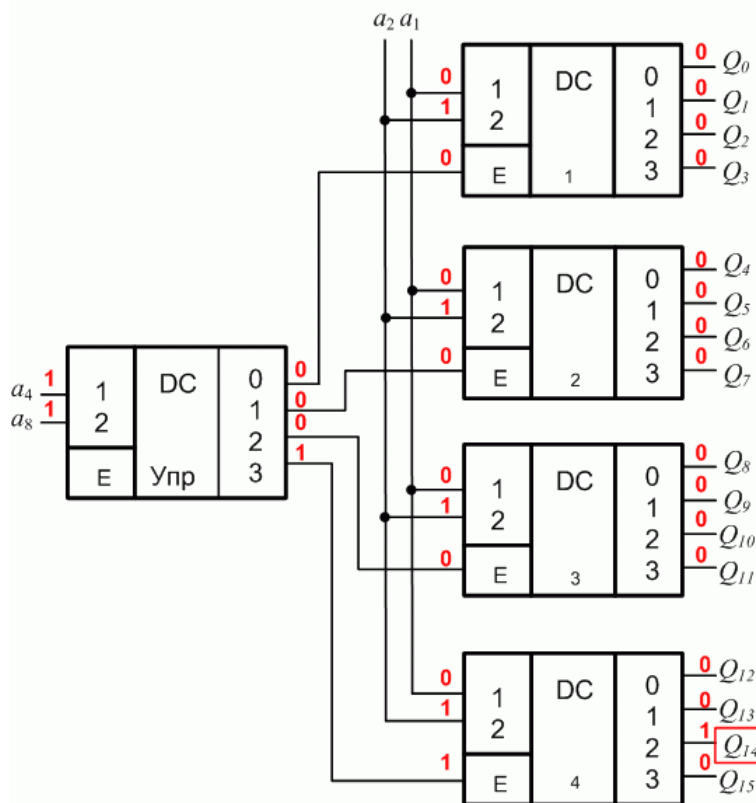


Рисунок 1. Каскад дешифраторов на 16 выходов на базе дешифраторов на 2 входа

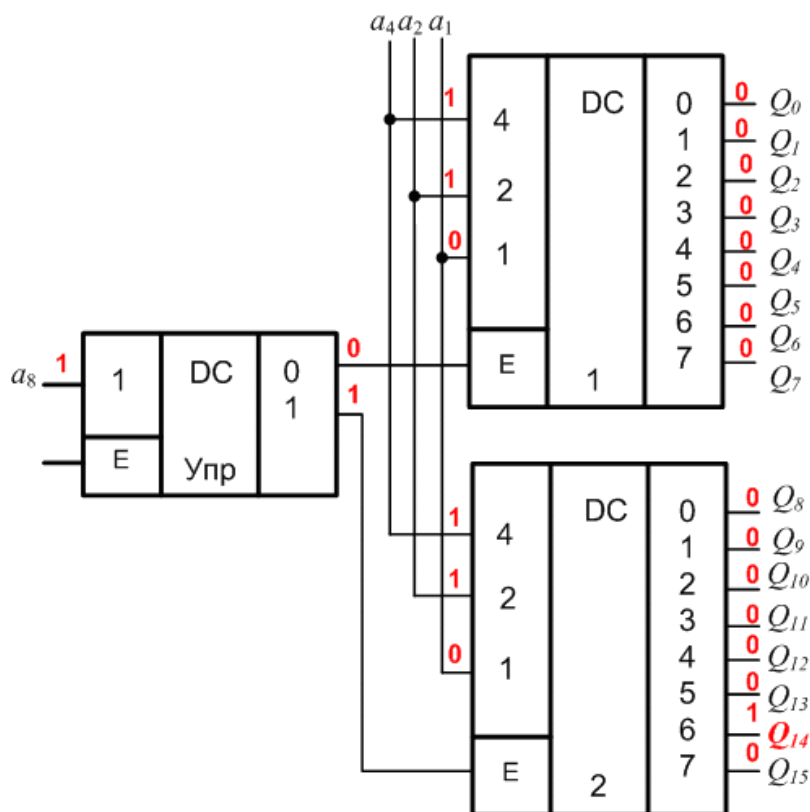


Рисунок 2. Каскад дешифраторов на 16 выходов на базе дешифраторов на 3 входа

Рассмотрим пример с большей разрядностью. Необходимо синтезировать схему дешифратора на 7 входов на базе 4-входовых дешифраторов.

У этой схемы должно быть $2^7 = 128$ выходных сигналов. Один базовый 4-входовой дешифратор обеспечивает наличие $2^4 = 16$ выходов. Поэтому в выходном каскаде схемы должно быть $128:16 = 8$ базовых дешифраторов (рис. 3).

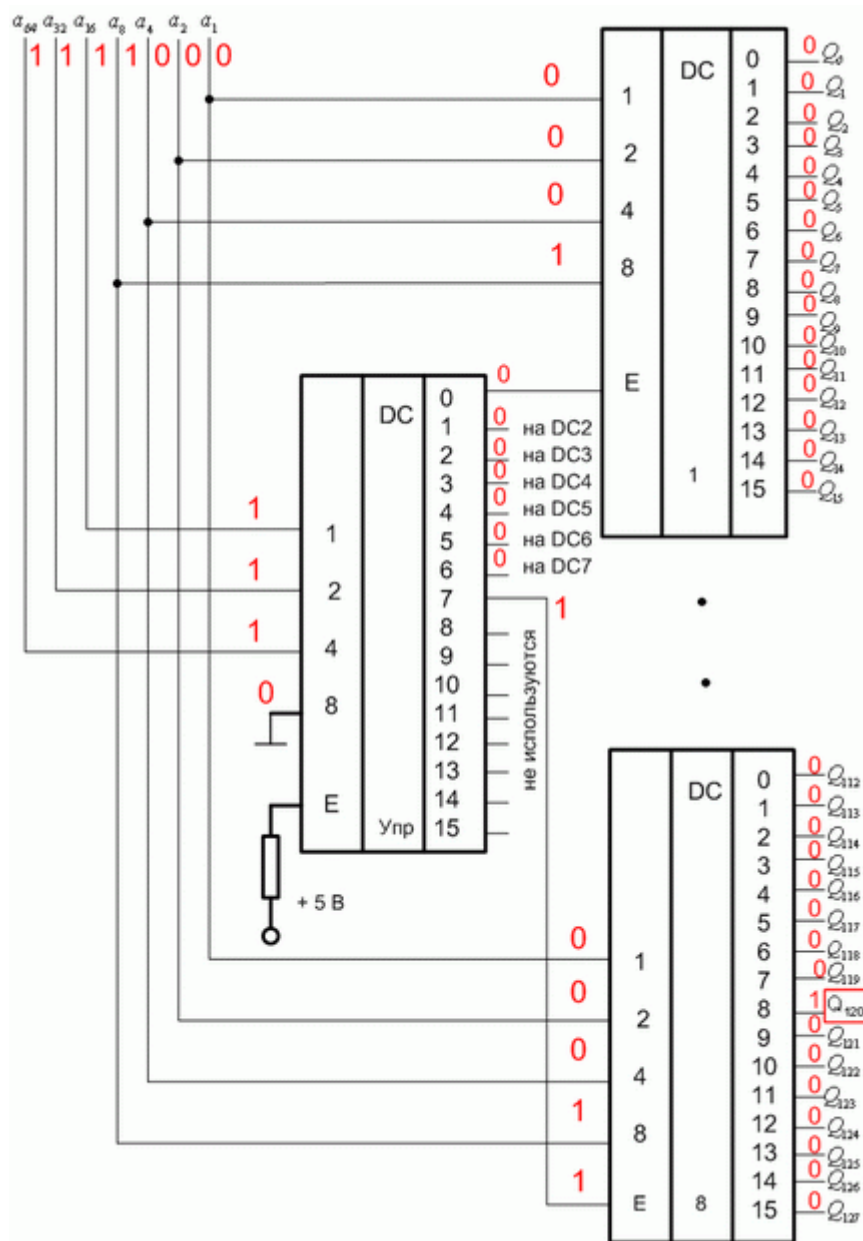


Рисунок 3. Каскад на 7 входов на базе 4-входовых дешифраторов

Они подключаются своими входами "8", "4", "2" и "1" к младшим разрядам входной информации a_8 , a_4 , a_2 и a_1 соответственно. На рис. 3 показаны только первый и восьмой дешифраторы, остальные подключаются аналогично.

Для разрешения работы каждого из этих восьми дешифраторов служит еще один дешифратор (девятый, обозначенный на схеме Упр - управляющий каскадом). Его разряды "4", "2" и "1" подключены к старшим разрядам входной информации a_{64} , a_{32} и a_{16} соответственно. На вход "8" постоянно подается логический 0, то есть этот вход заземлен. Если необходима постоянная работа каскада, то управляющий дешифратор должен работать постоянно, поэтому на его разрешающий вход постоянно подается логическая 1. В данном примере половина его выходов не используется: поскольку старший информационный вход заземлен, на выходах управляющего дешифратора с номерами 8 ... 15 всегда будет пассивный уровень логического 0.

Для понимания работы схемы рассмотрим случай, когда на входы подается код $120_{10} = 1111000_2$. Поэтому на входы постоянно работающего управляющего дешифратора будет поступать информация $a_{64} = 1$, $a_{32} = 1$ и $a_{16} = 1$. Следовательно, активный сигнал будет на одном его выходе - "7". Поэтому работать будет только восьмой дешифратор каскада, у которого $E = 1$. На его входы "8", "4", "2" и "1" будет поступать информация со входов схемы $a_8 = 1$, $a_4 = 0$, $a_2 = 0$ и $a_1 = 0$. Поэтому на его выходе "8" будет активный сигнал логической единицы, а на остальных - логический 0. На выходах всех остальных дешифраторов будут пассивные сигналы, так как у них разрешающий сигнал $E = 0$. Таким образом, на выходе схемы $Q_{120} = 1$, остальные сигналы равны 0: происходит преобразование двоичного кода в унарный в каскадной схеме включения.

Ту же схему можно изобразить более наглядно с помощью **шин** (рис. 4).

Шина - это совокупность линий, имеющих одинаковое функциональное назначение. С помощью шины можно объединить несколько линий, дав каждой из них свой номер. Номер сигнала ставится рядом с той шиной, в которую входит и из которой он выходит.

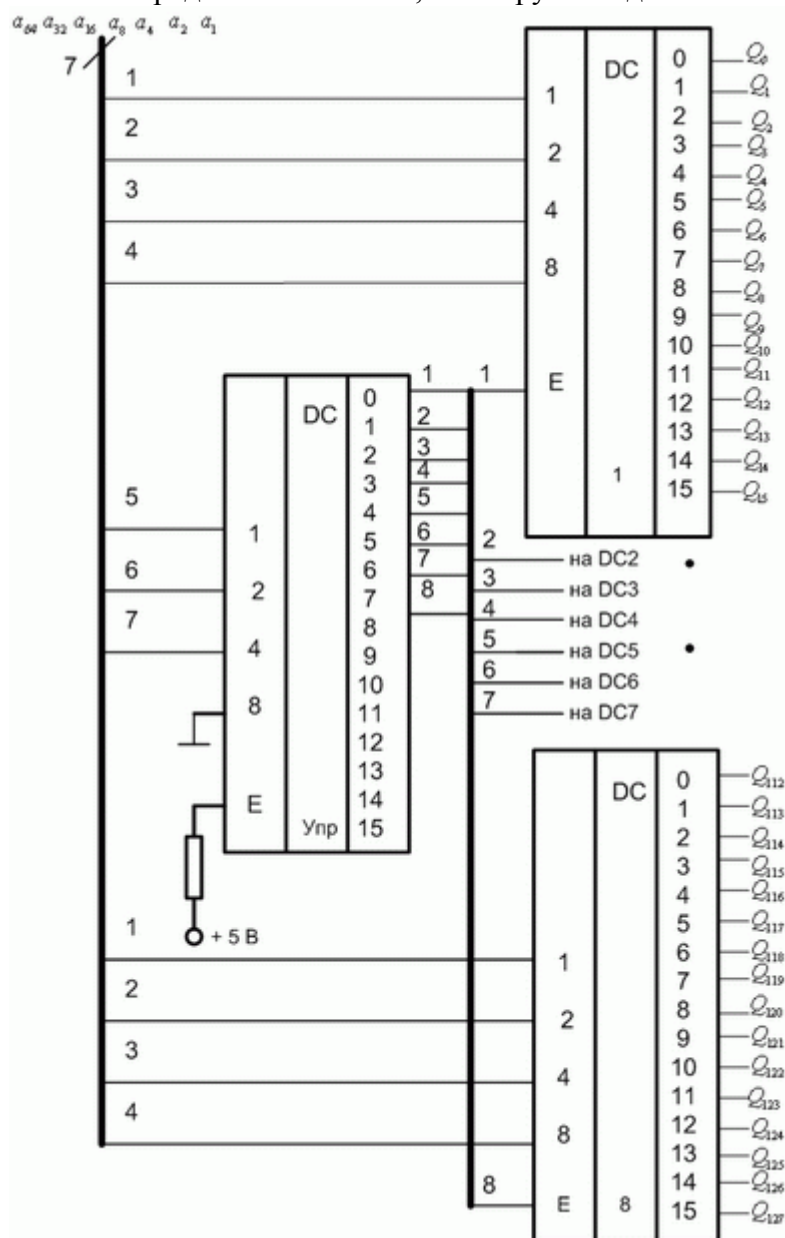


Рисунок 4. Шинная организация каскадного соединения дешифраторов

На рис. 4 изображены две шины. Одна из них объединяет входные сигналы $a_{64}, a_{32}, a_{16}, a_8, a_4, a_2$ и a_1 , а вторая - выходные сигналы управляющего дешифратора, использующиеся как разрешающие сигналы для дешифраторов 1...8 в выходной очереди каскада.

Во всех предыдущих примерах схемы содержали по две очереди каскада - входную и выходную. В тех случаях, когда разрядность каскада большая, а количество входов базового дешифратора маленькое, количество очередей возрастает. Так, например, в схеме дешифратора на 16 выходов на базе 1-входов дешифраторов, будет четыре очереди (рис. 5).

Базовый дешифратор имеет 2 выхода, поэтому для построения каскада на 16 выходов необходимо соединить 8 базовых ИС, которые образуют выходную (первую очередь) каскада. Информационные входы всех дешифраторов выходной (первой) очереди каскада подключаются к младшим входным линиям соответственно разрядности базового дешифратора. Так, в схеме на рис. 5 на дешифраторы с номерами 1.1-1.8 приходит информация с младшей входной линии a_1 . На дешифраторы 2.1 - 2.4 второй очереди - следующий, второй сигнал со входа a_2 , на дешифраторы 3.1 - 3.2 - третий a_4 и на последний дешифратор 4.1 - последний, старший входной сигнал a_8 .

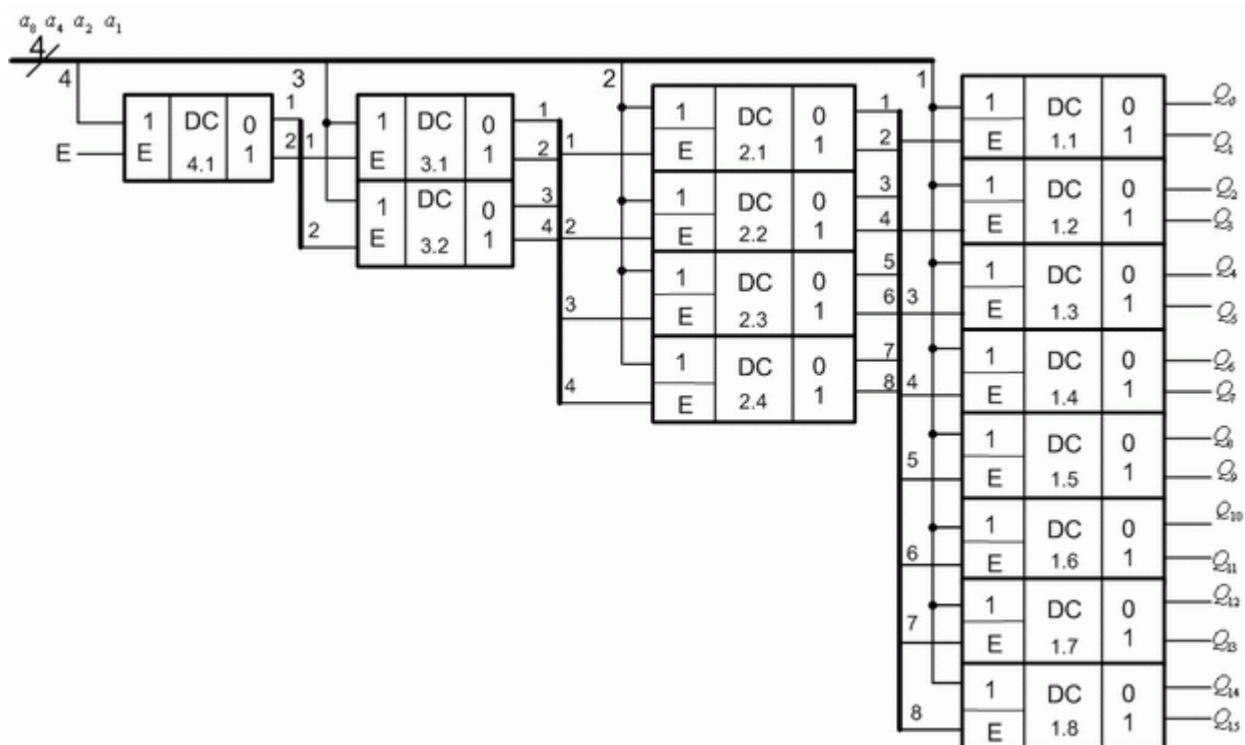


Рисунок 5. Каскад на 16 выходов на базе 1-входовых дешифраторов

По вышеизложенному принципу можно построить каскад дешифраторов на любое количество разрядов. При синтезе схемы рекомендуется придерживаться следующей последовательности действий:

1. Нарисовать базовый дешифратор.
2. Определить количество дешифраторов в выходной части каскада (общее количество выходов схемы разделить на количество выходов базового дешифратора) и нарисовать выходную очередь каскада.
3. Нарисовать входную шину каскада нужной разрядности (из расчета $N = 2^n$, где N - количество выходов схемы, n - количество необходимых входных линий).
4. Соединить входные информационные линии дешифраторов выходного каскада и младшие разряды входной информационной шины.
5. Количество выходов дешифраторов следующей очереди каскада равно количеству дешифраторов предыдущей очереди. Следовательно, определить количество дешифраторов в следующем каскаде можно, поделив количество дешифраторов в предыдущем каскаде на количество выходов базового дешифратора. Нарисовать следующую очередь.
6. Подключить выходы следующей очереди к разрешающим входам дешифраторов предыдущей очереди (на схемах очереди следуют *справа налево*). Входные информационные линии дешифраторов следующей очереди подключаются к следующим разрядам входной шины каскада и т.д.

Задания к практической работе

№ варианта	Задание
1.	Каскад на 5 входов на базе 3-входовых дешифраторов
2.	Каскад на 4 входа на баз 2-входовых дешифраторов
3.	Каскад на 32 выхода на базе 3-входовых дешифраторов
4.	Каскад на 64 выхода на базе 3-входовых дешифраторов
5.	Каскад на 8 входов на базе 6-входовых дешифраторов
6.	Каскад на 32 выхода на базе 2-входовых дешифраторов
7.	Каскад на 7 входов на на базе 5-входовых дешифраторов

8.	Каскад на 64 выхода на базе 4-входовых дешифраторов
9.	Каскад на 6 входов на базе 4-входовых дешифраторов
10.	Каскад на 64 выхода на базе 2-входовых дешифраторов
11.	Каскад на 4 входа на баз 2-входовых дешифраторов
12.	Каскад на 32 выхода на базе 2-входовых дешифраторов
13.	Каскад на 8 входов на базе 6-входовых дешифраторов
14.	Каскад на 5 входов на базе 3-входовых дешифраторов
15.	Каскад на 5 входов на баз 2-входовых дешифраторов
16.	Каскад на 7 входов на на базе 5-входовых дешифраторов
17.	Каскад на 64 выхода на базе 3-входовых дешифраторов
18.	Каскад на 32 выхода на базе 3-входовых дешифраторов
19.	Каскад на 64 выхода на базе 4-входовых дешифраторов
20.	Каскад на 64 выхода на базе 2-входовых дешифраторов

Практическая работа №12. Синтез преобразователя двоичных кодов.

Цель работы: ознакомиться с методикой и получить практические навыки синтеза преобразователя кода.

Теоретические сведения

1. Анализ функционирования преобразователя кода

Преобразователи кода – устройства, предназначенные для преобразования одного кода в другой.

В настоящее время применяется достаточно большое количество различных видов кодов, однако наибольшее распространение получили следующие коды: 8-4-2-1; 2-4-2-1; 7-4-2-1; с избытком 3; 3а+2; 2 из 5; Грея; прямой, обратный и дополнительный двоичный (таблица 1).

Таблица 1.

Цифра	Код									
	8421	2421	7421	С из 5.3	3а+2	2 из 5	Грея	ПДК	ОДК	ДДК
0	0000	0000	0000	0011	00010	11000	0000	0000	1111	0001
1	0001	0001	0001	0100	00101	01100	0001	0001	1110	0000
2	0010	0010	0010	0101	01000	00110	0011	0010	1101	0011
3	0011	0011	0011	0110	01011	00011	0010	0011	1100	0010
4	0100	0100	0100	0111	01110	10001	0110	0100	1011	0101
5	0101	1011	0101	1000	10001	10100	0111	0101	1010	0100
6	0110	1100	0110	1001	10100	01010	0101	0110	1001	0111
7	0111	1101	1000	1010	10111	00101	0100	0111	1000	0110
8	1000	1110	1001	1011	11010	10010	1100	1000	0111	1001
9	1001	1111	1010	1100	11101	01001	1101	1001	0110	1000

Преобразователи кодов используются для шифрации и дешифрации цифровой информации и имеют n входов и k выходов. Соотношения между числами n и k могут быть любыми: $n=k$, $n>k$, $n<k$.

Преобразователи кодов могут быть двух типов:

1. С невесовым преобразованием кодов.
2. С весовым преобразованием кодов.

Примером преобразователей первого типа являются преобразователи для отображения цифровой информации на основе семисегментного индикатора.

На основании данной классификации существует несколько разновидностей преобразователей кода:

1. Преобразователи двоичного кода в двоичный.
2. Преобразователи двоично-десятичных кодов в двоичный и обратно.
3. Преобразователи для отображения информации.

Условное графическое обозначение преобразователя кода представлено на рисунке 1.

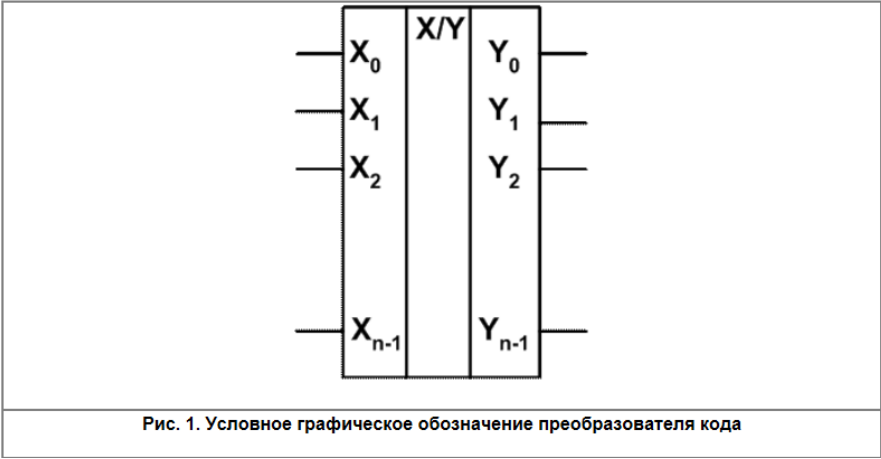


Рис. 1. Условное графическое обозначение преобразователя кода

2. Пример синтеза преобразователя кода

Рассмотрим пример синтеза на основе преобразователя кода 8-4-2-1 в код 2-4-2-1.

Таблица соответствия комбинаций приведена в таблице 2.

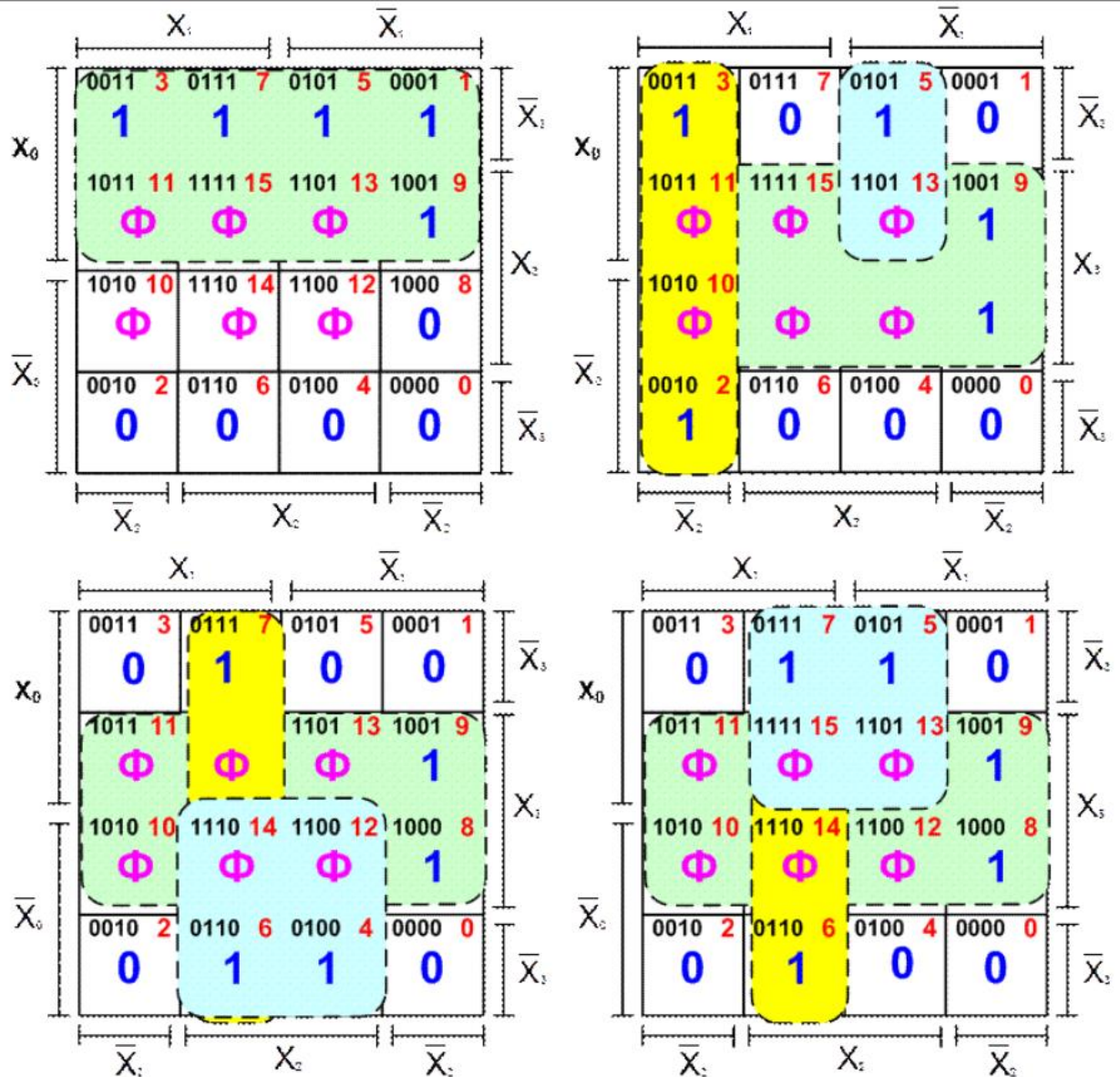
Таблица 2

Код	8421				2421			
Разряды	X ₃	X ₂	X ₁	X ₀	Y ₃	Y ₂	Y ₁	Y ₀
Вес	8	4	2	1	2	4	2	1
Цифра								
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	0
3	0	0	1	1	0	0	1	1
4	0	1	0	0	0	1	0	0
5	0	1	0	1	1	0	1	1
6	0	1	1	0	1	1	0	0
7	0	1	1	1	1	1	0	1
8	1	0	0	0	1	1	1	0
9	1	0	0	1	1	1	1	1

Составляем ДНФ.

$$\begin{aligned}
 Y_0 &= \overline{X_3}\overline{X_2}\overline{X_1}X_0 \wedge \overline{X_3}\overline{X_2}X_1X_0 \wedge \overline{X_3}X_2\overline{X_1}X_0 \wedge \overline{X_3}X_2X_1X_0 \wedge X_3\overline{X_2}\overline{X_1}X_0 \\
 Y_1 &= \overline{X_3}\overline{X_2}X_1\overline{X_0} \wedge \overline{X_3}\overline{X_2}X_1X_0 \wedge \overline{X_3}X_2\overline{X_1}X_0 \wedge X_3\overline{X_2}\overline{X_1}\overline{X_0} \wedge X_3\overline{X_2}\overline{X_1}X_0 \\
 Y_2 &= \overline{X_3}X_2\overline{X_1}\overline{X_0} \wedge \overline{X_3}X_2X_1\overline{X_0} \wedge \overline{X_3}X_2X_1X_0 \wedge X_3\overline{X_2}\overline{X_1}\overline{X_0} \wedge X_3\overline{X_2}\overline{X_1}X_0 \\
 Y_3 &= \overline{X_3}X_2X_1\overline{X_0} \wedge \overline{X_3}X_2X_1X_0 \wedge \overline{X_3}X_2\overline{X_1}X_0 \wedge X_3\overline{X_2}\overline{X_1}\overline{X_0} \wedge X_3\overline{X_2}\overline{X_1}X_0
 \end{aligned}$$

Проводим минимизацию.



Получаем аналитические выражения:

$$Y_0 = X_0$$

$$Y_1 = X_3 + \bar{X}_2 X_1 + X_2 \bar{X}_1 X_0$$

$$Y_2 = X_3 + X_2 X_1 + X_2 \bar{X}_0$$

$$Y_3 = X_3 + X_2 X_1 + X_2 \bar{X}_1 X_0$$

Записываем выражения в базисе И-НЕ:

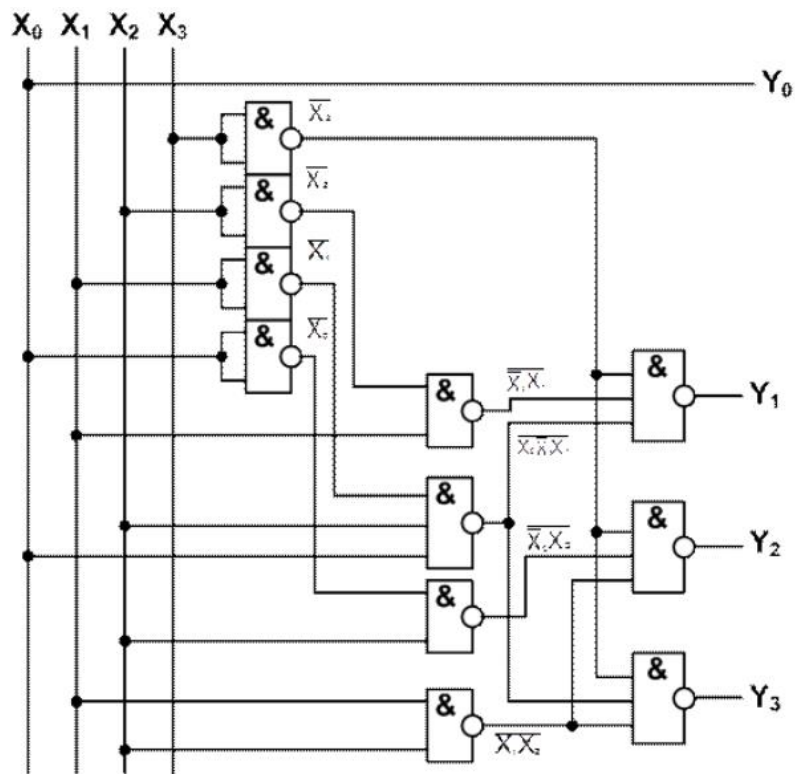
$$Y_0 = \overline{\bar{X}_0} = X_0$$

$$Y_1 = \overline{\overline{X_3} + \overline{\bar{X}_2 X_1} + \overline{X_2 \bar{X}_1 X_0}} = \overline{\bar{X}_3 \cdot \bar{X}_2 X_1 \cdot X_2 \bar{X}_1 X_0}$$

$$Y_2 = \overline{\overline{X_3} + \overline{X_2 X_1} + \overline{X_2 \bar{X}_0}} = \overline{\bar{X}_3 \cdot \bar{X}_2 X_1 \cdot X_2 \bar{X}_0}$$

$$Y_3 = \overline{\overline{X_3} + \overline{X_2 X_1} + \overline{X_2 \bar{X}_1 X_0}} = \overline{\bar{X}_3 \cdot \bar{X}_2 X_1 \cdot X_2 \bar{X}_1 X_0}$$

Строим схему устройства.



Задание на практическую работу

1. Выбрать из таблицы задание, согласно своему варианту (по номеру в журнале).

№ варианта	Исходный код	Конечный код
1.	8421	4221
2.	8421	7421
3.	8421	Грея
4.	8421	Грея-Штибица
5.	8421	Айкена
6.	Грея-Штибица	ПДК
7.	Грея-Штибица	ОДК
8.	Грея-Штибица	ДДК
9.	Грея-Штибица	4221
10.	Грея-Штибица	7421
11.	7421	Грея
12.	7421	Грея-Штибица
13.	7421	Айкена
14.	7421	ПДК
15.	7421	ОДК
16.	2421	ДДК
17.	2421	7421
18.	4221	Айкена
19.	4221	Грея
20.	4221	Грея-Штибица

2. Произвести синтез преобразователя кода.

Таблица соответствия двоичных кодов										
Десятичная цифра	8421	7421	4221	2421	Грея	Грея-Штибица	Айкена	ПДК	ОДК	ДДК
0	0000	0000	0011	0000	0000	0010	0000	0000	1111	0001
1	0001	0001	0100	0001	0001	0110	0001	0001	1110	0000
2	0010	0010	0101	0010	0011	0111	0010	0010	1101	0011
3	0011	0011	0110	0011	0010	0101	0011	0011	1100	0010

4	0100	0100	0111	0100	0110	0100	0100	0100	1011	0101
5	0101	0101	1000	1011	0111	1100	1011	0101	1010	0100
6	0110	0110	1001	1100	0101	1001	1100	0110	1001	0111
7	0111	1000	1010	1101	0100	1101	1101	0111	1000	0110
8	1000	1001	1011	1110	1100	1111	1110	1000	0111	1001
9	1001	1010	1100	1111	1101	1110	1111	1001	0110	1000

Содержание отчета

Отчет по практической работе должен содержать:

1. Исходное задание.
2. Этапы минимизации.
3. Реализацию задания в виде функциональной схемы.

Практическая работа №13. Преобразователи произвольных кодов.

Цель работы: рассмотреть принцип синтеза логических схем, реализующих заданную математическую формулу.

Теоретическая часть

Преобразователь произвольных кодов - это логическая схема, реализующая определённую математическую формулу. На вход такой схемы подаётся двоичный код, соответствующий значению переменной $x_{вх}$. На выходе схемы получается двоичный код, соответствующий значению функции $x_{вых}$, вычисленному по указанной математической формуле.

В качестве примера синтеза логической схемы рассмотрим 3-входовую схему, реализующую увеличение входного кода в три раза: $x_{вых} = 3x_{вх}$. Последовательность действий при решении подобных задач следующая.

1. Определим максимально возможный код на выходе 3-входовой схемы: $7_{10} \times 3_{10} = 21_{10} = 10101_2$ - это *пятиразрядное* двоичное число. Поэтому количество выходов для данной схемы будет равно *пяти*.

2. Заполним таблицу истинности для синтезируемой схемы (табл. 1). Поскольку количество выходов данной схемы больше одного, таблица включает в себя несколько (здесь *пять*) столбцов, соответствующих двоичным разрядам выходного сигнала.

3. Для каждого выхода найдем минимальное выражение с помощью карт Карно (рис. 1).

4. По полученным выражениям построим логическую схему на пять выходов, каждый из которых соответствует двоичному разряду вычисляемого по заданной формуле числа (рис. 2).

Таблица 1. Таблица истинности трехвходовой схемы умножения на 3									
Входной код				Выходной код					
В десятичном выражении	В двоичном коде			В десятичном выражении	В двоичном коде				
	a	b	c		Q ₄	Q ₃	Q ₂	Q ₁	Q ₀
	2 ²	2 ¹	2 ⁰		2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
0	0	0	0	0	0	0	0	0	0
1	0	0	1	3	0	0	0	1	1
2	0	1	0	6	0	0	1	1	0
3	0	1	1	9	0	1	0	0	1
4	1	0	0	12	0	1	1	0	0
5	1	0	1	15	0	1	1	1	1
6	1	1	0	18	1	0	0	1	0

7	1	1	1	21	1	0	1	0	1
---	---	---	---	----	---	---	---	---	---

bc

	<i>00</i>	<i>01</i>	<i>11</i>	<i>10</i>
<i>0</i>	0	0	0	0
<i>1</i>	0	0	1	1

$Q_4 = ab$

bc

	<i>00</i>	<i>01</i>	<i>11</i>	<i>10</i>
<i>0</i>	0	0	1	0
<i>1</i>	1	1	0	0

$Q_3 = a\bar{b} + \bar{a}bc$

bc

	<i>00</i>	<i>01</i>	<i>11</i>	<i>10</i>
<i>0</i>	0	0	0	1
<i>1</i>	1	1	1	0

$Q_2 = a\bar{b} + ac + \bar{a}b\bar{c}$

bc

	<i>00</i>	<i>01</i>	<i>11</i>	<i>10</i>
<i>0</i>	0	1	0	1
<i>1</i>	0	1	0	1

$Q_1 = \bar{b}c + b\bar{c}$

bc

	<i>00</i>	<i>01</i>	<i>11</i>	<i>10</i>
<i>0</i>	0	1	1	0
<i>1</i>	0	1	1	0

$Q_0 = c$

Рисунок 1. Минимизация логических выражений для выходных сигналов преобразователя, реализующего формулу $x_{\text{вых}} = 3x_{\text{вх}}$

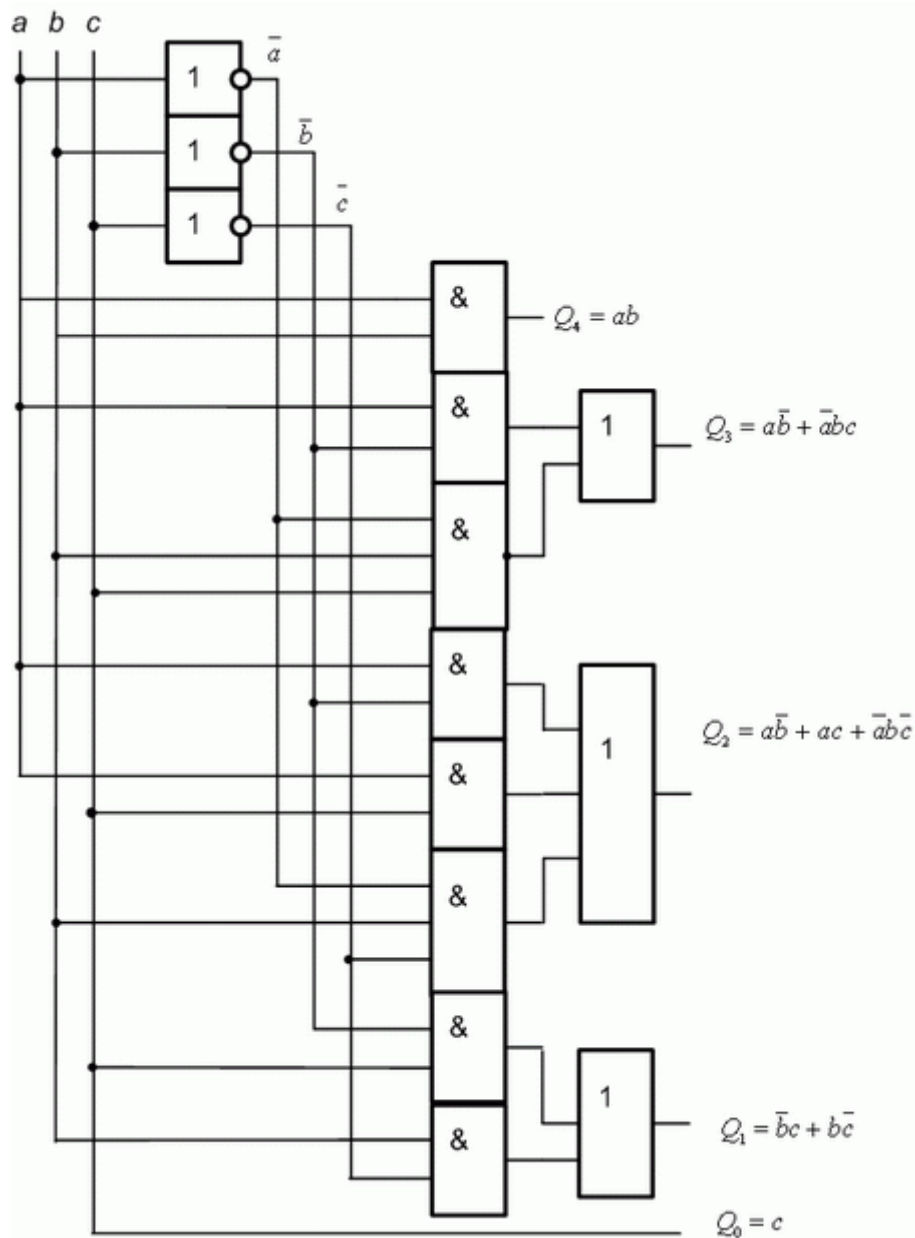


Рисунок 2. Логическая схема преобразователя на 3 входа, реализующего формулу умножения на 3

Рассмотрим далее схему на 4 входа, реализующую ту же самую формулу $x_{\text{вых}} = 3x_{\text{вх}}$. Алгоритм решения тот же.

1. Определим максимально возможный код на выходе 4-входовой схемы: $15_{10} \times 3_{10} = 45_{10} = 101101_2$ — это *шестизначное* двоичное число. Поэтому количество выходов для данной схемы будет равно *шести*.

2. Заполним таблицу истинности для синтезируемой схемы (табл. 2). Она включает в себя *шесть* столбцов, соответствующих двоичным разрядам выходного сигнала.

3. Для каждого выхода найдем минимальное выражение с помощью карт Карно (рис. 3).

4. По полученным выражениям построим логическую схему на шесть выходов, каждый из которых соответствует двоичному разряду вычисляемого по заданной формуле числа (рис. 4).

5. Если весь столбец значений для выхода $Q_i = 1$, это означает, что независимо от состояния входных сигналов на выход Q_i подаётся напряжение источника питания.

6. Если весь столбец значений для выхода $Q_i = 0$, это означает, что независимо от состояния входных сигналов выход Q_i подключен к общей точке ("земле").

Таблица 2. Таблица истинности четырёхвходовой схемы умножения на 3											
Входной код					Выходной код						
В десятичном выражении	В двоичном коде				В десятичном выражении	В двоичном коде					
	a	b	c	d		Q ₅	Q ₄	Q ₃	Q ₂	Q ₁	Q ₀
	2 ³	2 ²	2 ¹	2 ⁰		2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	3	0	0	0	0	1	1
2	0	0	1	0	6	0	0	0	1	1	0
3	0	0	1	1	9	0	0	1	0	0	1
4	0	1	0	0	12	0	0	1	1	0	0
5	0	1	0	1	15	0	0	1	1	1	1
6	0	1	1	0	18	0	1	0	0	1	0
7	0	1	1	1	21	0	1	0	1	0	1
8	1	0	0	0	24	0	1	1	0	0	0
9	1	0	0	1	27	0	1	1	0	1	1
10	1	0	1	0	30	0	1	1	1	1	0
11	1	0	1	1	33	1	0	0	0	0	1
12	1	1	0	0	36	1	0	0	1	0	0
13	1	1	0	1	39	1	0	0	1	1	1
14	1	1	1	0	42	1	0	1	0	1	0
15	1	1	1	1	45	1	0	1	1	0	1

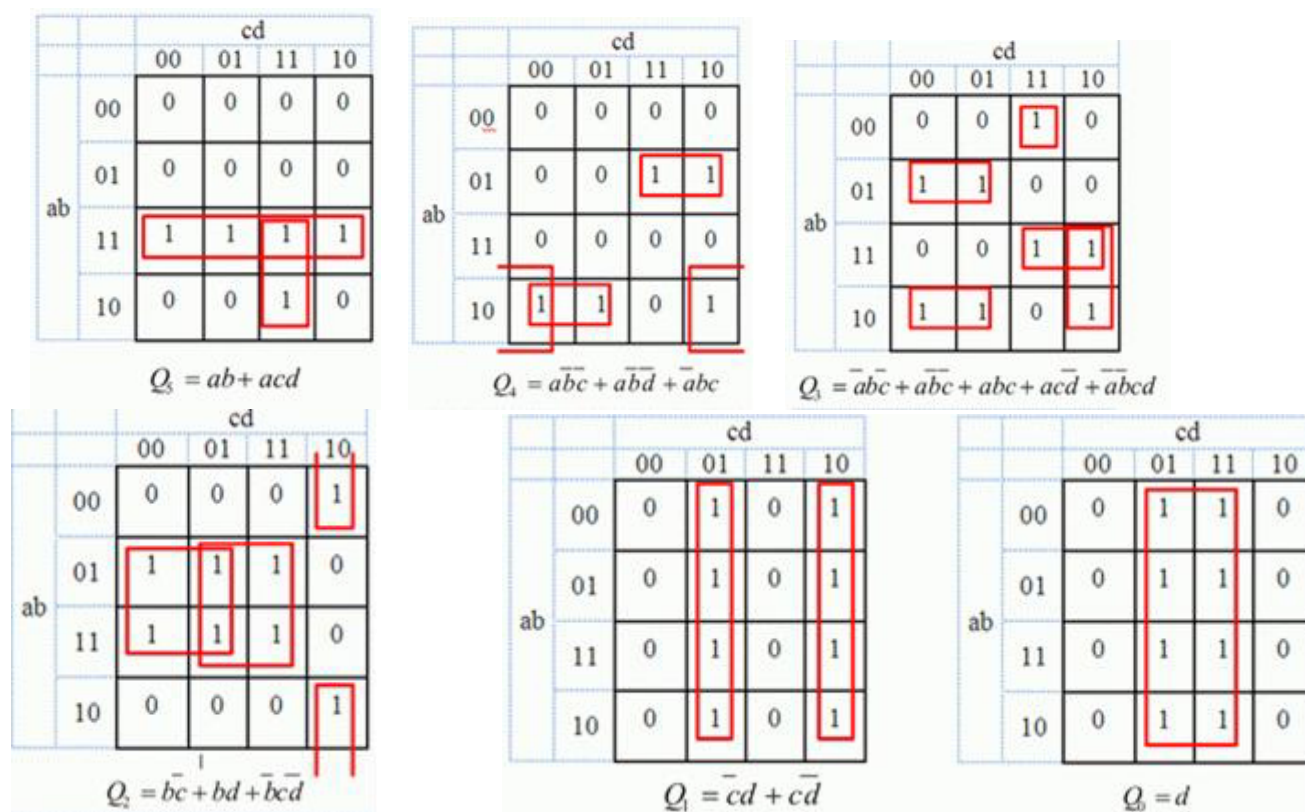


Рисунок 3. Минимизация логических выражений для выходных сигналов 4-входового преобразователя, реализующего формулу $x_{\text{вых}} = 3x_{\text{вх}}$.

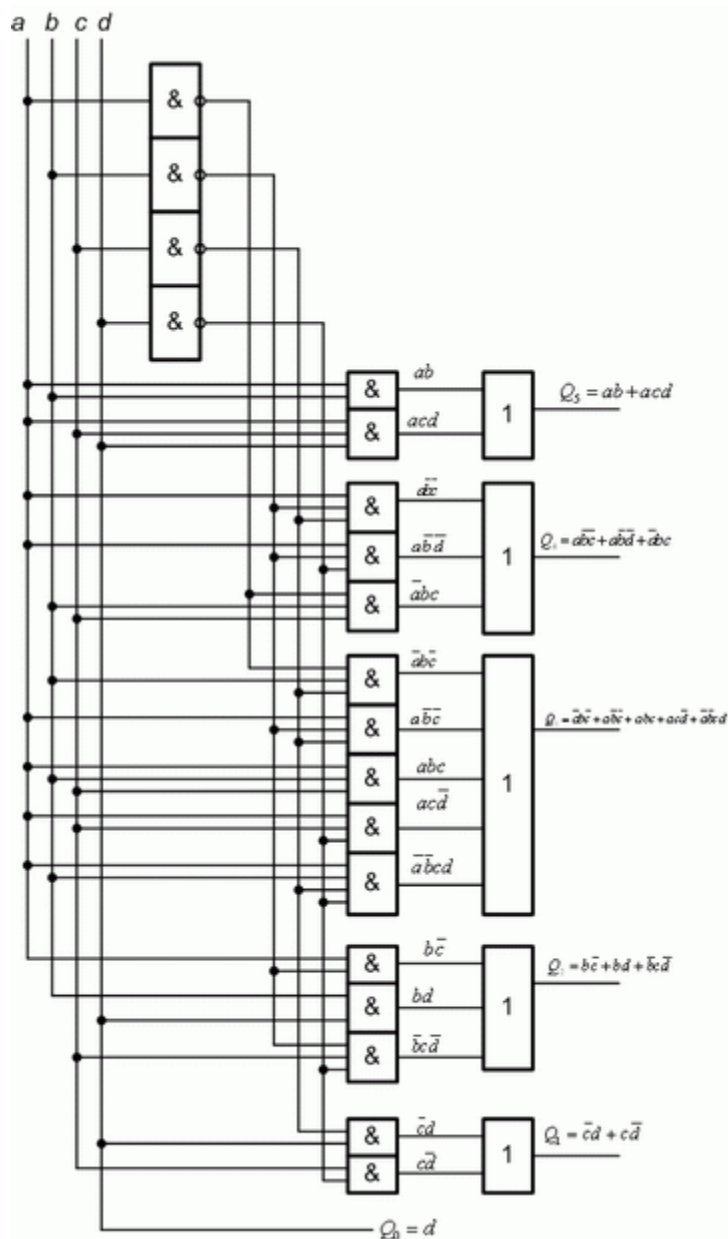


Рисунок 4. Логическая схема преобразователя на 4 входа, реализующего формулу умножения на 3

Практическая часть

1. Синтезировать логические схемы, реализующие заданную математическую формулу.

№ варианта	Математическая формула
1.	$X_{\text{ВЫХ}} = 3X_{\text{ВХ}} + 1$
2.	$X_{\text{ВЫХ}} = 3X_{\text{ВХ}} + 2$
3.	$X_{\text{ВЫХ}} = 3X_{\text{ВХ}} + 3$
4.	$X_{\text{ВЫХ}} = 3X_{\text{ВХ}} + 4$
5.	$X_{\text{ВЫХ}} = 3X_{\text{ВХ}} + 5$
6.	$X_{\text{ВЫХ}} = 3X_{\text{ВХ}} + 6$
7.	$X_{\text{ВЫХ}} = 3X_{\text{ВХ}} + 7$
8.	$X_{\text{ВЫХ}} = 3X_{\text{ВХ}} + 8$
9.	$X_{\text{ВЫХ}} = 3X_{\text{ВХ}} + 9$
10.	$X_{\text{ВЫХ}} = 3X_{\text{ВХ}} + 10$
11.	$X_{\text{ВЫХ}} = 2X_{\text{ВХ}} + 11$
12.	$X_{\text{ВЫХ}} = 2X_{\text{ВХ}} + 12$
13.	$X_{\text{ВЫХ}} = 2X_{\text{ВХ}} + 13$
14.	$X_{\text{ВЫХ}} = 2X_{\text{ВХ}} + 14$
15.	$X_{\text{ВЫХ}} = 2X_{\text{ВХ}} + 15$

16.	$X_{ВЫХ} = 2X_{ВХ} + 16$
17.	$X_{ВЫХ} = 2X_{ВХ} + 17$
18.	$X_{ВЫХ} = X_{ВХ} + 18$
19.	$X_{ВЫХ} = X_{ВХ} + 19$
20.	$X_{ВЫХ} = X_{ВХ} + 20$

Содержание отчета

Отчет по практической работе должен содержать:

1. Исходное задание.
2. Этапы минимизации.
3. Реализацию задания в виде функциональной схемы.

Практическая работа №14. Синтез мультиплексоров.

Цель работы: ознакомиться с методикой и получить практические навыки синтеза дешифраторов и мультиплексоров на интегральных микросхемах малой степени интеграции.

Краткие теоретические сведения и методические указания

В вычислительной технике многие часто используемые комбинационные схемы выпускаются в виде отдельных микросхем. К таким схемам относятся мультиплексоры.

Мультиплексором называется комбинационная схема, осуществляющая передачу сигнала с одного из своих информационных входов на единственный свой выход. В общем случае мультиплексор имеет 2^n информационных входов (X_i), один выход (Y) и n управляющих входов (Z_i). Последние задают номер входа, с которого передается сигнал. Работу мультиплексора можно описать уравнением

$$Y_0 = \bar{Z}_0 \cdot \bar{Z}_1 \cdot \dots \cdot \bar{Z}_{n-1} \cdot X_0 \cup Z_0 \cdot \bar{Z}_1 \cdot \dots \cdot \bar{Z}_{n-1} \cdot X_1 \cup \dots \cup Z_0 \cdot Z_1 \cdot \dots \cdot Z_{n-1} \cdot X_{2^n-1}$$

На рис.1 приведено условное обозначение мультиплексора на 2 управляющих входа.

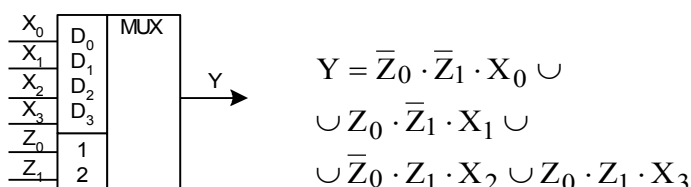


Рисунок 1. Пример функциональной схемы мультиплексора.

Зачастую в выпускаемых промышленно мультиплексорах имеется еще один управляющий вход E , который используется для стробирования и наращивания числа входов мультиплексора. При активном сигнале на входе E мультиплексор работает, как описано выше. При пассивном сигнале на входе E сигнал на выходе Y также будет пассивным независимо от уровней сигналов на остальных входах. На рис.2 показана реализация мультиплексора $8 \rightarrow 1$ на двух мультиплексорах $4 \rightarrow 1$ с управляющими входами E .

Здесь схема И3 реализует функцию $E_1 = \overline{\text{строб}} \cdot \bar{Z}_2 = \overline{\text{строб}} \cup \bar{Z}_2$. Т.е. MUX1 будет работать в том случае, если оба эти сигнала (строб и Z_2) активны (равны единице).

Схема И4 реализует функцию

$$E_2 = \overline{\text{строб}} \cdot Z_2 \cdot \text{строб} = \text{строб} \cdot Z_2 \cup \overline{\text{строб}} = Z_2 \cup \overline{\text{строб}}$$

То есть MUX2 будет работать при условии, что строб = 1 и $Z_2 = 0$.

Схема И5 реализует функцию

$$Y = \overline{Y_1} \cdot \overline{Y_2} = Y_1 \cup Y_2$$

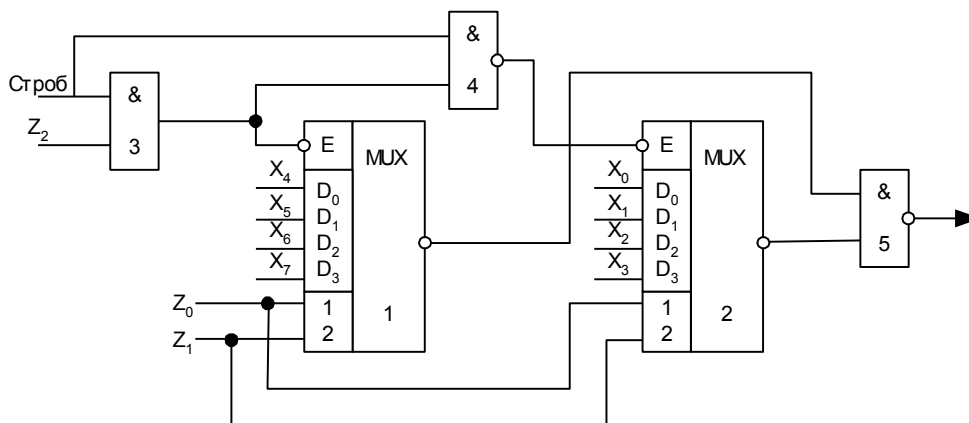


Рисунок 2. Пример увеличения размерности мультиплексора.

Мультиплексор $2^n \rightarrow 1$, как следует из описывающего его работу уравнения, позволяет простейшим образом реализовать любую логическую функцию от n переменных. Для этого достаточно рассматривать управляющие входы Z_j как входы, на которые подаются аргументы функции, а на информационные входы X_i завести значение функции на соответствующих наборах переменных. Пример показан на рис. 3.

0	1	

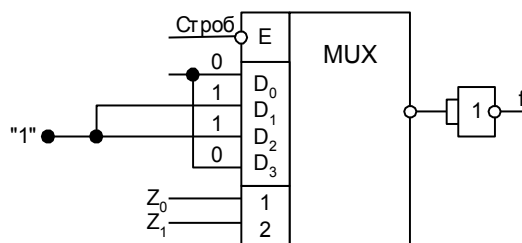


Рисунок 3. Пример реализации логической функции с использованием мультиплексора.

На мультиплексоре $2n \rightarrow 1$ можно реализовать (с помощью дополнительной логики) и функции с числом аргументов большим, чем n . На рисунке 4 показан пример реализации $F(Z_0, Z_1, Z_2)$ на мультиплексоре $4 \rightarrow 1$.

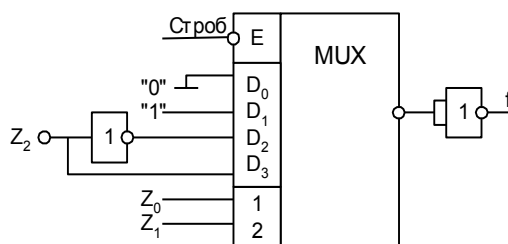


Рисунок 4. Пример реализации логической функции с числом аргументов большим, чем n .

В практической работе предполагается использовать мультиплексор $8 \rightarrow 1$ КП7. Его условное обозначение приведено на рис.5.

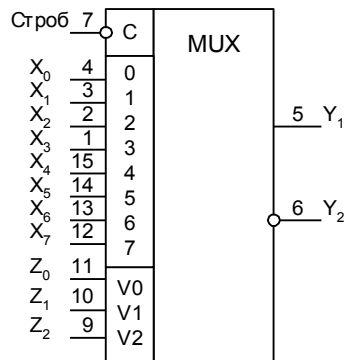


Рисунок 5. Условное графическое обозначение мультиплексора КР1533КП7.

Варианты заданий на практическую работу

В соответствии с номером варианта согласно таблице 1 построить комбинационную схему на ИМС мультиплексоров.

Таблица 1

№ вар	Задание
1.	Реализовать на КП7 и дополнительной логике функцию $F = \overline{X_1} \cup \overline{X_2} \oplus X_3 \cup X_2 \overline{X_4}$
2.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \cdot \overline{X_2} \cdot (X_2 \oplus \overline{X_3}) \cup (\overline{X_1} X_2 X_3)$
3.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \cup X_2 \cup X_3 \cdot X_4$
4.	Реализовать на КП7 и дополнительной логике функцию $F = \overline{X_1} X_2 \cup \overline{X_2} X_3 \cup X_1 X_2 \overline{X_4}$
5.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 X_2 \cup (X_1 \oplus X_2 \oplus X_3)$
6.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \oplus \overline{X_2} \cup \overline{X_3} X_4$
7.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \overline{X_2} (X_2 \oplus \overline{X_3}) \cup \overline{X_1} X_2 X_3$
8.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 X_2 \cup X_2 X_3 \cup X_3 X_4$
9.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \oplus X_2 \oplus X_3 \oplus X_4$
10.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \cdot \overline{X_4} \cdot (X_2 \cup \overline{X_3}) \oplus (\overline{X_1} \overline{X_2} X_3)$
11.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \cdot X_2 \cdot X_3 \cup \overline{X_2} \overline{X_4}$
12.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 X_2 X_4 \oplus \overline{X_1} \overline{X_2} \overline{X_3}$
13.	Реализовать на КП7 и дополнительной логике функцию $F = \overline{X_1} \cup X_3 \cdot (X_2 \cup \overline{X_4}) \cup (X_1 X_2 \oplus \overline{X_3})$
14.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \oplus X_2 \cup X_2 \oplus \overline{X_3} \cup X_3 X_4$
15.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \cup X_2 \oplus X_3 \cup \overline{X_4}$
16.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 X_2 \cup X_3 \oplus \overline{X_1} X_2 \overline{X_3}$
17.	Реализовать на КП7 и дополнительной логике функцию $F = \overline{X_1} X_2 \oplus (X_1 X_2 \cup \overline{X_3})$
18.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \overline{X_2} \oplus \overline{X_3} \cup X_4$
19.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \cup X_2 X_3 \oplus \overline{X_1} X_3$
20.	Реализовать на КП7 и дополнительной логике функцию $F = X_1 \cdot X_2 \oplus X_3 \cup \overline{X_2} X_4$

Порядок выполнения работы

1. Выбрать из таблицы задание, согласно варианту.
2. Составить таблицу истинности для заданной функции.

3. Реализовать заданную функцию на мультиплексоре.
4. Проверить правильность функционирования.

Содержание отчета

Отчет по практической работе должен содержать:

1. Исходное задание.
2. Реализацию задания в виде функциональной схемы.
3. Краткое описание применяемой микросхемы.

Контрольные вопросы

1. Что такое мультиплексор?
2. Для чего в промышленных мультиплексорах добавлен еще один управляющий вход?
3. Как на мультиплексоре $2n \rightarrow 1$ можно реализовать функции с числом аргументов, больших n ?

Практическая работа № 15. Построение функциональной схемы сумматора.

Цель работы: составить схемы сумматоров для реализации сложения четырехразрядных и восьмиразрядных двоичных чисел.

Краткие теоретические положения

Сумматор – это комбинационное цифровое устройство, предназначенное для сложения двоичных чисел. При сложении одноразрядных чисел образуется двухразрядное число (с учетом переноса) и схема полного сумматора должна складывать три одноразрядных числа: два слагаемых разряда двоичных числа a_i , b_i и перенос p_{i-1} , получаемый при сложении предыдущих младших разрядов.

Логическая схема полного одноразрядного сумматора представлена на рис. 1.

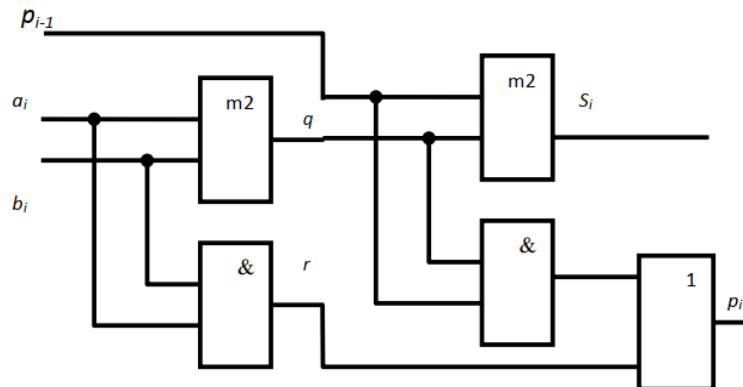


Рисунок 1. Логическая схема полного одноразрядного сумматора

Данное суммирование производится в соответствии с таблицей 1.

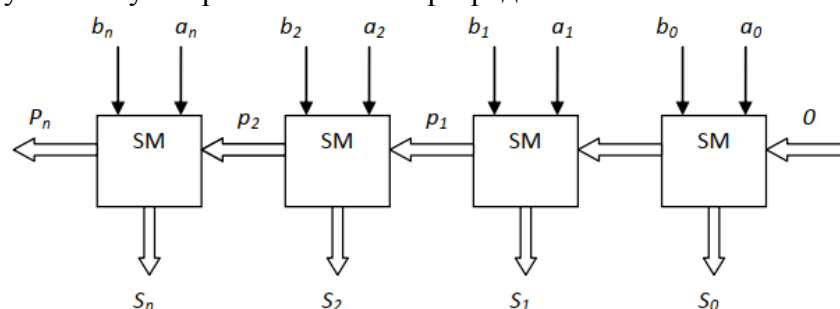
Таблица истинности сложения трех чисел

a_i	b_i	p_{i-1}	q	r	s	S_i	p_i
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	1	0	0	1	0
0	1	1	1	0	1	0	1
1	0	0	1	0	0	1	0
1	0	1	1	0	1	0	1
1	1	0	0	1	0	0	1
1	1	1	0	1	0	1	1

где a_i , b_i – разряды слагаемых чисел; p_{i-1} – перенос от предыдущей суммы; q , r , s – промежуточные результаты, получаемые в соответствии с выражениями:

$$q = a_i \oplus b_i, \quad r = a_i \cdot b_i, \quad s = p_{i-1} \cdot q, \quad S_i = q \oplus p_{i-1}, \quad p_i = r + s$$

При суммировании многоразрядных двоичных чисел одноразрядные сумматоры соединяют в последовательные структуры (рис.2). Выходы переноса P_i микросхем младших разрядов подключаются к входам переноса P_{i-1} микросхем старших разрядов. Разряды слагаемых подаются на входы a_i и b_i микросхем соответствующих разрядов, а с их выходов S_i снимаются разряды результата суммирования. Выход переноса P_n микросхемы самого старшего разряда является выходом переноса результата суммирования всего n -разрядного числа.

Рисунок 2. Структурная схема n -разрядного сумматора

Последовательность выполнения

1. Построить логическую схему сумматора для сложения двух двоичных четырехразрядных чисел a и b . Для этого необходимо взять четыре одноразрядных сумматора (рис.1) и соединить их в соответствии с рис. 2. Подключить на входы схемы четыре ключа для задания значений числа a и четыре ключа для числа b . Выход последнего переноса P и выхода результата суммы S последовательно подключить к логическому анализатору.

2. Задать переключением ключей значения логических сигналов на входах построенной схемы четырехразрядного сумматора при сложении заданных в табл. 1 двоичных чисел a и b (графы 2,3).

Исходные данные

№ варианта	Двоичные четырёхразрядные числа	
	<i>a</i>	<i>b</i>
1	2	3
<i>Пример</i>	1001	1100
0	1011	1111
1	1100	1100
2	1111	1101
3	1110	1001
4	0011	0111
5	0111	1110
6	1100	1000
7	0111	0110
8	1011	1001
9	1111	1111

3. Зафиксировать в отчет показания логического анализатора.
4. Выполнить математически (в столбик) сложение двух заданных числе *a* и *b*.
5. Сравнить результат сложения двух заданных двоичных чисел п. 4 и п.5. Если результаты одинаковые, то построение и расчет выполнены верно.
6. Оформить отчет.

Содержание отчета

1. Тема и цель практической работы.
2. Схема включения четырехразрядного сумматора с зафиксированным показанием логического анализатора.
3. Результат сложения двух заданных двоичных чисел, выполненный математическим способом.
4. Вывод о проделанной работе.

Контрольные вопросы

1. Пояснить маркировку сумматоров, привести примеры ИМС сумматоров, вычитателей.
2. В каком году был построен первый механический сумматор?
3. Перечислить элементы, на которых строятся логические схемы сумматоров.

Практическая работа №16. Исследование работы АЛУ

Цель работы: проверить правильность функционирования микросхемы арифметико-логического устройства.

Краткие теоретические сведения

Арифметико-логическое устройство АЛУ (Arithmetic-Logic Unit - ALU) представляет собой комбинационное устройство на основе сумматора, выполняющее ряд дополнительных функций обработки данных. На рисунке 1 представлено условное обозначение типичного 4-разрядного АЛУ, реализованного в модуле 74181.

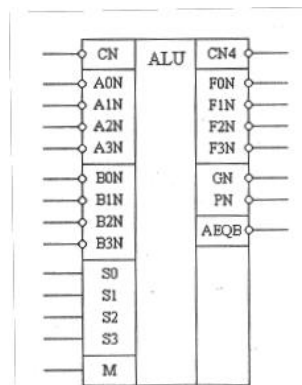


Рисунок 1. Условное обозначение четырёхразрядного АЛУ (модуль 74181)

Представленное АЛУ содержит две группы входов для двух 4-разрядных операндов (обрабатываемых двоичных чисел) A0N...A3N и B0N...B3N. На выходах F0N...F3N формируется 4-разрядный результат выполненной операции. Вход CN предназначен для подачи входного переноса в младший разряд устройства, если такой перенос существует. Выход CN4 содержит перенос в пятый разряд в случае его возникновения в ходе выполнения операции. Выходы GN и PN предназначены для подачи групповых функций генерации и распространения на внешнюю схему ускоренного формирования переносов (74182). Эта схема может быть использована для построения АЛУ с разрядностью более 4 и параллельным распространением переноса между 4-разрядными блоками. На выходе AEQV формируется сигнал равенства операндов A и B.

Входы S0...S3 и M обеспечивают выбор одной из 32 возможных операций, выполняемых устройством. Список операций представлен в таблице 1. Все операции делятся на две группы: логические и арифметические. Переключение между группами осуществляется с помощью входа M. Приведенная таблица соответствует положительной логике линий данных (1 - высокий уровень). Входные 4-разрядные операнды (данные) и 4-разрядный результат операции обозначены в таблице буквами A, B и F. Бит входного переноса носит имя соответствующего вывода модуля CN. Логические операции выполняются над операндами побитово (поразрядно), т. е. без взаимодействия разрядов. Эти операции присутствуют и в логическом и в арифметическом режимах. Для отличия от арифметических операций, выполняемых с учетом переносов в разрядах и обозначаемых в таблице обычными символами + и логические операции указаны с помощью кратких обозначений языка AHDL (символом решетка (#) обозначена операция дизъюнкции (ИЛИ), символом доллар (\$) – операция сложения по модулю 2 (исключающее ИЛИ)). Если при выполнении операций арифметического режима возникает перенос в пятый разряд, то на выходе CN4 устанавливается 0, в противном случае на нем остается 1.

Таблица истинности 4-разрядного АЛУ 74181

Входы выбора операции				Результат операции (F)		
S3	S2	S1	S0	Логический режим M=H	Арифметический режим M=L	
					CN = H	CN = L
L	L	L	L	!A	A	A + 1
L	L	L	H	!(A \$ B)	A # B	(A # B) + 1
L	L	H	L	(!A) & B	A # (!B)	(A # (!B)) + 1
L	L	H	H	0	- 1 *	0 ***
L	H	L	L	!(A & B)	A + (A & (!B))	A + (A & (!B)) + 1
L	H	L	H	!B	(A # B) + (A & (!B))	(A # B) + (A & (!B)) + 1
L	H	H	L	A \$ B	A - B - 1	A - B
L	H	H	H	A & (!B)	(A & (!B)) - 1	(A & (!B))
H	L	L	L	(!A) # B	A + (A & B)	A + (A & B) + 1
H	L	L	H	!(A \$ B)	A + B	A + B + 1
H	L	H	L	B	(A # (!B)) + (A & B)	(A # (!B)) + (A & B) + 1
H	L	H	H	A & B	(A & B) - 1	(A & B)
H	H	L	L	1	A + A **	A + A + 1
H	H	L	H	A # (!B)	(A # B) + A	(A # B) + A + 1
H	H	H	L	A # B	(A # (!B)) + A	(A # (!B)) + A + 1
H	H	H	H	A	A - 1	A

* -1 представляется в дополнительном коде, т. е. 1111 для 4-разрядного устройства.

** Сдвиг влево на один разряд (умножение на 2).

*** 0 представляется в 4-разрядном устройстве как 0000.

Задание на практическую работу

В качестве микросхемы АЛУ взять микросхему серии 74 – 74181. Подключить, проверить работу микросхемы по таблице 2.

В отчёте расписать 6 операций с двумя разными наборами операндов. Результаты работы оформить по таблице 2.

Таблица 2

Результаты работы

Код операции	Операция	Операнд А	Операнд В	Результат

Практическая работа №17. Цифровые компараторы.

Цель работы: изучение правил выполнения операции сравнения двоичных чисел и исследование принципов построения цифровых компараторов.

Теоретические основы практической работы

Компаратором (устройством сравнения) называют функциональный узел, обеспечивающий сравнение двух чисел А и В. Если А и В – n-разрядные двоичные числа, то компаратор именуют цифровым.

Простейшие компараторы формируют на выходе однобитовый сигнал равенства, или неравенства сравниваемых чисел А и В. Эти отношения используются как логические условия в микропрограммах, в устройствах контроля и диагностики ЭВМ, в устройствах автоматики компараторы используются для сигнализации о выходе величин за установленные пределы и т.д.

Компараторы строятся на основе поразрядных операций над одноименными разрядами обоих слов. Слова равны, если попарно равны все одноименные их разряды. Признак (условие) равенства i -х разрядов сравниваемых слов A и B :

$$r_{ip} = a_i b_i + \bar{a}_i \bar{b}_i = \overline{a_i \bar{b}_i + \bar{a}_i b_i} = \overline{a_i \oplus b_i} = 1 \quad (1)$$

Условие неравенства i -х разрядов:

$$r_{in} = a_i \bar{b}_i + \bar{a}_i b_i = a_i \oplus b_i = 1 \quad (2)$$

Схемная реализация приведенных условий изображена на рис. 1, а.

Схема n-разрядного компаратора на равенство показана на рис.1, б.

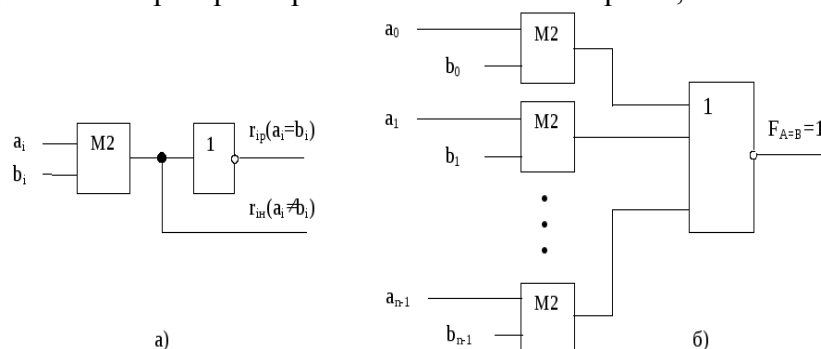


Рис. 1. Схема компаратора 2-х переменных (а) и n-разрядных слов (б)

Более сложные компараторы выявляют не только факт равенства двух n -разрядных чисел, но и сравнивают числа по значению. Такие компараторы имеют три выхода: “ $A > B$ ”, “ $A = B$ ”, “ $A < B$ ”, и в зависимости от соотношения величин A и B , активный уровень (уровень логической 1) появляется на одном из этих выходов.

Построить такой компаратор можно на базе двоичного сумматора, выполнив на нем операцию вычитания $A-B$ и проанализировав полученный результат. Для этого на сумматор нужно число B подать в дополнительном коде. Тогда выходной перенос сумматора (p_1) будет равен 0 лишь в том случае, когда A строго меньше B . Равенство разности 0 является признаком того, что $A=B$. Единица переноса при нулевой сумме указывает на то, что A строго больше B . Сказанное иллюстрируют следующие примеры:

$$\begin{array}{r} \text{A} > \text{B} \\ -\text{A} \quad -13 \quad +1101 \\ -\text{B} \quad -12 \quad \underline{+0100} \\ \hline \quad \quad \quad 1.0001 \\ \text{p}_i=1 \quad \text{s} \neq 0 \end{array} \qquad \begin{array}{r} \text{A} = \text{B} \\ -\text{A} \quad -12 \quad +1100 \\ -\text{B} \quad -12 \quad \underline{+0100} \\ \hline \quad \quad \quad 1.0000 \\ \text{s}=0 \end{array} \qquad \begin{array}{r} \text{A} < \text{B} \\ -\text{A} \quad -11 \quad +1011 \\ -\text{B} \quad -12 \quad \underline{+0100} \\ \hline \quad \quad \quad 0.1111 \\ \text{p}_i=0 \end{array}$$

Примечание. Вычитание из числа А числа $B=12_{10}=1100_2$ заменено прибавлением к А дополнительного кода числа В, равного 0100_2 .

Правила справедливы, если числа А и В рассматриваются, как положительные величины, без знака. Если же их старшие разряды трактуются как знаки, то правила будут несколько иные. Их легко вывести самостоятельно, если есть навыки обращения с обратными и дополнительными кодами.

Схема, реализующая описанный алгоритм, изображена на рис. 2.

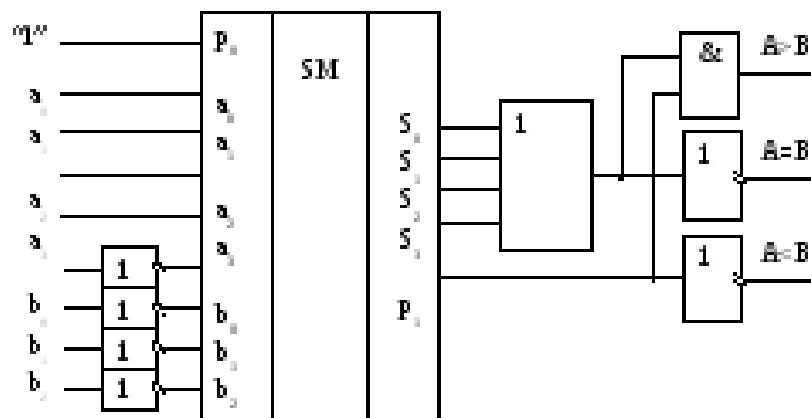


Рис. 2. Схема 4-разрядного компаратора на базе двоичного сумматора.



Примером компаратора двоично-кодированных чисел может служить ИС 4-разрядного компаратора К555СП1 (рис. 3).

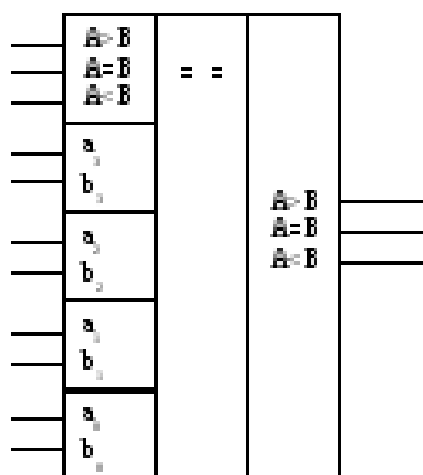


Рис. 3. Условное изображение ИС компаратора К555СП1

Компаратор имеет 11 входов. Четыре пары входов $a_i b_i$ ($i=0,1,2,3$) используются для подачи на них соответствующих разрядов сравниваемых чисел, входы $A < B$, $A = B$, $A > B$ позволяют каскадировать несколько ИС компараторов для увеличения разрядности сравниваемых чисел. Компаратор имеет три выхода результата сравнения: $A > B$, $A = B$ и $A < B$. При каскадировании выходы $A > B$, $A = B$ и $A < B$ схемы, сравнивающей младшие разряды, следует присоединить к одноименным входам последующего каскада. Этим способом с помощью двух компараторов СП1 можно сравнивать два восьмиразрядных слова. Нетрудно подсчитать необходимое число каскадов для любой большей длины сравниваемых слов.

Все возможные комбинации поразрядных соотношений входных кодов, а также уровней на входах каскадирования сведены в таблицу, где показаны соответствующие результирующие уровни на выходах $A > B$, $A = B$ и $A < B$ (табл. 1).

Таблица 1

	Входы сравнения данных				Входы наращивания каскадов			Выходы		
	a_3, b_3	a_2, b_2	a_1, b_1	a_0, b_0	$I(A>B)$	$I(A<B)$	$I(A=B)$	$A>B$	$A<B$	$A=B$
1.	$a_3>b_3$	x	x	x	x	x	x	В	Н	Н
2.	$a_3<b_3$	x	x	x	x	x	x	Н	В	Н
3.	$a_3=b_3$	$a_2>b_2$	x	x	x	x	x	В	Н	Н
4.	$a_3=b_3$	$a_2<b_2$	x	x	x	x	x	Н	В	Н
5.	$a_3=b_3$	$a_2=b_2$	$a_1>b_1$	x	x	x	x	В	Н	Н
6.	$a_3=b_3$	$a_2=b_2$	$a_1<b_1$	x	x	x	x	Н	В	Н
7.	$a_3=b_3$	$a_2=b_2$	$a_1=b_1$	$a_0>b_0$	x	x	x	В	Н	Н
8.	$a_3=b_3$	$a_2=b_2$	$a_1=b_1$	$a_0<b_0$	x	x	x	Н	В	Н
9.	$a_3=b_3$	$a_2=b_2$	$a_1=b_1$	$a_0=b_0$	Н	Н	В	Н	Н	В

Задание на практическую работу

1. На базе 2-х входных ЛЭ “М2”(К155ЛП5) спроектировать схему и исследовать работу n-разрядного компаратора на равенство, (n=2 для 1-й и 2-й бригад; n=3 для 3-й, 4-й и 5-й бригад).

Порядок выполнения пункта 1 задания. При фиксированном значении одного из сравниваемых чисел, например А, равном номеру вашей бригады, установить значение второго числа (В) равным, большим и меньшим А на единицу. Результаты сравнения свести в таблицу.

2. На базе 4-х разрядного двоичного сумматора спроектировать и исследовать работу компаратора с тремя выходами: $A=B$, $A>B$ и $A<B$.

Порядок выполнения пункта 2 задания. Четырехразрядный двоичный сумматор построить из двухразрядных сумматоров К155ИМ2.

При фиксированном значении одного из сравниваемых чисел, например А, равном номеру вашей бригады плюс 3, установить значение второго числа (В) равным, большим и меньшим А на 2. Результаты сравнения свести в таблицу.

3. Исследовать работу ИС 4-х разрядного компаратора К555СП1.

Порядок выполнения пункта 3 задания аналогичен порядку выполнения п. 2.2. Кроме того, для фиксации равенства $A=B$ на входы наращивания ($A>B$) и ($A<B$) следует подать напряжение низкого уровня, а на вход $I(A=B)$ – напряжение высокого уровня (см. строку 9 табл. 1).

Контрольные вопросы

1. Приведите определение цифрового компаратора и перечислите его возможные применения.

2. Запишите условия равенства (неравенства) одноименных разрядов сравниваемых чисел А и В.

3. Докажите справедливость выражений (1) и (2).

4. Используя 2-х входные ЛЭ “М2”, спроектируйте схему n-разрядного цифрового компаратора на равенство (неравенство). Определите необходимое для этого число ЛЭ “М2”, других ЛЭ, если n=2, 3, 4, 5.

5. Чему равно значение выхода схемы (рис. 1, б) при а) $A=B$, б) $A<B$ и в) $A>B$?

6. Выполните требования п. 2.1 задания на лабораторную работу, если в вашем распоряжении имеются 2-х входные ЛЭ “М2” с инверсными выходами.

7. Используя ИС К555СП1 спроектируйте схему 8-ми разрядного цифрового компаратора.

Практическая работа №18. Разработка и моделирование процессов передачи информации через драйверные схемы

Теоретическая часть

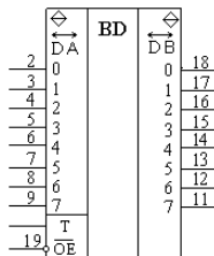
Драйвер – интегральная комбинационная схема с повышенной нагрузочной способностью, назначением которой является организация связи с периферийными устройствами. Если связь осуществляется с помощью двунаправленной шины данных, то в качестве драйвера используется схема с Z-состоянием.

Шинные драйверы обозначаются на схемах как BD.

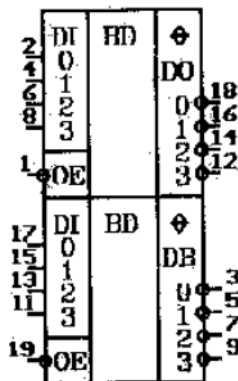
Рассмотрим, к примеру, работу шестиразрядного шинного драйвера серии 155 ЛП 10. Данные поступают на входы DI 0-5 и передаются на выходы DO 0-5 в инвертированном виде при условии, что на входах разрешения присутствуют сигналы низкого уровня. В остальных случаях выходы драйвера переводятся в третье состояние.

Таблица истинности:

OE	T	Направление передачи
0	0	DB → DA
0	1	DA → DB
1	X	Z-состояние



Примеры схем шинных драйверов ТТЛ-технологии



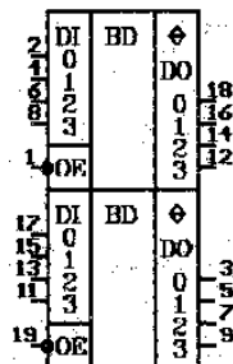
Два четырехразрядных инвертирующих шинных драйвера

530-, 531-, 533-, 555-, 1533АПЗ

(10—общ., 20 — +5 В).

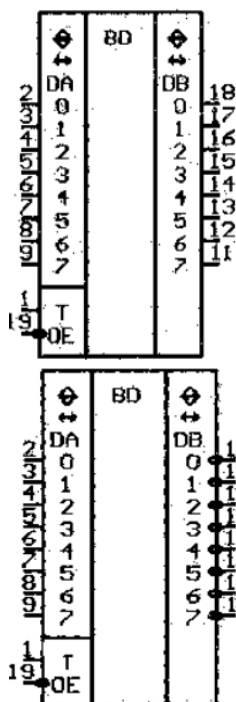
Управление ИС аналогично ЛП8.

При подаче на вход OE=0 происходит передача информации, при OE=1 выходы схемы переводятся в z-состояние.



Два четырехразрядных шинных драйвера 533-, 555-, 1533АП5 (10—общ., 24— +5 В).

Управление ИС аналогично ЛП8.



Восьмиразрядный шинный драйвер 533-, 555-, 1533АП6 (10—общ., 24—+5 В).
Функционирование ИС задается в соответствии с таблицей режимов:

OE	T	Направление передачи
0	0	DB → DA
0	1	DA → DB
1	x	Z-состояние

Восьмиразрядный инвертирующий шинный драйвер 1533АП9 (10—общ., 20—+5 В).

Функционирование ИС задается в соответствии с таблицей режимов:

OE	T	Направление передачи
0	0	$\overline{DB} \rightarrow DA$
0	1	$DA \rightarrow \overline{DB}$
1	x	Z-состояние

Практическое задание

Смоделировать работу драйверной схемы по заданным условиям любыми программными средствами. Построить временную диаграмму и таблицу истинности работы полученной схемы.

Вариант 1. Схема 155 ЛП 10. Режим параллельной передачи 18-разрядного числа.

Вариант 2. Схема на базе 1533 АП 3. Режим работы – асинхронная передача двух 8-разрядных чисел.

Вариант 3. Схема на базе 1533 АП 5. Организовать асинхронную передачу 12-разрядного и 4-разрядного данного.

Вариант 4. На базе 1533 АП 6 организовать двунаправленную передачу 32-разрядного данного.

Вариант 5. На базе 1533 АП 9 построить схему асинхронной передачи двух 8-разрядных чисел.

Вариант 6. На базе 155 ЛП 10 построить схему асинхронной передачи трех 6-разрядных чисел.

Вариант 7. Схема на базе 555 АП 3. Организовать передачу 20-разрядного адреса.

Вариант 8. На базе схем 555 АП 5 организовать асинхронную передачу трех 8-разрядных данных.

Вариант 9. На базе 533 АП 6 построить схему асинхронной двунаправленной передачи двух 16-разрядных данных.

Вариант 10. Схема на базе 1533 АП 9. Организовать 16-разрядную шину адреса.

Практическая работа №19. Исследование принципов построения и функционирования цифровых автоматов.

Цель работы: Изучение принципов синтеза цифровых автоматов на основе их исходного описания.

Теоретическая часть

Цифровой автомат (ЦА) с памятью – это любое последовательностное логическое устройство. Цифровой автомат в общем случае содержит N триггеров, поэтому его состояния характеризуются N -разрядным двоичным словом. В таком случае ЦА может находиться в 2^N

состояниях. В общем случае ЦА автомат может быть представляет собой систему состоящую из двух подсистем: комбинационной и триггерной. Структура ЦА представлена на рисунке 2.1.

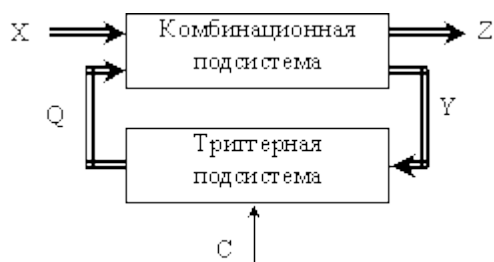


Рисунок 2.1. Структурная схема цифрового автомата с памятью.

Поведение ЦА характеризуется пятью типами сигналов:

X – входное кодовое слово;

Z – выходное кодовое слово;

Y – управляющее кодовое слово, обеспечивающее требуемый порядок смены состояний

ЦА;

Q – кодовое слово, характеризующее состояние ЦА;

C – сигнал синхронизации, инициирующий процесс переключения триггеров.

Исходное описание работы ЦА может быть задано одним из трех способов: словесным описанием, таблицей состояний или графом переходов. Все формы могут взаимно трансформироваться одна в другую, при этом наиболее лаконичными и наиболее удобными для синтеза являются таблица состояний и граф переходов.

В общем случае правила синтеза ЦА формулируются следующим образом:

1. По условию работы ЦА определяется число необходимых состояний и требуемый объем его триггерной подсистемы;

2. Формализуют описание ЦА путем составления таблицы состояний или графа переходов;

3. Выбирают тип триггера для реализации триггерной подсистемы;

4. Используя формализованный алгоритм работы ЦА (результат п.2) и таблицу истинности выбранного типа триггера составляют расширенную таблицу переходов;

5. Из расширенной таблицы переходов получают функции алгебры логики (ФАЛ), описывающие комбинационную подсистему и минимизируют их;

6. По полученным ФАЛ строят логическую схему ЦА.

Рассмотрим приведенные пункты более подробно.

При словесном способе описания ЦА как правило необходимо определить число его состояний. Оно может быть определено как максимальное число комбинаций выходного кода, которые могут быть получены при одном значении входного сигнала. Из этого показателя по формуле (2.1) определяется объем триггерной подсистемы.

$$n \geq \log_2^M \quad (2.1)$$

где: M – число необходимых состояний ЦА;

n – количество триггеров.

Параметр n должен быть получен путем округления в большую сторону до ближайшего целого значения выражения (2.1). При использовании таблицы состояний или графа переходов объем триггерной подсистемы уже известен.

Таблица состояний и граф переходов отражают картину перехода автомата из одного состояния в другое под воздействием входных сигналов и показывают значения выходных сигналов.

При заполнении таблицы состояний на пересечении j-го столбца и I-той строки записывается дробь. В числителе указывается состояние в которое попадает ЦА после прихода очередного импульса синхронизации C, если он находился в i-том состоянии и на его входе действовал j-тый входной сигнал. В знаменателе указывается текущее значение выходного сигнала, существующее в ЦА до прихода очередного импульса C при нахождении его в том

состоянии при действии j-того входного сигнала. Пример таблицы переходов представлен в таблице 2.1.

Таблица 2.1. Таблица состояний цифрового автомата.

S	X	
	0	1
00	01/0	11/0
01	10/1	00/0
10	11/0	01/0
11	00/1	10/1

Граф переходов представляет собой графическую интерпретацию работы цифрового автомата. Каждое состояние ЦА изображается в виде окружности с указанием кода соответствующего состояния. Переход из одного состояния в другое изображается в виде стрелки. Над стрелкой записывается дробь. В числителе указывается значение входного сигнала под действием которого будет происходить переход. В знаменателе записывается текущее значение выходного сигнала, соответствующее указанному (исходному) состоянию и значению входного сигнала. Граф переходов, соответствующий ЦА, описанному таблице 2.1. представлен на рисунке 2.2.

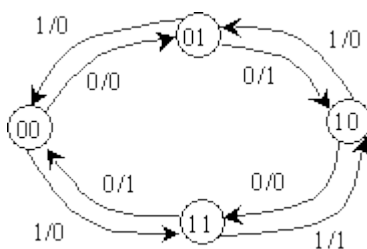


Рисунок 2.2. Граф переходов цифрового автомата.

Для реализации триггерной подсистемы при синтезе ЦА выбирают синхронный RS, T или JK триггеры. Возьмем для реализации описанного ЦА JK триггер. Таблица истинности этого типа триггера приведена в таблице 2.2.

Таблица 2.2. Таблица истинности JK триггера.

C	J	K	Q _n	Q _{n+1}
0	*	*	*	Q _n
1	*	*	*	Q _n
/	0	0	0	0
/	0	0	1	1
/	0	1	0	0
/	0	1	1	0
/	1	0	0	1
/	1	0	1	1
/	1	1	0	1
/	1	1	1	0

Символом «/» в данной таблице отмечен переход сигнала из состояния Лог.0 в состояние Лог.1. Из таблицы состояний и графа переходов видно, что объем триггерной подсистемы равен 2, а количество состояний – 4. Руководствуясь формальным описанием ЦА, заданным в виде таблицы состояний или графа переходов и таблицы истинности для выбранного типа триггеров построим расширенную таблицу переходов. Данная таблица представлена в таблице 2.3.

Таблица 2.3. Расширенная таблица переходов цифрового автомата.

X	Q _{1n}	Q _{0n}	Q _{1n+1}	Q _{0n+1}	J1	K1	J0	K0	Z
1	2	3	4	5	6	7	8	9	10
0	0	0	0	1	0	*	1	*	0
0	0	1	1	0	1	*	*	1	1
0	1	0	1	1	*	0	1	*	0
0	1	1	0	0	*	1	*	1	1
1	0	0	1	1	1	*	1	*	0
1	0	1	0	0	0	*	*	1	0
1	1	0	0	1	*	1	1	*	0
1	1	1	1	0	*	0	*	1	1

В столбце 1 приведены значения входного сигнала X, воздействующего на ЦА, пребывающий в состоянии, отраженном в столбцах 2 и 3. В результате этого на выходе ЦА появляется сигнал Z и автомат, под действием сигнала синхронизации, переходит в новое состояние, отраженное в столбцах 4 и 5. В ЦА используются два триггера, каждый из них имеет входы J, K, C. На вход C необходимо подавать синхросимпульсы, под действием которых автомат будет переходить в новые состояния. На входы J и K должны быть поданы сигналы, обеспечивающие переход триггеров из состояния Q_n в состояние Q_{n+1}. Для определения значений управляющих сигналов необходимо по таблице истинности триггера определить условия перехода его из исходного состояния в новое. Так в первой строке таблицы 2.3. показано, что триггер T1 должен перейти из состояния 0 в состояние 0. По таблице истинности триггера (таблица 2.2.) находим строки, в которых отражен данный переход. Таких строки две. Рассмотрим их отдельно по таблице 2.4.

Таблица 2.4. Условия перехода триггера в новое состояние.

C	J	K	Q _n	Q _{n+1}
/	0	0	0	0
/	0	1	0	0

Из таблицы 2.4. видно, что для обеспечения данного перехода необходимо обеспечить на входе J присутствие Лог.0, состояние же на входе K – безразлично. Данный факт отмечается символом «*». Подобным образом производится анализ условий переключения триггеров для всех остальных ситуаций.

Столбцы 6 – 10 описывают поведение соответствующих выходов комбинационной подсистемы. Фактически комбинационная подсистема представляет логическое устройство со многими выходами. Для его синтеза возможно применение соответствующих методик, учитывающих взаимовлияние функциональных выражений. Однако, для упрощения процедуры синтеза будем рассматривать каждую из функций как независимую от других. Поэтому минимизацию их будем проводить индивидуально.

Для минимизации ФАЛ воспользуемся картами Карно. Аргументами всех отмеченных выше функций являются входные сигналы и исходные состояния автомата (столбцы 1, 2, 3).

Функция Z полностью определена на всех наборах. Соответствующая ей карта Карно приведена на рисунке 2.3.

	Q_{1n}		$\overline{Q_{1n}}$	
X	0	1	0	0
$\overline{X}$	0	1	1	0
	$\overline{Q_{0n}}$		Q_{0n}	$\overline{Q_{0n}}$

Рисунок 2.3. Минимизация функции Z.

Из карты Карно выписываем две импликанты, в соответствии с выделенными областями. Минимальная ДНФ для функции Z будет иметь вид:

$$Z = X\overline{Q_{0n}} \vee Q_{1n}\overline{Q_{0n}} \quad (2.2)$$

Функции J1, J0, K1, K0 являются недоопределенными. Поэтому при минимизации символы «*» могут быть заменены на значения Лог.0 или Лог.1. Карты Карно для функций J1, J0, K1, K0 приведены на рисунках 2.4, 2.5, 2.6, 2.7 соответственно.

	Q_{1n}		$\overline{Q_{1n}}$	
X	*1	*0	0	1
$\overline{X}$	*0	*1	1	0
	$\overline{Q_{0n}}$		Q_{0n}	$\overline{Q_{0n}}$

Рисунок 2.4. Минимизация функции J1.

В клетки, содержащие символы «*» вписываются символы «0» или «1» для достижения минимальной реализации функции. Минимальная ДНФ для функции J1 будет иметь вид:

$$J1 = \overline{X}\overline{Q_{0n}} \vee \overline{X}Q_{0n} \quad (2.3)$$

	Q_{1n}		$\overline{Q_{1n}}$	
X	1	0	*0	*1
$\overline{X}$	0	1	*1	*0
	$\overline{Q_{0n}}$		Q_{0n}	$\overline{Q_{0n}}$

Рисунок 2.5. Минимизация функции K1.

Для функции K1 вид карты Карно аналогичен карте для функции J1. Минимальная ДНФ для функции K1 будет иметь вид:

$$K1 = X\overline{Q_{0n}} \vee \overline{X}Q_{0n} \quad (2.4)$$

	Q_{1n}		$\overline{Q_{1n}}$	
X	1	* 1	* 1	1
$\overline{X}$	1	* 1	* 1	1
	$\overline{Q_{0n}}$		Q_{0n}	$\overline{Q_{0n}}$

Рисунок 2.6. Минимизация функции J0.

Заменяя в карте Карно символы «*» на символ «1» получаем карту содержащую 1 во всех клетках. Данная карта соответствует функции «Константа 1». Поэтому Минимальная ДНФ для функции J0 будет иметь вид:

$$J0 = 1 \quad (2.5)$$

Подобное преобразование справедливо и для функции K0. Она также вырождается в функцию «Константа 1».

	Q_{1n}		$\overline{Q_{1n}}$	
X	1	* 1	* 1	1
$\overline{X}$	1	* 1	* 1	1
	$\overline{Q_{0n}}$		Q_{0n}	$\overline{Q_{0n}}$

Рисунок 2.7. Минимизация функции K0.

Минимальная ДНФ для функции K0 имеет вид:

$$K0 = 1 \quad (2.5)$$

Функциональные выражения (2.3) и (2.4) представим в виде функции «Исключающее ИЛИ». Получив минимальные ФАЛ комбинационной подсистемы возможно построение функциональной схемы цифрового автомата. Данная схема приведена на рисунке 2.8.

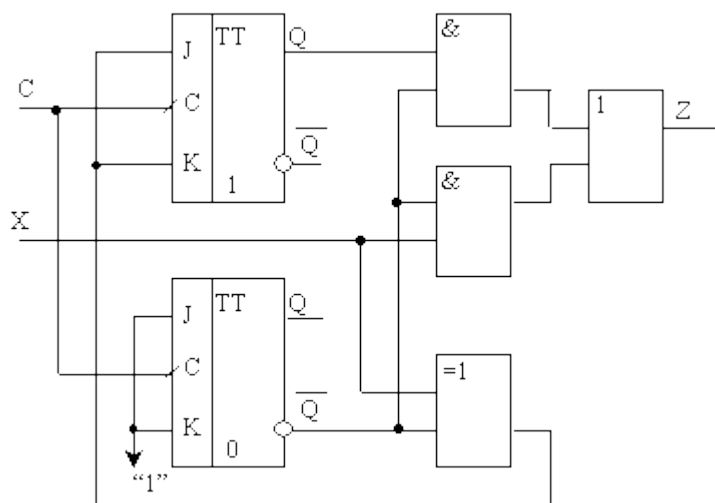


Рисунок 2.8. Функциональная схема цифрового автомата.

Задание на практическую работу

1. По исходному описанию, представленному в табл. 2.5 (вариант указывается преподавателем), построить граф переходов.
2. В соответствии с заданным по варианту типом триггера построить расширенную таблицу переходов.
3. Произвести минимизацию ФАЛ, описывающих работу комбинационной подсистемы.
4. Начертить функциональную схему ЦА.

Таблица 2.5. Варианты индивидуальных заданий.

№ варианта	S	X		Тип триггера
		0	1	
1	00	01/0	10/1	JK
	01	10/0	11/0	
	10	11/1	00/1	
	11	00/0	10/1	
2	00	01/0	10/1	T
	01	10/0	11/0	
	10	11/1	00/1	
	11	00/0	10/1	
3	00	01/0	11/1	JK
	01	10/1	00/0	
	10	11/1	11/1	
	11	00/0	01/0	
4	00	01/0	11/1	T
	01	10/1	00/0	
	10	11/1	11/1	
	11	00/0	01/0	
5	00	11/0	01/0	JK
	01	10/1	11/1	
	10	00/0	00/0	
	11	01/0	10/1	
6	00	11/0	01/0	T
	01	10/1	11/1	
	10	00/0	00/0	
	11	01/0	10/1	
7	00	01/0	11/1	JK
	01	00/0	11/1	
	10	01/1	11/0	
	11	01/0	10/1	
8	00	01/0	11/1	T
	01	00/0	11/1	
	10	01/1	11/0	
	11	01/0	10/1	

Содержание отчета

Отчет должен содержать:

1. Исходную таблицу состояний.
2. Граф переходов.
3. Таблицу истинности указанного типа триггера.
4. Расширенную таблицу состояний.
5. Процедуры минимизации ФАЛ комбинационной подсистемы ЦА.
6. Функциональную схему реализации цифрового автомата.

7. Выводы по выполненной работе.

Контрольные вопросы

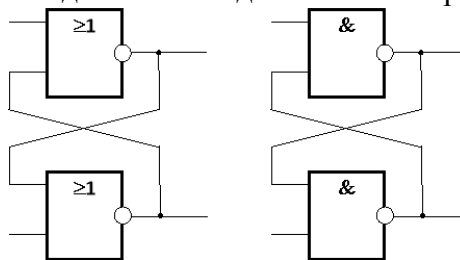
1. Каким образом определить требуемый объем триггерной подсистемы цифрового автомата?
2. Что указывается в надписях к дугам в графе переходов?
3. В чем отличие выходных сигналов Z и Y комбинационной подсистемы?
4. Что является инициирующим событием для переключения цифрового автомата?
5. Каким образом определяется поведение выходных сигналов группы Y комбинационной подсистемы?
6. Каким образом из расширенной таблицы переходов получается функциональная схема цифрового автомата?

Практическая работа №20. Моделирование и исследование логики работы асинхронных RS-триггеров.

Цель работы: ознакомление с принципом работы асинхронных RS-триггеров, получение практических навыков в построении и контроле работоспособности асинхронных RS-триггеров, а также исследование логики работы асинхронных RS-триггеров методом моделирования с использованием программы Electronics Workbench.

Теоретическая часть

Для построения цифровых устройств кроме логических элементов требуются элементы памяти, предназначенные для хранения двоичных кодов в течение требуемого времени.



В качестве статического элемента памяти используются бистабильные ячейки (БЯ), имеющие два устойчивых состояния. Бистабильные ячейки могут быть построены на двух логических элементах И-НЕ или ИЛИ-НЕ, соединенных перекрестными связями (см. рисунок 1).

В качестве элементов памяти используются так называемые триггеры. Триггер - это цифровая электронная схема с двумя устойчивыми состояниями, которые устанавливаются при подаче соответствующей комбинации входных сигналов и сохраняются после снятия этих сигналов.

Триггер имеет несколько входов и два выхода - прямой и инверсный. Сигналы на выходах триггера всегда имеют различные значения. Если на прямом выходе сигнал равен 1, то на инверсном - 0 и наоборот. Состояние триггера определяется значением сигнала на прямом выходе (Q). Если сигнал на прямом выходе равен 1, то триггер находится в состоянии 1.

Триггеры могут быть синхронными или асинхронными. Если изменения сигнала Q происходит только при наличии специального сигнала С, являющегося сигналом синхронизации, то такой триггер называется синхронным триггером. Синхронизация триггера может происходить либо по уровню сигнала, либо по фронту сигнала (переднему или заднему).

Асинхронный триггер не имеет входа синхронизации, поэтому переключение триггера происходит только при поступлении на вход информационных входных сигналов X.

Логика переключения триггера из одного состояния в другое зависит от количества и назначения входов. Наиболее часто используются в цифровой технике следующие типы триггеров: RS-триггеры, JK-триггеры, D-триггеры и T-триггеры. Буквами R, S, J, K, D и T обозначаются информационные входы триггеров (X).

Асинхронные и синхронные триггеры разных типов

Асинхронные RS-триггеры

Асинхронный RS-триггер имеет два информационных входа - R и S. Вход S используется для установки триггера в состояние 1, а вход R - для установки в состояние 0.

Работа триггера описывается таблицей переходов, которая имеет вид таблицы 1.

Таблица 1

Входы		Состояния	
R	S	Q(0)	Q(1)
0	0	0	1
0	1	1	1
1	0	0	0
1	1	Не определено	

Из таблицы 1 может быть получено уравнение переходов триггера. После минимизации (например, с использованием карт Карно) уравнение переходов примет вид:

$$Q_{t+1} = S + \bar{R} \cdot Q_t$$

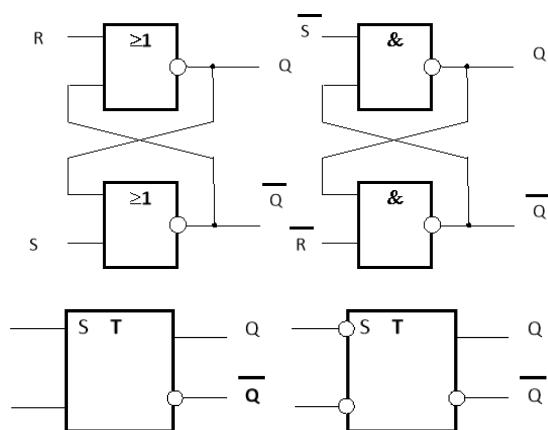


Рисунок 3

Из уравнения следует, что при S=1, R=0 всегда $Q_{t+1}=1$, при S=0, R=1 всегда $Q_{t+1}=0$, а при S=0, R=0 $Q_{t+1}=Q_t$. Комбинация сигналов S=1, R=1 является запрещенной, так состояние триггера не определено.

Для построения триггера на элементах И-НЕ уравнение необходимо преобразовать (двойным инвертированием) к другому виду:

$$Q_{t+1} = \overline{\overline{S} \cdot \overline{R} \cdot Q_t}$$

Для построения триггера на элементах ИЛИ-НЕ уравнение имеет вид:

$$Q_{t+1} = \overline{\overline{S + R + Q_t}}$$

Функциональные схемы асинхронных RS-триггеров, построенные на элементах ИЛИ-НЕ (слева) и И-НЕ (справа), и их условные графические обозначения (УГО) показаны на рисунке 3.

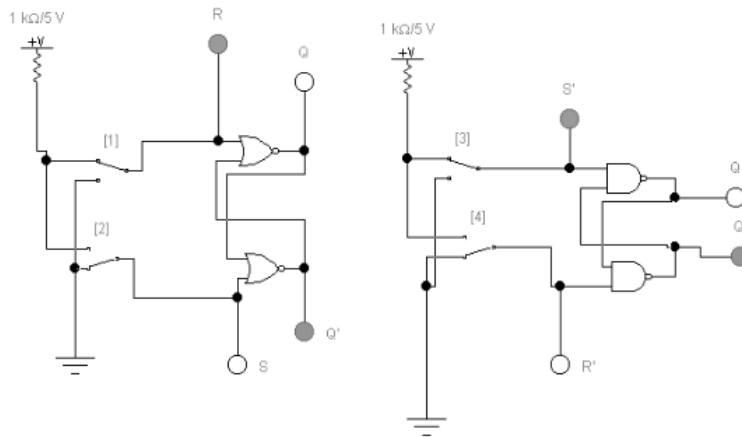
Как видно из рисунка 3, асинхронный RS-триггер представляет собой бистабильную ячейку, построенную на элементах И-НЕ или ИЛИ-НЕ.

При построении RS-триггера на элементах И-НЕ действующими установочными сигналами являются инверсные значения информационных сигналов R и S.

Порядок выполнения работы

Построить на элементах 2И-НЕ и 2ИЛИ-НЕ схемы асинхронных RS-триггеров (см. рисунок 3) и исследовать логику их работы в статическом режиме.

Путем моделирования работы триггеров получить таблицы переходов и сравнить их с таблицей 1. Образцы схем для моделирования приведены ниже. Исследуемые схемы и таблицы занести в отчет.



Содержание отчета

В отчет о выполненной работе включить следующие материалы:

1. Тему и цель работы.
2. Результаты выполнения заданий: исследуемые схемы, полученные таблицы переходов.
3. Анализ полученных результатов.
4. Выводы по работе.

Контрольные вопросы

1. Из каких логических элементов можно построить схему триггера?
2. Чем отличаются синхронные триггеры от асинхронных триггеров?

Практическая работа №21. Моделирование и исследование логики работы синхронных RS-триггеров.

Цель работы: ознакомление с принципом работы синхронных триггеров, получение практических навыков в построении и контроле работоспособности синхронных триггеров, а также исследование логики работы синхронных триггеров методом моделирования с использованием программы Electronics Workbench.

Теоретическая часть

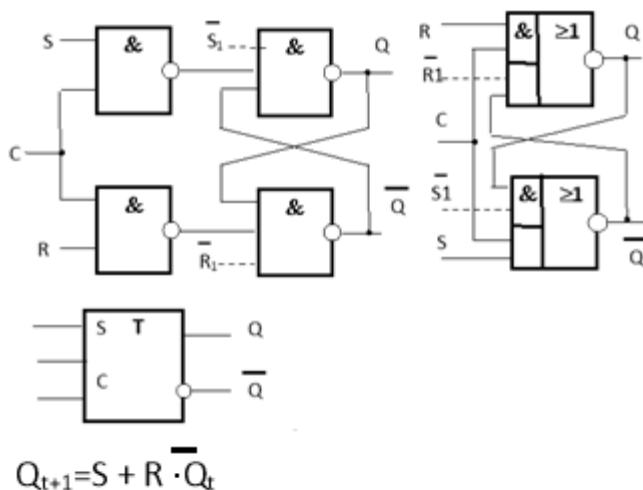
Синхронный RS-триггер дополнительно имеет вход синхронизации С, на который поступает синхросигнал. Информационные сигналы R и S воздействуют на состояние триггера только при значении синхросигнала $C=1$.

Таблица переходов синхронного RS-триггера состоит из двух частей. Первая часть таблицы описывает переходы триггера при $C=1$ и совпадает с таблицей переходов асинхронного триггера, а вторая – при $C=0$.

При $C=0$ триггер не меняет своего состояния при любой комбинации сигналов на информационных входах R и S. В этом случае всегда $Q_{t+1} = Q_t$.

Уравнение синхронного RS-триггера имеет вид:

$$Q_{t+1} = S \cdot C + (R + C) \cdot Q_t$$



Из уравнения следует, что при $C=0$ $Q_{t+1}=Q_t$, т.е. работа описывается уравнением асинхронного триггера. На рисунке приведены функциональные схемы синхронных RS-триггеров, реализованных на элементах И - НЕ для уравнения

$$\overline{Q_{t+1}} = \overline{S \cdot C + (R \cdot C) \cdot Q_t}$$

и на элементах И-ИЛИ-НЕ для уравнения

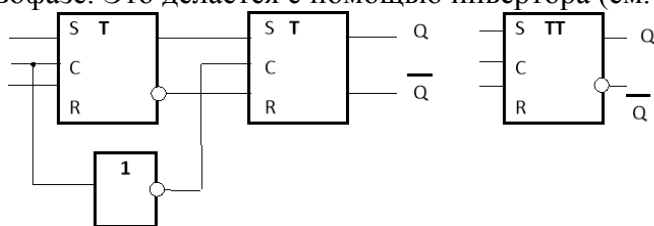
$$\overline{Q_{t+1}} = \overline{S \cdot C + (R \cdot C) \cdot Q_t}$$

На рисунке, кроме основных входов R и S, показаны дополнительные инверсные асинхронные входы R_1 и S_1 .

Двухтактные RS-триггеры

Триггеры в ЭВМ используются в различных узлах, между которыми осуществляется передача информации. Устойчивая работа цепочки триггеров возможна только в том случае, если запись новой информации в триггер осуществляется после считывания ранее записанной информации и передачи её в следующий по цепочке триггер. Это возможно при использовании двух серий синхроимпульсов, сдвинутых относительно друг друга на 180° . Такой принцип управления и синхронизации применяется в двухтактных триггерах.

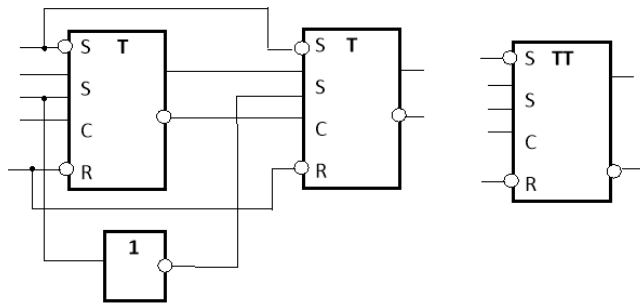
Простейшая схема двухтактного RS-триггера может быть построена на двух одноклапных триггерах, причём синхроимпульсы на входы C первого и второго триггеров должны подаваться в противофазе. Это делается с помощью инвертора (см. рисунок).



При поступлении на вход первого одноклапного триггера импульса $C=1$ информация на входах R и S устанавливает триггер в соответствующее новое состояние Q_{t+1} , а второй одноклапный триггер хранит информацию о предыдущем состоянии Q_t , так как на его входе C сигнал равен нулю. По окончании действия синхроимпульса, т.е. при $C=0$, первый триггер переходит в режим хранения, а информация Q_{t+1} , записанная в первом триггере, передается во второй, так как на его входе C сигнал становится равным единице. В результате к началу следующего такта на выходе двухтактного RS-триггера появится сигнал, определяемый состоянием Q_{t+1} первого триггера. В таком триггере выходной сигнал формируется по заднему фронту синхроимпульса.

Двухтактный синхронный RS-триггер может быть использован для построения других типов триггеров, таких как D-, T- и JK-триггеров.

Для установки RS-триггера в 0 или 1 независимо от присутствия сигнала на входе C в схему вводят прямые или инверсные входы R и S асинхронной установки, как показано на следующем рисунке:



Практическая часть

Задание 1. Построить на элементах 2И-НЕ и 2-2И-2ИЛИ-НЕ схемы синхронных RS-триггеров и исследовать логику их работы в статическом режиме. Образцы схем для моделирования приведены на рисунках 1 и 2. В качестве элементов 2-2И-2ИЛИ-НЕ использована микросхема 7455, в которой располагается элемент 4-4И-2ИЛИ-НЕ. Исследуемые схемы и таблицы занести в отчет.

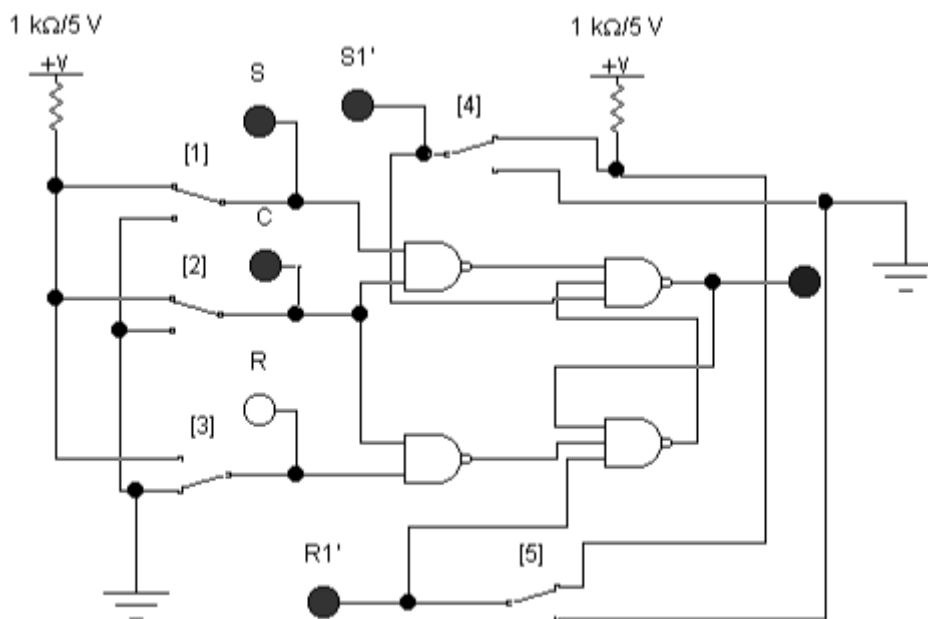


Рисунок 1

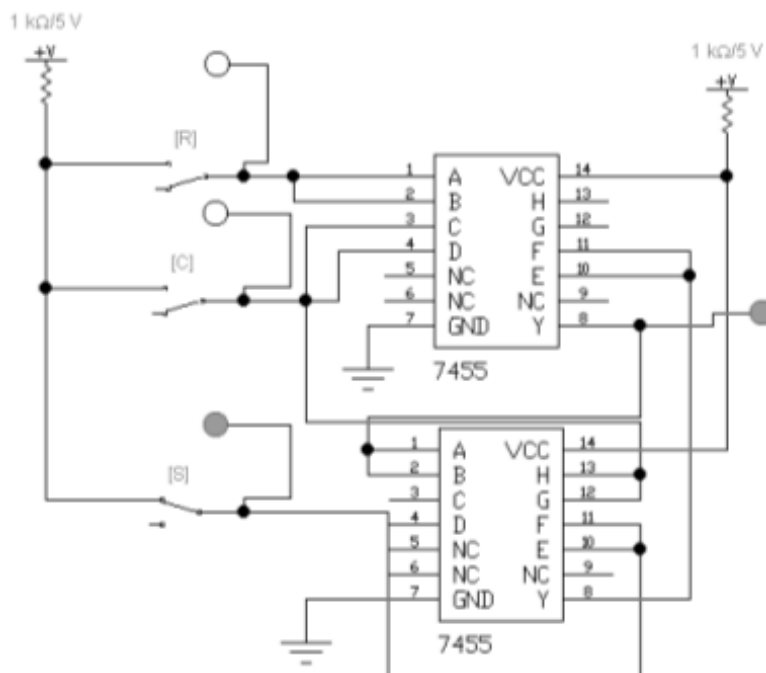


Рисунок 2

Задание 2. Исследовать в статическом режиме логику работы RS-триггера, который имеется в библиотеке программы. Для этого собрать схему, показанную на рисунке 3. Получить таблицу переходов триггера и сравнить ее с таблицей 1. Исследуемую схему и таблицу занести в отчет.

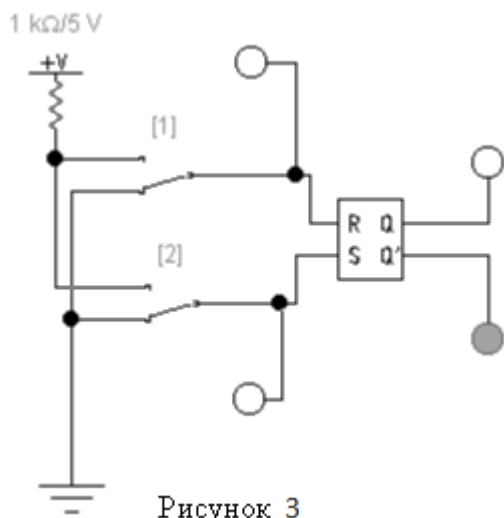


Рисунок 3

Задание 3. Исследовать в статическом режиме логику работы двухтактного RS-триггера. Для этого собрать схему, показанную на рисунке 4. Получить таблицу переходов триггера и сравнить ее с таблицей 1. Исследуемую схему и таблицу занести в отчет.

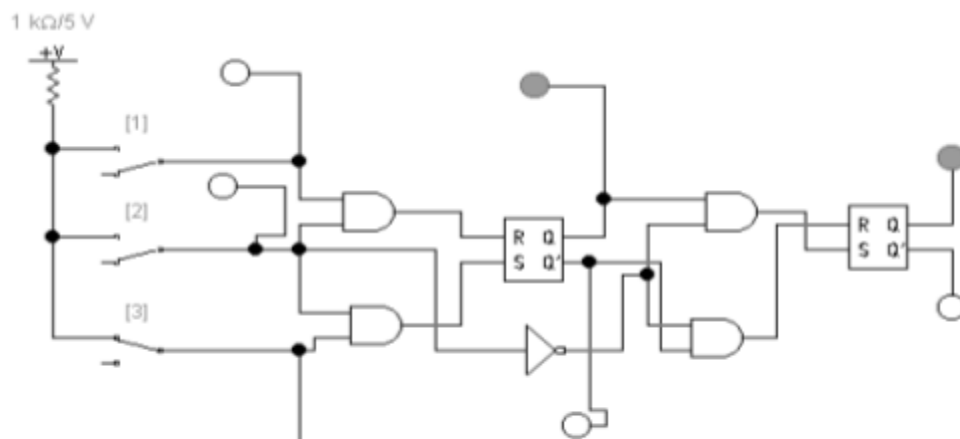


Рисунок 4

Содержание отчета

В отчет о выполненной работе включить следующие материалы:

1. Тему и цель работы.
2. Результаты выполнения заданий: исследуемые схемы, полученные таблицы переходов.
3. Анализ полученных результатов.
4. Выводы по работе.

Контрольные вопросы

1. Из каких логических элементов можно построить схему триггера?
2. Чем отличаются синхронные триггеры от асинхронных триггеров?

Цель работы: ознакомление с принципом работы D-триггеров, получение практических навыков в построении и контроле их работоспособности, а также исследование логики работы D-триггеров методом моделирования с использованием программы Electronics Workbench.

Теоретическая часть

Асинхронный и синхронный D-триггеры

В вычислительной технике широко применяется D-триггер, который реализует функцию временной задержки входного сигнала. D-триггер имеет один информационный вход. Логика работы асинхронного D -триггера описывается таблицей переходов, которая имеет вид таблицы 1. По таблице 1 может быть записано уравнение переходов D-триггера:

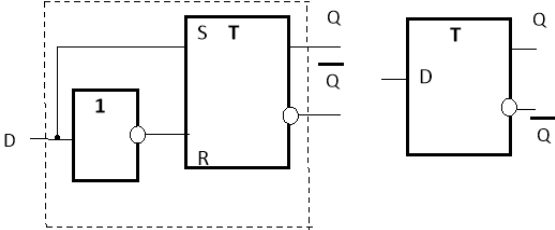
$Q_{t+1} = D_t$,
 где: t - текущий момент времени; t+1 - последующий момент времени.

Таблица 1

Вход D	Состояния	
	Q(0)	Q(1)
0	0	0
1	1	1

Как видно из уравнения, в асинхронном D-триггере состояние (выходной сигнал) Q_{t+1} повторяет значение входного сигнала D_t . Поэтому асинхронный D-триггер по существу является не элементом памяти, а элементом задержки, и рассматривается только как основа для построения синхронного D-триггера.

Функциональная схема и УГО асинхронного D-триггера, построенного на основе асинхронного RS-триггера, показаны на следующем рисунке:



Для построения счётчиков, регистров и других цифровых схем используются синхронные D-триггеры как одноктактные, так и двухтактные. Логика работы синхронного D-триггера описывается таблицей переходов, которая имеет вид таблицы 2.

Таблица 2

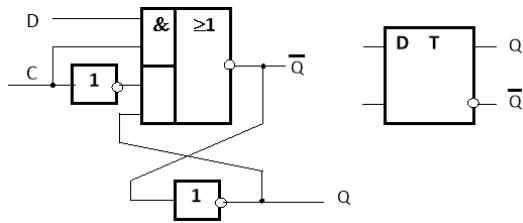
Входы		Состояния	
C	D	Q(0)	Q(1)
1	0	0	0
1	1	1	1
0	0	0	1
0	1	0	1

Уравнение переходов синхронного триггера, записанное по таблице 3, имеет следующий вид:

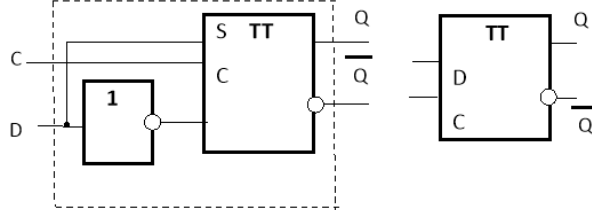
$Q_{t+1} = C \cdot Q_t + C \cdot D$

В соответствии с уравнением синхронный D-триггер при C=0 сохраняет свое состояние, а при C=1 работает как асинхронный.

Функциональная схема синхронного D-триггера на элементах ИЛИ-НЕ приведена на следующем рисунке:

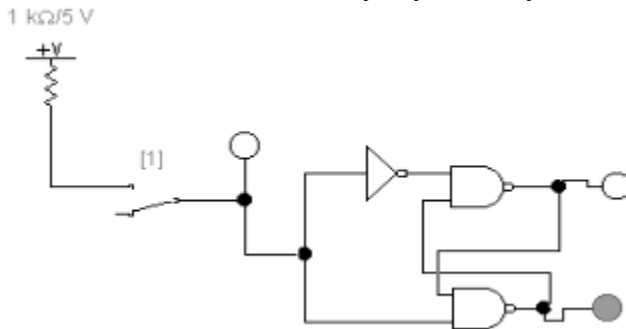


Функциональная схема двухтактного D-триггера, построенного на основе двухтактного RS-триггера, приведена на следующем рисунке:

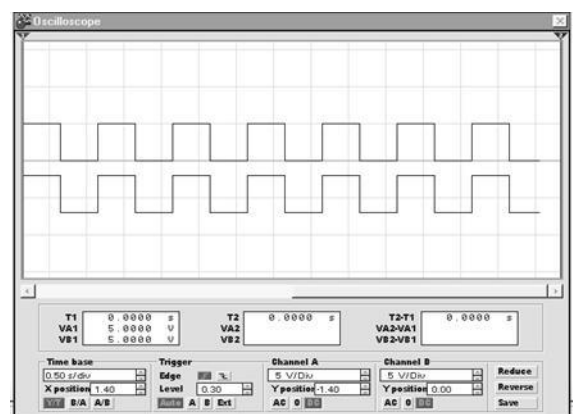
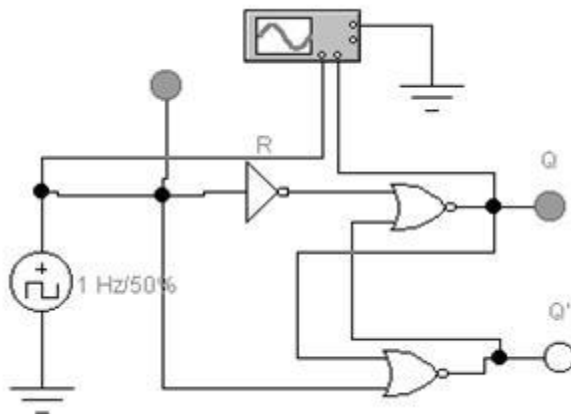


Практическая часть

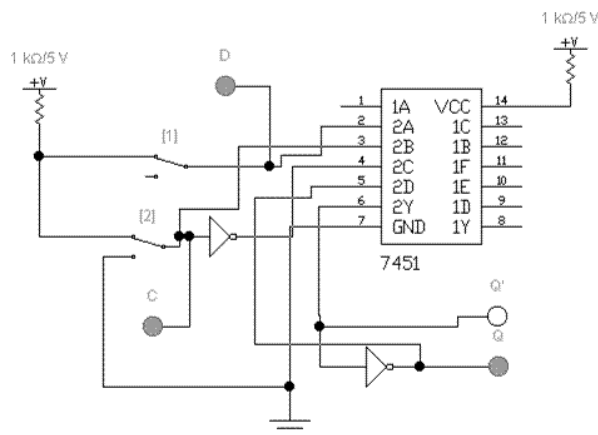
Задание 1. Исследовать в статическом режиме логику работы асинхронного D-триггера. Для этого собрать схему, показанную на рисунке. Получить таблицу переходов триггера и сравнить ее с таблицей 2. Исследуемую схему и таблицу занести в отчет.



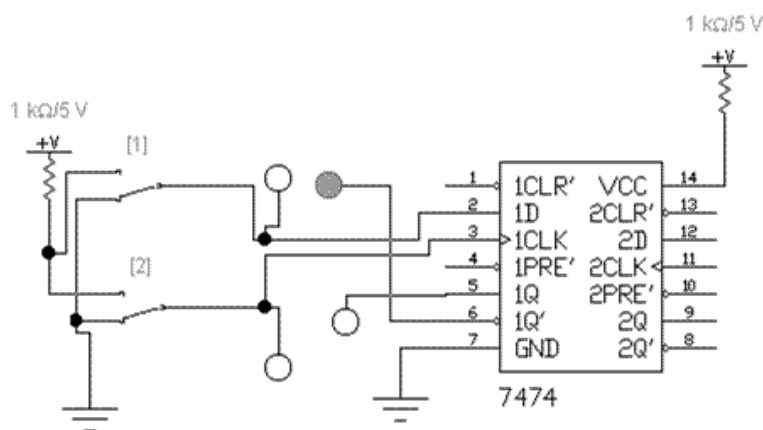
Задание 2. Исследовать в динамическом режиме логику работы асинхронного D-триггера. Для этого собрать схему, показанную на рисунке. Для визуального наблюдения работы схемы установить частоту генератора 1 Гц. Зарисовать полученную осциллограмму. Исследуемую схему и таблицу занести в отчет.



Задание 3. Собрать и исследовать в статическом режиме схему синхронного D-триггера на элементе 2И-2И-2ИЛИ-НЕ, в качестве которого использовать микросхему 7451 с 2-мя элементами 2И-2И-2ИЛИ-НЕ. Схема для исследования показана на рисунке ниже. Результаты исследования занести в отчет.



Задание 4. Собрать и исследовать микросхему 7474, состоящую из 2-х синхронных D-триггеров. Схема показана на рисунке ниже. Результаты исследования занести в отчет.



Содержание отчета

В отчет о выполненной работе включить следующие материалы:

1. Тему и цель работы.
2. Результаты выполнения заданий: исследуемые схемы, полученные таблицы переходов.
3. Анализ полученных результатов.
4. Выводы по работе.

Контрольные вопросы

1. Можно ли построить схему D-триггера на основе RS - триггера ?
2. На какое время может быть задержана установка синхронного D-триггера по отношению к сигналу на его входе?
3. На какое время может быть задержана установка в 1 асинхронного D-триггера по отношению к сигналу на его входе?
4. Чем отличается двухтактный триггер от одноктактного триггера?

Практическая работа №23. Моделирование и исследование логики работы Т-триггеров.

Цель работы: ознакомление с принципом работы Т-триггеров, получение практических навыков в построении и контроле их работоспособности, а также исследование логики работы Т-триггеров методом моделирования с использованием программы Electronics Workbench.

Теоретическая часть

Асинхронный и синхронный Т-триггеры

Т-триггер имеет один информационный вход. Логика работы асинхронного Т-триггера может быть описана таблицей переходов, которая имеет вид таблицы 1.

Таблица 4

ход	Состо яния	
	Q(0)	(1)

По таблице 1 может быть получено следующее уравнение асинхронного Т-триггера:

$$Q_{t+1} = \overline{T}Q_t + T\overline{Q}_t$$

Как видно из таблицы 1 и уравнения триггера, при T=1 асинхронный Т-триггер меняет свое состояние на противоположное, а при T=0 состояние триггера не изменяется.

Так как Т-триггер суммирует (или подсчитывает) по модулю два количество единиц, поступающих на его информационный вход, то Т-триггер называют также триггером со счетным входом.

Логика работы синхронного Т-триггера описывается таблицей переходов, которая имеет вид таблицы 2.

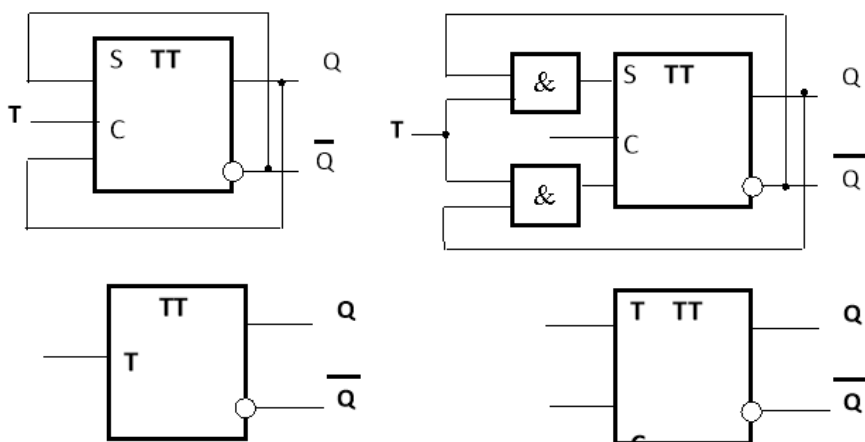
Таблица 2

Входы		Состояния	
C	T	Q(0)	Q(1)
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

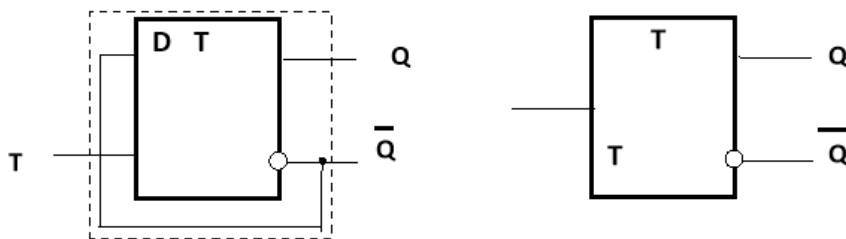
Из таблицы 2 видно, что при C=0 триггер не изменяет своего состояния, а при C=1 работает как асинхронный Т-триггер.

Функциональная схема Т-триггера может быть построена на основе синхронного RS-триггера (однотактного или двухтактного).

Схемы асинхронного и синхронного Т-триггера приведены на рисунках ниже. Обе схемы построены на основе синхронного двухтактного RS-триггера. Аналогичные схемы можно строить на основе однотактного RS-триггера. В двухтактных асинхронных Т-триггерах выходной сигнал формируется по заднему фронту входного сигнала Т, а в однотактных - по переднему фронту. В двухтактных синхронных Т-триггерах выходной сигнал формируется по заднему фронту сигнала С.

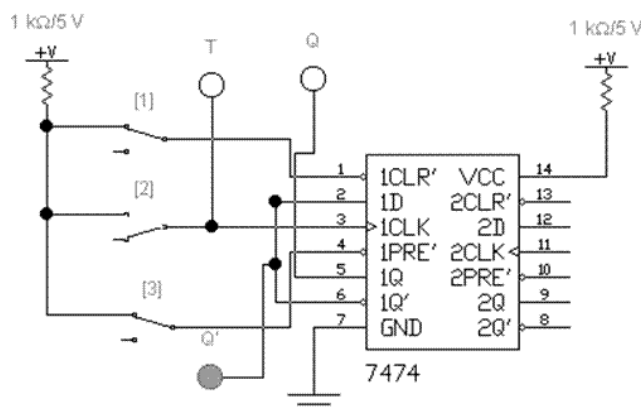


Схему асинхронного Т-триггера, в свою очередь, можно получить из D-триггера простой коммутацией входов и выходов:



Практическая часть.

Задание 1. Собрать схему и исследовать работу асинхронного Т-триггера, построенного на базе синхронного D-триггера в статическом режиме. Соответствующая схема показана на рисунке ниже. В качестве синхронного D-триггера использовать микросхему 7474 с дополнительными асинхронными входами установки и сброса (инверсные входы R и S). Результаты исследования занести в отчет.



Содержание отчета

В отчет о выполненной работе включить следующие материалы:

1. Тему и цель работы.
2. Результаты выполнения заданий: исследуемые схемы, полученные таблицы переходов.
3. Анализ полученных результатов.
4. Выводы по работе.

Контрольные вопросы

1. Как построить схему Т-триггера, если использовать схему RS - триггера и логические элементы?
2. Почему Т-триггер называется триггером со счетным входом?

Практическая работа №24. Моделирование и исследование логики работы JK-триггеров.

Цель работы: ознакомление с принципом работы JK-триггеров, получение практических навыков в построении и контроле их работоспособности, а также исследование логики работы JK-триггеров методом моделирования с использованием программы Electronics Workbench.

Теоретическая часть

JK-триггер называется также универсальным триггером. Универсальность схемы JK-триггера состоит в том, что простой коммутацией входов и выходов можно получать схемы других типов триггеров.

JK-триггер имеет два информационных входа. Вход J используется для установки триггера в состояние 1, а вход K -для установки в состояние 0, т.е. входы J и K аналогичны входам R и S RS-триггера. Отличие заключается в том, что на входы J и K могут одновременно поступать сигналы 1. В этом случае JK- триггер изменяет свое состояние на противоположное.

Таблица переходов JK-триггера при C=1 имеет вид таблицы 1.

Таблица 1

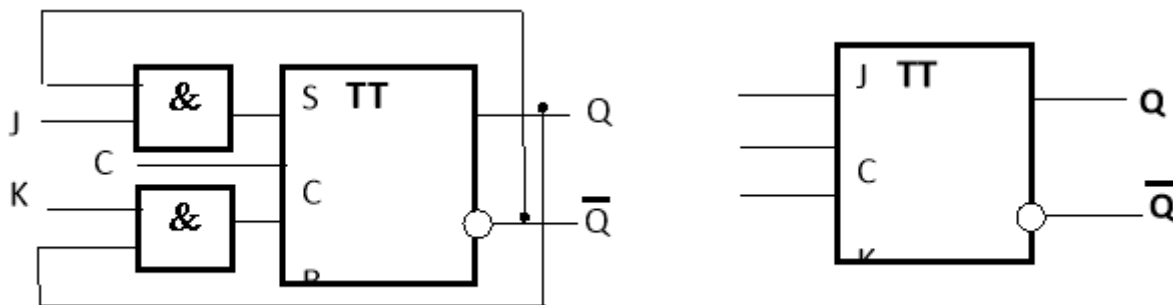
Входы		Состояния	
J	K	Q(0)	Q(1)
0	0	0	1
0	1	0	0
1	0	1	1
1	1	1	0

Из таблицы 1 можно получить следующее уравнение JK-триггера:

$$Q_{t+1} = J \bar{Q}_t + K Q_t$$

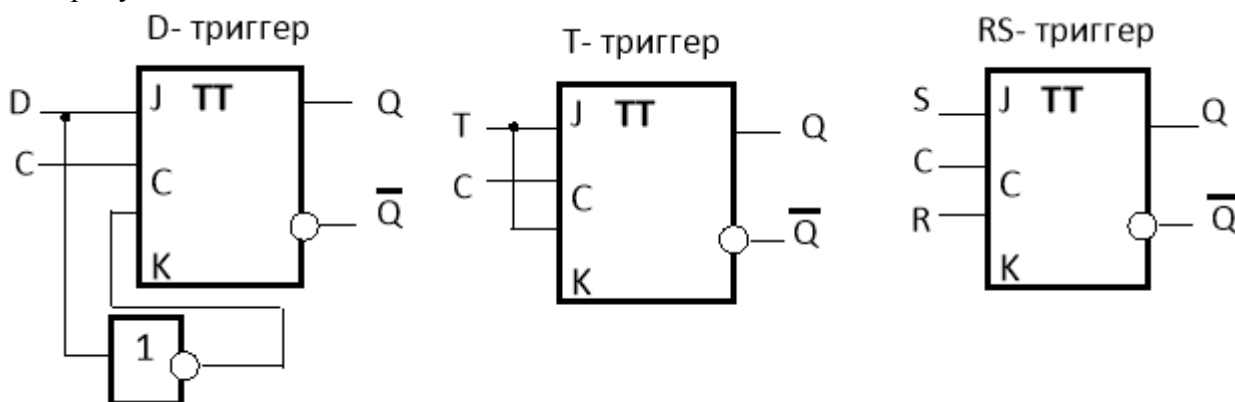
Следовательно, при J=1, K=0 всегда $Q_{t+1}=1$, а при J=0, K=1 всегда $Q_{t+1}=0$, т.е. JK-триггер работает как RS-триггер, если рассматривать входы J и K как входы S и R.

Функциональная схема двухтактного JK-триггера и УГО триггера показаны на рисунке ниже.



В свою очередь, при J=1, K=1 $Q_{t+1}=Q_t$, т.е. триггер переходит в противоположное состояние (работает как Т-триггер).

Примеры получения других типов триггеров на основе JK-триггера представлены на следующем рисунке:

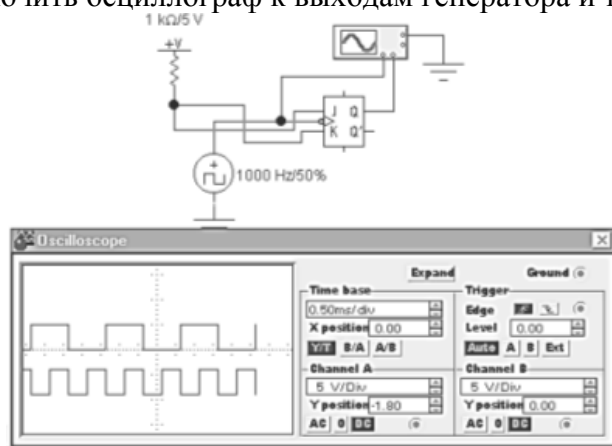


JK-триггер, кроме основных информационных входов и входа синхронизации, может иметь также дополнительные информационные входы, например, дополнительные инверсные асинхронные входы R и S, которые используются для установки триггера в 0 или 1 независимо от значения сигнала на входе синхронизации. Кроме того, триггер может иметь несколько входов J или K, объединенных по схеме И.

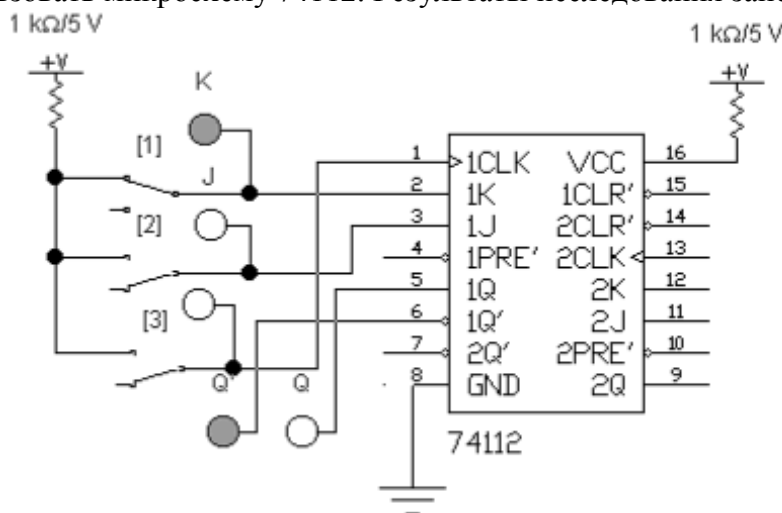
Практическая часть

Задание 1. Исследовать работу синхронного JK-триггера в динамическом режиме. Для этого собрать схему, показанную на рисунке ниже. При подаче на входы J и K сигналов высокого уровня, а на вход синхронизации импульсов от генератора, триггер будет работать в режиме

переключения с частотой в два раза ниже, чем частота генератора. Для визуальной индикации подключить осциллограф к выходам генератора и триггера.



Задание 2. Собрать схему и исследовать работу синхронного JK-триггера в статическом режиме. Соответствующая схема показана на рисунке ниже. В качестве синхронного JK-триггера использовать микросхему 74112. Результаты исследования занести в отчет.



Содержание отчета

В отчет о выполненной работе включить следующие материалы:

1. Тему и цель работы.
2. Результаты выполнения заданий: исследуемые схемы, полученные таблицы переходов.
3. Анализ полученных результатов.
4. Выводы по работе.

Контрольные вопросы

1. В каких случаях таблица переходов JK-триггера совпадает с таблицей переходов RS-триггера, в каких случаях отличается?
2. Почему JK-триггер называется универсальным триггером?
3. Чем отличается двухтактный триггер от однотактного триггера?

Практическая работа №25. Реализация функций, выполняемых статическими регистрами.

Цель работы: Исследование статического и динамического режимов работы регистров. Изучение использования регистров для реализации различных процедур.

Основные теоретические положения

Статический регистр - это операционный элемент ЭВМ, предназначенный для хранения двоичной информации в виде машинных слов и выполнения логических операций над словами, а также для выработки осведомительных сигналов относительно характера содержащейся в нем информации. Регистр представляет собой совокупность элементов памяти (как правило, триггеров), число которых определяется разрядностью машинных слов, и комбинационных схем, обеспечивающих выполнение указанных функций. Особенностью статического регистра является регулярность его структуры: каждый элемент памяти (ЭП), выполняющий функцию хранения одного разряда слова, дополняется комбинационной схемой, обеспечивающей функции вывода информации, а также выполнения логических операций над словами. В результате синтез статического регистра, реализующего определенные функции над словами, сводится к синтезу необходимой схемы для одного разряда регистра и повторению полученных схем для отдельных его разрядов.

Предметом изучения в настоящей работе являются три группы функций статического регистра:

- установочные функции;
- логические операции;
- формирование осведомительных сигналов.

Сделаем два замечания относительно функций первых двух групп, поскольку их реализация сопровождается переключением триггеров регистра. Во-первых, будем считать, что на выполнение каждой функции отводится такт машинного времени, а образование значений функции для всех разрядов регистра синхронизировано во времени путем подачи специального синхросигнала на входы С всех триггеров регистра. Во-вторых, реализация каждой функции происходит под действием определенного управляющего сигнала. Таким образом, с регистром, выполняющим несколько функций, связывается множество управляющих сигналов, на которые накладывается одно ограничение: в каждый момент времени активный (в нашем случае - высокий) уровень может иметь лишь один управляющий сигнал множества.

Формирование осведомительного сигнала реализуется комбинационной схемой, на входы которой подаются сигналы с выходов триггеров регистра. Поэтому значение осведомительного сигнала в момент времени t определяется состоянием триггеров регистра в тот же момент времени и, следовательно, отдельного машинного такта на формирование осведомительного сигнала не требуется.

Рассмотрим функции, выполняемые статическим регистром.

Установочные функции

Функция сброса. Выполнение этой функции сводится к обнулению (установке в 0) всех разрядов регистра. Соответствующая микрооперация может быть записана следующим образом:

$$Y_{\text{сбр}} : RG [0 : n - 1] := 0, \\ Q_0^{t+1}, \dots, Q_{n-1}^{t+1} = 0.$$

Функция ввода в регистр прямого кода числа В. В результате выполнения этой функции триггеры регистра устанавливаются в состояния, соответствующие значениям разрядов вводимого слова:

$$Y_{\text{вв}}^{\text{пр}} : RG [0 : n - 1] := B, \\ \text{где } B = b_{n-1}b_{n-2} \dots b_1b_0.$$

Функция ввода в регистр обратного кода числа. После выполнения этой функции триггеры регистра устанавливаются в состояния, противоположные значениям разрядов вводимого слова:

$$Y_{\text{вв}}^{\text{обр}} : RG [0 : n - 1] := \bar{B}.$$

Группа логических операций

Отметим особенности реализации логических операций в статическом регистре.

1. Выполнение любой логической операции над словами носит поразрядный характер, т.е. результатом операции является слово, значение каждого разряда которого есть результат этой операции над значениями одноименных разрядов слов, участвующих в операции.

Пусть $A * B = C$, где $A = a_{n-1}, \dots, a_0$; $B = b_{n-1}, \dots, b_0$; $C = c_{n-1}, \dots, c_0$; * - знак логической операции.

Разряды слова С определяются так:

$$c_0 = a_0 * b_1; \dots; c_{n-1} = a_{n-1} * b_{n-1}.$$

Пример. Определить $C = A \& B$;

$$A = 10110110,$$

$$B = 01111101.$$

$$\begin{array}{r} 10110110 \\ 01111101 \\ \hline 00110100 \end{array}$$

2. Будем считать, что к моменту выполнения логической операции слово, являющееся одним из операндов, находится в регистре, а слово - второй операнд - вводится в регистр извне; результат логической операции образуется в регистре. Таким образом, микрооперация выполнения логической операции может быть записана следующим образом:

$$Y * : RG [0 : n - 1] := RG [0 : n - 1] * B,$$

$$Q_0^{t+1} = Q_0^t * b_0^t; \dots; Q_{n-1}^{t+1} = Q_{n-1}^t * b_{n-1}^t,$$

где Y^* - управляющий сигнал, инициирующий выполнение логической операции, обозначенной знаком (*).

Синтез схем статического регистра. Рассмотрим синтез схем регистра для выполнения функций указанных двух групп. Для решения задачи достаточно ограничиться построением схемы одного разряда регистра. В качестве исходной информации будем использовать множество функций, которое должен реализовать регистр, тип триггера и таблицу входов триггера указанного типа.

Пусть множество функций включает в себя:

- функцию сброса;
- функцию ввода обратного кода числа;
- логическую операцию: конъюнкцию.

Рассмотрим триггер JK-типа (табл.1). Цель синтеза - построить комбинационную схему для одного разряда статического регистра. Входными сигналами схемы являются:

$Y_{сбр}$ - управляющий сигнал, инициирующий функцию сброса;

$Y_{вв\text{обр}}$ - управляющий сигнал, инициирующий функцию ввода обратного кода числа;

$Y_{\&}$ - управляющий сигнал, инициирующий логическую операцию - конъюнкцию;

b_i^t - сигнал i-го разряда вводимого слова;

Q_i^t - состояние триггера i-го разряда к моменту реализации функции.

Q_i^{t+1} - состояние триггера i-го разряда после прохождения синхроимпульса.

Таблица 1

Таблица функционирования JK-триггера

Q^t	Q^{t+1}	J^t	K^t
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0

Схема должна вырабатывать функции возбуждения J_i^t и K_i^t , однозначно определяющие состояние триггера i-го разряда в момент времени $t+1$, которое является результатом выполнения функций для i-го разряда.

Синтез схемы содержит следующие этапы:

- определение таблицы истинности для функций J_i и K_i ;
- построение на их основе диаграмм Вейча и минимизацию функций;
- составление минимальных аналитических выражений для функции возбуждения;
- построение по полученным выражениям принципиальных схем в заданном базисе.

Таблица истинности для функций J_i и K_i содержит в общем случае 32 строки, поскольку они являются функциями от пяти аргументов: $Y_{сбр}$, $Y_{вв}^{обр}$, $Y\&$, b_i^t , Q_i^t . С учетом ограничения на управляющие сигналы (только один сигнал имеет высокий уровень и, таким образом, выполняется только одна операция) сокращается число всевозможных наборов, а, следовательно, и число строк в таблице истинности до 16 (табл.2).

Таблица 2

Функционирование разряда статического регистра

$Y_{сбр}$	$Y_{вв}^{обр}$	$Y\&$	b_i^t	Q_i^t	Q_i^{t+1}	K_i	J_i	Комментарий
0	0	0	0	0	0	-	0	Регистр выполняет операцию хранения Нет управляющих сигналов
0	0	0	0	1	1	0	-	
0	0	0	1	0	0	-	0	
0	0	0	1	1	1	0	-	
1	0	0	0	0	0	-	0	Сброс $Y_{сбр}=1$ $Q_i^{t+1}=0$ При любом значении b_i^t
1	0	0	0	1	0	1	-	
1	0	0	1	0	0	-	0	
1	0	0	1	1	0	1	-	
0	1	0	0	0	1	-	1	Ввод обратного _ кода числа $Q_i^{t+1}=b_i^t$
0	1	0	0	1	1	0	-	
0	1	0	1	0	0	-	0	
0	1	0	1	1	0	1	-	
0	0	1	0	0	0	-	0	Выполняется конъюнкция $b_i^t \& Q_i^t$
0	0	1	0	1	0	1	-	
0	0	1	1	0	0	-	0	
0	0	1	1	1	1	0	-	

Для каждой функции J_i (K_i) строятся диаграммы Вейча (рис.1).

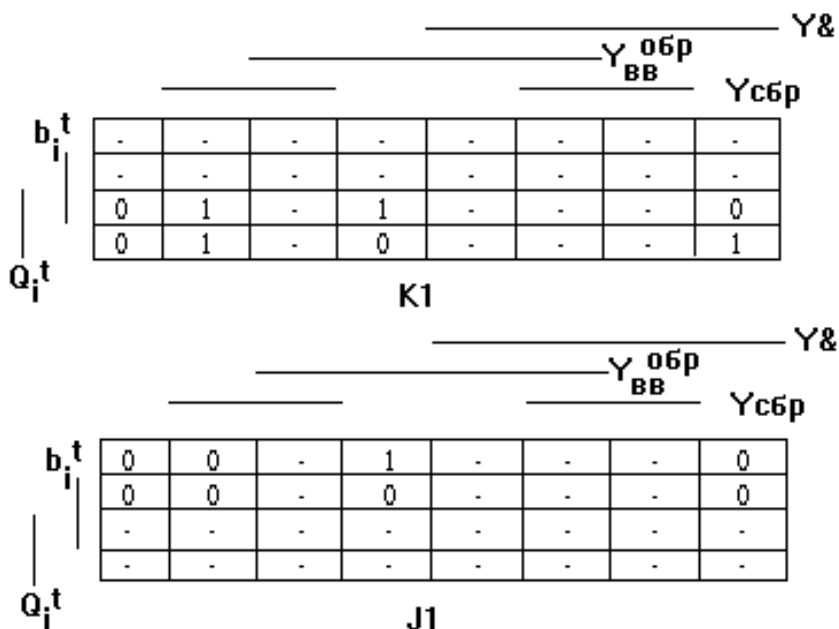


Рис.1

При этом из таблицы можно заполнить лишь 16 клеток, а остальные заполняются прочерками.

По полученным выражениям строим схемы, вырабатывающие сигналы для функций возбуждения. При этом необходимо учитывать особенности базиса, в котором строятся схемы. В данном примере в случае использования микросхем серии K155 можно схему для J_i реализовать, используя встроенную в триггер схему И. Логические выражения, соответствующие функциям K_i и J_i , имеют вид:

$$\begin{aligned}
 K_i &= Y_{сбр} \vee Y_{ВВ}^{обр} \cdot \bar{b}_i^t \vee Y_{\&} \cdot b_i^t; \\
 J_i &= Y_{ВВ}^{обр} \cdot \bar{b}_i^t.
 \end{aligned}
 \tag{1}$$

Синтезированная схема одного разряда статического регистра в соответствии с выражением (1), выполняющая три заданные функции, показана на рис.2.

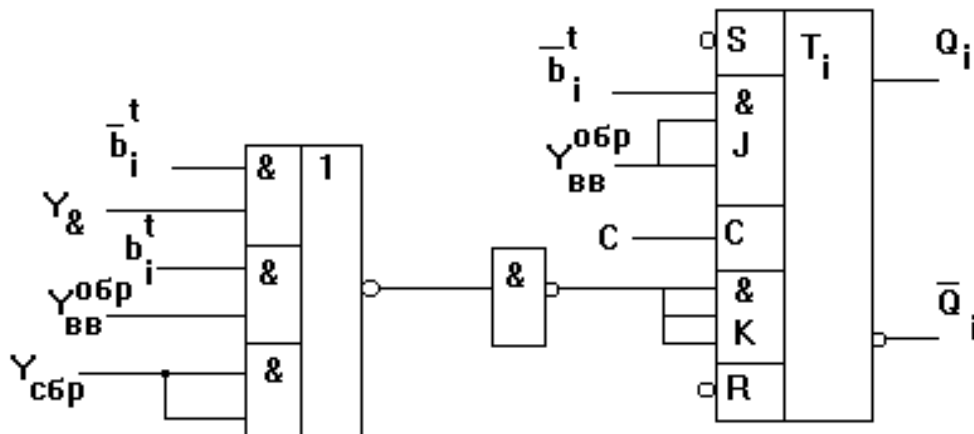


Рис.2

Порядок выполнения работы

1. В соответствии с заданием синтезировать схему статического регистра.
2. Разработать схему, формирующую осведомительный сигнал.

Содержание отчета

1. Формулировка задания.
2. Схемы исследования по п.1, 2, 3; таблицы функционирования, по которым строятся комбинационные схемы.

Варианты заданий

Задание 1. Синтезировать схему статического регистра.

Характеристики синтезируемого регистра:

I. Обнуление: 1) есть, 2) нет

II. Код ввода: 1) монофазный прямой, 2) монофазный обратный.

III. Код вывода: 1) парафазный прямой, 2) парафазный обратный,

IV. Выполняемая логическая операция:

1. Конъюнкция: $b \& Q$.

2. Дизъюнкция: $b \vee Q$.

3. Исключающее ИЛИ: $b \oplus Q$.

4. Функция Шеффера: $b \cdot Q$.

5. Функция Пирса: $b \vee Q$.

6. Равнозначность: $b \oplus Q$.

7. Запрет по Q: $b \bar{Q}$.

8. Запрет по b: $\bar{b} Q$.

9. Импликация: $b \vee \bar{Q}$.

10. Импликация: $\bar{b} \vee Q$.

Таблица 5

Номер вари- анта	I	II	III	IV
1	1	1	-	1
2	2	2	1	2
3	1	1	-	3
4	2	2	2	4
5	1	1	-	5
6	2	2	1	6
7	1	1	-	7

Цель работы: Изучение регистров памяти, сдвига, параллельно-последовательных регистров. Изучение способов ввода и вывода информации из регистров. Изучение особенностей организации регистров на триггерах различного типа.

Теоретическая часть

Регистр (англ. register) – устройство, предназначенное для хранения и преобразования многоразрядных двоичных чисел. В качестве запоминающего элемента в регистрах используют триггер.

Регистры классифицируют по различным признакам, основными из которых являются способы ввода информации в регистр и способы вывода информации из регистра.

Регистры делятся на три группы:

- параллельные регистры (иначе регистры памяти);
- регистры сдвига;
- параллельно-последовательные регистры.

По виду вводимой и выводимой информации различают регистры однофазного и парафазного типа.

В однофазных регистрах информация вводится (выводится) только в прямом или только в обратном коде.

В парафазных регистрах информация вводится (выводится) одновременно в прямом и обратном кодах.

Вывод информации из регистров осуществляется в прямом и обратном кодах (триггер имеет два выхода: Q – прямой и $\bar{Q}$ – инверсный).

Регистры памяти

Назначение регистров памяти – хранить двоичную информацию небольшого объема в течение некоторого промежутка времени. Эти регистры представляют собой набор триггеров, каждый из которых хранит один разряд двоичного числа.

В параллельных регистрах ввод (запись) и вывод (считывание) информации производится одновременно во всех разрядах параллельным кодом.

В случае если триггер имеет установочные входы и запись данных в регистр осуществляется по установочным входам, ввод обеспечивается подачей информационных сигналов и управляющего сигнала на установочные входы в прямом или обратном кодах (R , $\bar{R}$, S , $\bar{S}$).

Если триггер синхронный и не имеет установочных входов, то он представляет собой, по существу, наборы триггеров с независимыми информационными входами и обычно общим тактовым входом. В таких регистрах информация подается на информационные входы (например, вход D) и запись осуществляется подачей тактового командного импульса. С приходом очередного тактового импульса происходит обновление записанной информации.

Способы ввода информации в регистр, организованный на триггерах с установочными входами

Существует несколько способов записи информации по установочным входам:

- в прямом коде по установочному входу S с предварительной установкой регистра в «0»;
- в обратном коде по установочному входу R с предварительной установкой в «1» всех триггеров;
- с подачей парафазного кода информации (прямой код на вход S , обратный код на вход R).

Во всех перечисленных способах запись информации осуществляется уровнем сигнала.

Схема регистра с возможностью записи информации по установочному входу S с предварительной установкой регистра в «0» представлена на рисунке 1.а, диаграмма работы данного регистра – на рисунке 1.б.

Как видно из диаграммы, до момента действия сигнала $U_{ст}$ «0» (момент времени t_1), регистр находится в некотором состоянии ($Q_1 = 1$, $Q_2 = 1$, ..., $Q_m = 1$), т.е. хранит некоторую, ранее записанную информацию.

Для корректной работы схемы первым (в момент времени t_1) подается сигнал $U_{ст}$. «0» (высокий уровень), который устанавливает все разряды регистра в нулевое состояние.

На входы регистра подается в прямом коде некоторая информация ($D1 = 0, D2 = 1, \dots, Dm = 1$), которая поступает на один из входов логических элементов, образующих управляющую комбинационную схему. В момент прихода управляющего сигнала Тзап (момент времени $t2$), информация, присутствующая на входах $D1, D2, \dots, Dm$, записывается в регистр. Т.е. триггер, на установочный вход S которого подается сигнал логической «1» с ЛЭ И (вход $Di = 1, Тзап = 1$), переходит в состояние «1». Триггер, на установочный вход S которого подается сигнал логического «0» с ЛЭ И (вход $Di = 0, Тзап = 1$), остается в состоянии логического «0», обусловленном сигналом Уст «0».

Регистр хранит записанную информацию до момента времени $t3$. К этому моменту времени на входах $D1, D2, \dots, Dm$ установились новые данные. В момент $t3$ подается сигнал Уст «0». В момент $t4$ подается сигнал Тзап, и в регистр записываются новые данные.

Недостатком приведенной схемы является необходимость подачи сигнала Уст «0».

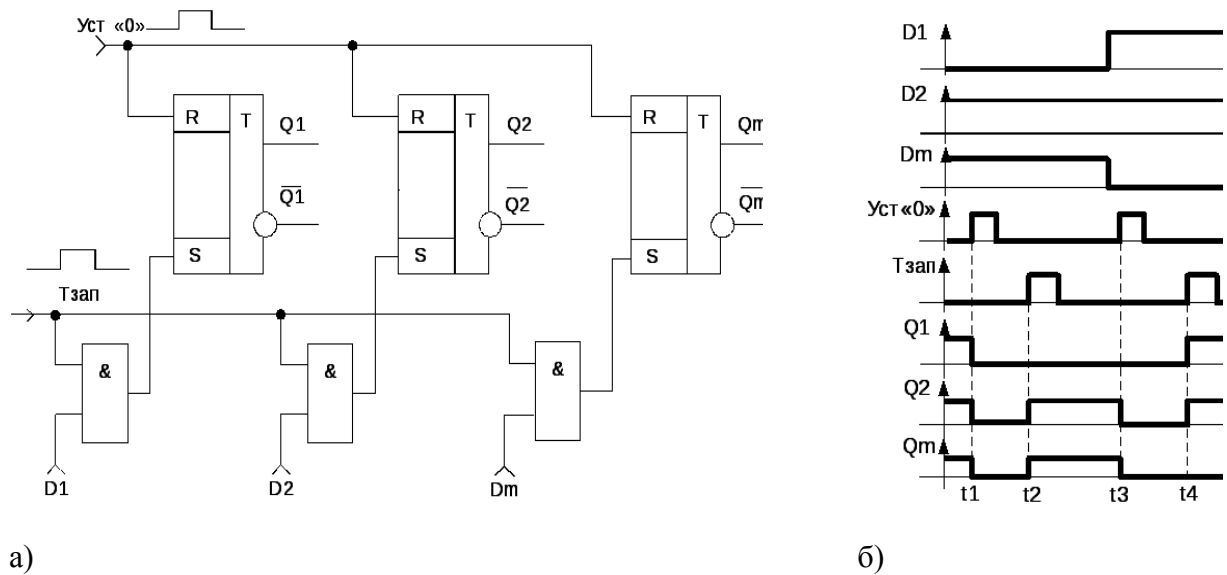


Рис. 1. – Запись информации в прямом коде по установочному входу S с предварительной установкой в «0» всех триггеров регистра

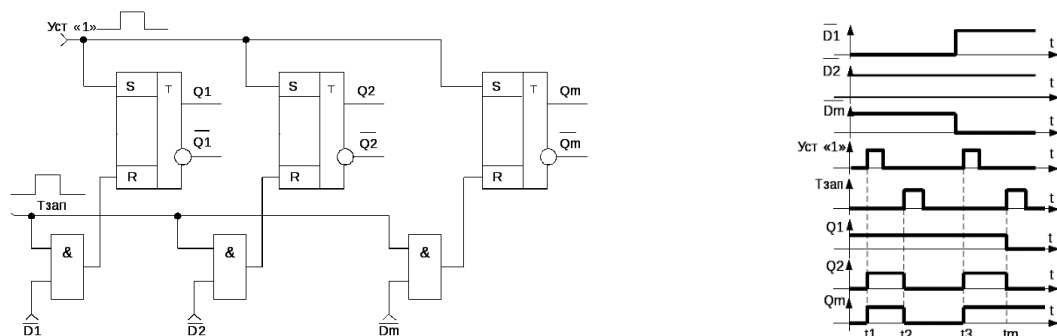
Если триггер имеет инверсные установочные входы ($\bar{R}\bar{S}$), то для записи информации, подаваемой на входы Di в прямом коде, необходимо вместо ЛЭ И использовать ЛЭ И-НЕ, а в цепь Уст «0» включить инвертор.

Схема регистра с возможностью записи информации по установочному входу R с предварительной установкой в «1» всех триггеров регистра приведена на рисунке 2.а, диаграмма его работы – на рисунке 2.б.

Особенностью работы данного регистра является то, что данные, поступающие на входы Di , должны подаваться в обратном коде.

При записи числа в регистр в состояние логического «0» устанавливаются только те триггеры, на которые подается обратный код разряда числа равный «1». Остальные триггеры останутся в состоянии «1».

Недостатком приведенной схемы является необходимость подачи сигнала Уст «1» на все триггеры.

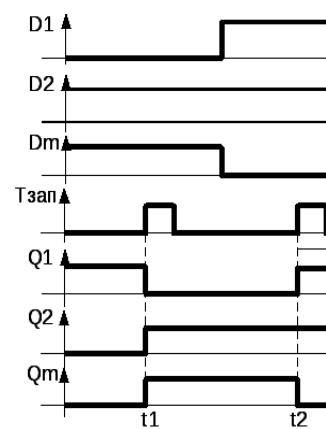
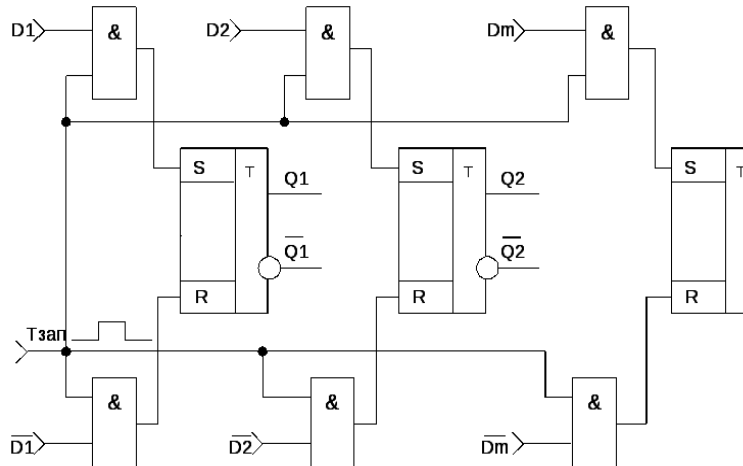


а)

б)

Рис. 2 – Запись информации по установочному входу R с предварительной установкой в «1» всех триггеров регистра

Схема регистра с возможностью записи информации парафазным кодом представлена на рисунке 3.а, диаграмма его работы – на рисунке 3.б.



а)

б)

Рис. 3 – Запись информации по установочным входам парафазным кодом

При записи парафазным кодом прямой код числа подается на вход S, обратный код – на вход R.

Достоинством данной схемы является отсутствие сигнала предварительной установки регистра.

Ввод информации с использованием информационного входа и входа синхронизации триггера

Если регистр организован на D-триггерах, то по положительному фронту тактового сигнала Tзап, поступающему на входы синхронизации C, рисунок 4, информация, присутствующая на входах Di, записывается в регистр.

Информация в регистре сохраняется до момента выключения питания схемы или записи новой информации.

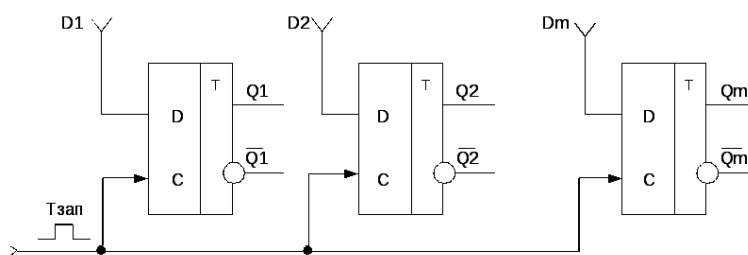


Рис. 4 – Запись информации параллельным кодом

На рисунке 5 представлена схема параллельного регистра на JK-триггерах.

В данном случае необходим парафазный код числа. Прямой код $D_1, D_2, \dots, D_m$ подается на входы J, обратный код – на входы K.

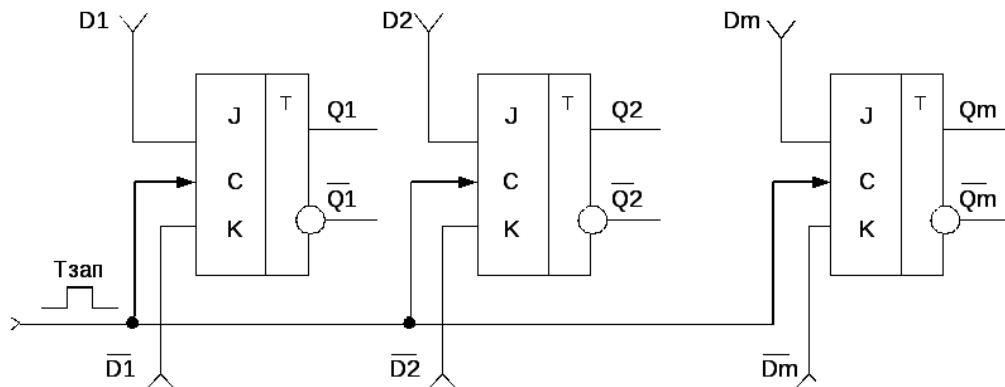


Рис. 5 – Запись информации парафазным кодом

Регистры сдвига

В регистрах сдвига все триггеры соединены в последовательную цепочку (выход каждого предыдущего триггера соединен со входом D следующего триггера). Входы синхронизации C всех триггеров объединены между собой.

По тактовому сигналу содержимое каждого предыдущего триггера переписывается в следующий по порядку в цепочке триггер. Код, хранящийся в регистре, с каждым тактом сдвигается на один разряд в сторону старших разрядов или в сторону младших разрядов в зависимости от организованных связей.

Для регистров сдвига указывается направление сдвига:

- вправо (основной режим, который есть у всех сдвиговых регистров);
- ← влево (этот режим есть только у некоторых, реверсивных сдвиговых регистров);
- ↔ реверсивный (двунаправленный), т. е. записанную информацию можно сдвигать по линейке триггеров вправо или влево. Для включения режима сдвига предусматривают специальный управляющий вход.

Направление сдвига отражает внутреннюю структуру регистров сдвига, рисунок 6. При этом триггеры нумеруются слева направо, например, от 0 до 7.

В результате сдвиг информации регистром вправо представляет собой сдвиг в сторону разрядов, имеющих большие номера (старших разрядов), т.е. на вход D число подается, начиная со старшего разряда.

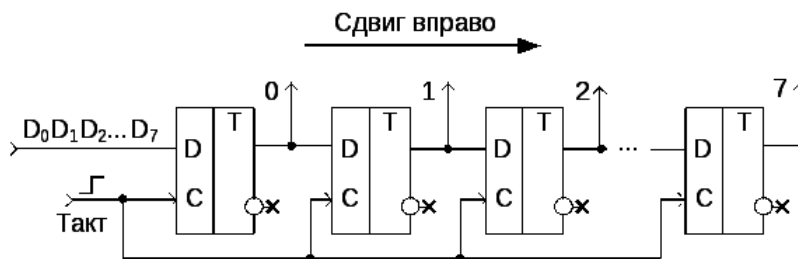


Рис. 6 – 8-ми разрядный регистр сдвига вправо

Сдвиг информации регистром влево – это сдвиг в сторону разрядов, имеющих меньшие номера (младших разрядов), т.е. на вход D число подается, начиная с младшего разряда, рисунок 7.

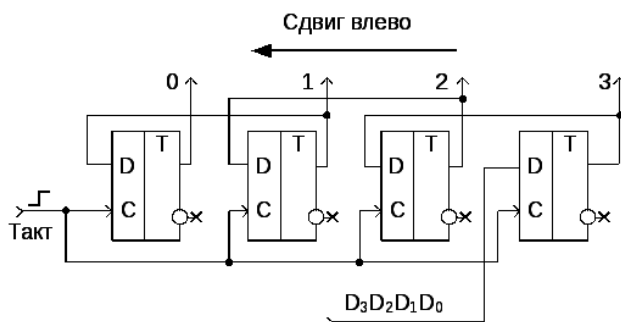


Рис. 7 – 4-х разрядный регистр сдвига влево

Однако, как известно, в любом двоичном числе слева расположены старшие разряды, а справа – младшие разряды. О такой особенности необходимо помнить разработчику цифрового прибора.

Все регистры сдвига строятся на базе двухступенчатых триггеров или синхронизируемых фронтом синхроимпульса.

Регистры сдвига могут быть построены и на триггерах одноступенчатой структуры. В этом случае в каждом разряде регистра нужно использовать два триггера, которые управляются двумя сдвинутыми во времени тактовыми импульсами. Если бы в регистре были применены одноступенчатые триггеры по одному на разряд, то правило работы регистра было бы нарушено: при первом же импульсе сдвига информация, записавшись в первый разряд, перешла бы во второй, затем в третий и т.д.

Реверсивные регистры сдвига объединяют в себе свойства регистров прямого и обратного сдвига.

Разряд реверсивного регистра представлен на рисунке 8.

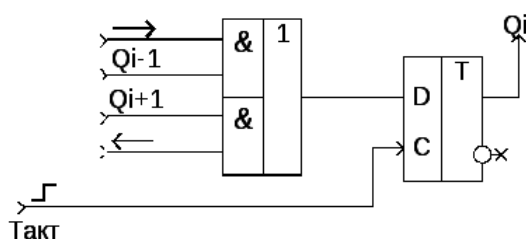


Рис. 8 – Разряд реверсивного регистра

Способы вывода информации из регистра

Т.к. регистры строятся на триггерах, а триггер имеет прямой Q и инверсный $\bar{Q}$ выходы, то существует несколько способов вывода информации из регистра.

1. Вывод прямым параллельным m -разрядным кодом на m -разрядную шину, рисунок 9. Для вывода информации подается сигнал $T_{\text{выв.пр.пар}}$ высокого уровня. Причем сигнал $T_{\text{выв.пр.пар}}$ должен быть подан после сигнала $T_{\text{зап}}$ (должно соблюдаться условие $T_{\text{зап}} * T_{\text{выв.пр.пар}} = 0$) и необходимо учитывать время задержки самого триггера, на котором реализован регистр.

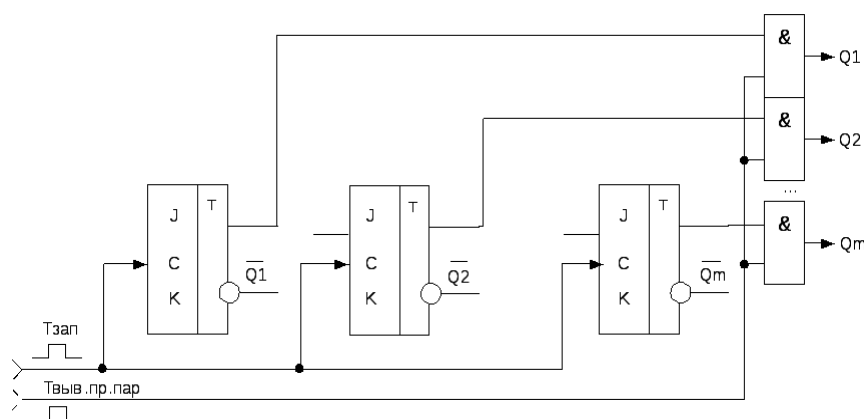


Рис. 9 – Вывод информации прямым m -разрядным кодом на m -разрядную шину

2. Вывод обратным параллельным m -разрядным кодом на m -разрядную шину, рисунок 10. Для вывода информации подается сигнал $T_{\text{выв.обр.пар}}$ высокого уровня. При подаче управляющих сигналов должно соблюдаться условие $T_{\text{зап}} * T_{\text{выв.обр.пар}} = 0$.

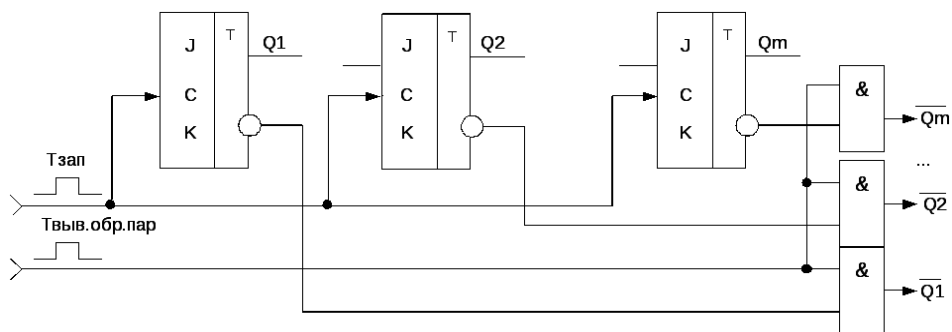


Рис. 10 – Вывод информации обратным m -разрядным кодом на m -разрядную шину

3. Прямым или обратным параллельным m -разрядным кодом на m -разрядную шину, рисунок 1.11. Сигналы Твыв.пр и Твыв.обр подаваться вместе не могут, т.е. должно соблюдаться условие $T_{\text{выв.пр}} * T_{\text{выв.обр}} = 0$. В зависимости от поданного в данный момент времени сигнала вывода выход Y_i соответствует прямому (Q_i) или обратному ($\bar{Q}_i$) кодам.

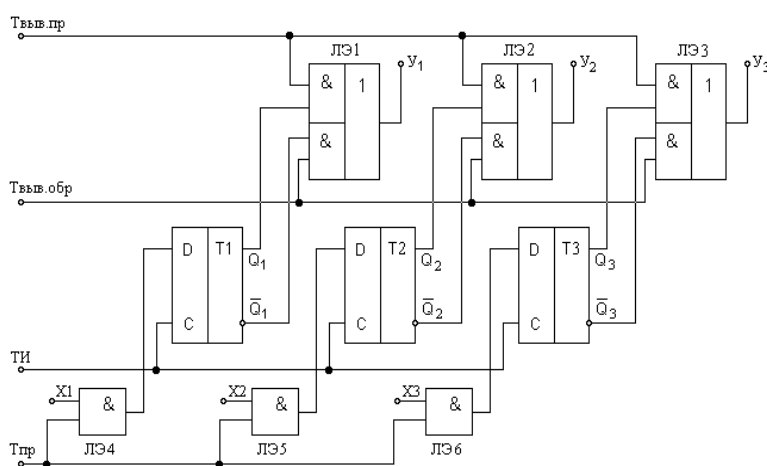


Рис. 1.11 – Вывод информации прямым или обратным m -разрядным кодом на m -разрядную шину

4. Парафазным m -разрядным кодом (прямой m -разрядный код на одну шину, обратный m -разрядный код – на другую шину). Из рисунка 12 видно, что при подаче управляющего сигнала Твыв.параф на одну m -разрядную шину выводится прямой код числа, а на вторую m -разрядную шину выводится обратный код числа. Для корректной работы схемы должно соблюдаться условие $T_{\text{зап}} * T_{\text{выв.параф}} = 0$.

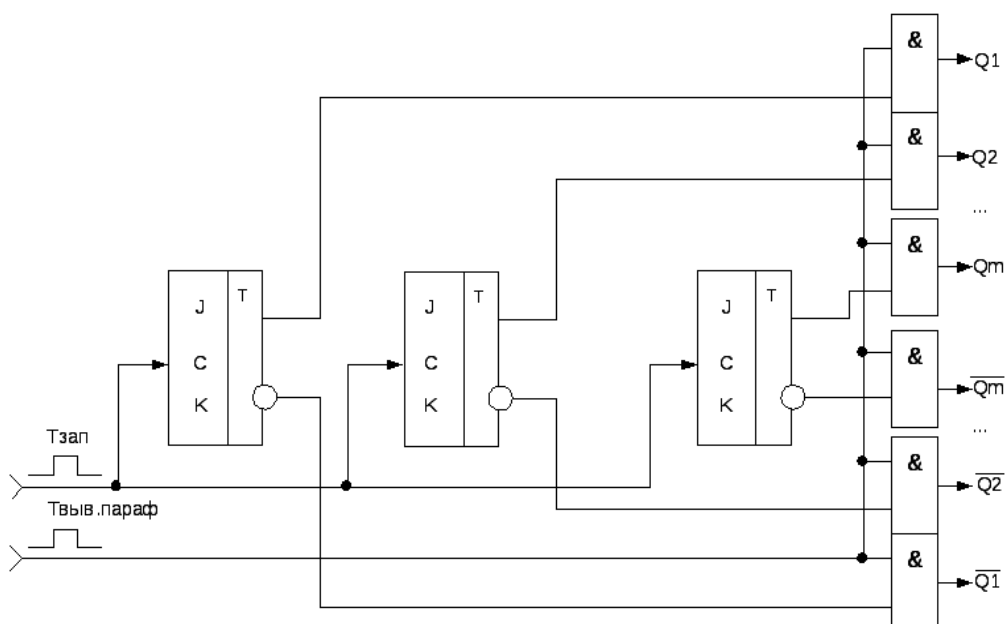


Рис. 12 – Вывод информации парафазным m-разрядным кодом

Все перечисленные способы действительны и для последовательного вывода информации из регистра.

Индивидуальные задания

Задание 1

В соответствии с вариантом, заданным преподавателем из таблицы 2.1, нарисовать принципиальную схему и диаграмму работы 4-х разрядного регистра.

Таблица 2.1 – Варианты для выполнения индивидуального задания 1

№ варианта	Триггер	Тип ввода	Тип вывода	Вводимые данные
В-01	JK-триггер	Параллельным кодом с предварительной установкой в 0 (высоким)	Со старшего разряда в прямом коде с обнулением регистра	0111
В-02	D-триггер	Парафазным кодом	В прямом или обратном кодах с младшего разряда	1100
В-03	JK-триггер	Парафазным кодом	С младшего разряда в прямом коде с обнулением регистра	1010
В-04	D-триггер	Параллельным кодом с предварительной установкой в 0 (низким)	Со старшего разряда в прямом коде	0101
В-05	JK-триггер	Параллельным кодом с предварительной установкой в 15 ₁₀ (низким)	Парафазным со старшего разряда с обнулением регистра	1100
В-06	D-триггер	В прямом коде с младшего разряда	В прямом или обратном коде на 4-х разрядную шину	1000
В-07	JK-триггер	В обратном коде с младшего разряда	В обратном коде на 4-х разрядную шину	0011
В-08	D-триггер	Параллельным кодом с предварительной установкой в 0 (низким)	В прямом или обратном коде со старшего разряда с обнулением регистра	1010
В-09	JK-триггер	Параллельным кодом с предварительной установкой в 0 (высоким)	Парафазным со старшего разряда	0011
В-10	D-триггер	Парафазным кодом	В обратном коде со старшего разряда с обнулением регистра	1101
В-11	JK-триггер	В обратном коде с младшего разряда	Парафазным	1101
В-12	D-триггер	В прямом коде со старшего разряда	В обратном или прямом коде на 4-х разрядную	0010

			шину	
B-13	JK-триггер	В прямом коде с младшего разряда	Парафазным	1000
B-14	D-триггер	Парафазным кодом	В обратном коде с младшего разряда	0001
B-15	JK-триггер	В обратном коде со старшего разряда	Парафазный на две шины	0101
B-16	D-триггер	В обратном коде со старшего разряда	В прямом коде на 4-х разрядную шину	1110
B-17	JK-триггер	В прямом или обратном кодах с младшего разряда	Парафазным	0100
B-18	D-триггер	В обратном коде с младшего разряда	В обратном коде на две 4-х разрядные шины	1010
B-19	JK-триггер	Параллельным кодом с предварительной установкой в 0 (низким)	В прямом или обратном со старшего разряда с обнулением	1011
B-20	D-триггер	Парафазным кодом	В прямом с младшего разряда с обнулением регистра	1100
B-21	JK-триггер	Параллельным кодом с предварительной установкой в 0 (низким)	Со старшего разряда с обнулением регистра	1011
B-22	D-триггер	В обратном коде с младшего разряда	Парафазным кодом	0101
B-23	JK-триггер	Парафазным кодом	В прямом коде с младшего разряда с обнулением регистра	1000
B-24	D-триггер	Параллельным кодом с предварительной установкой в 0 (низким)	В прямом или обратном коде на 4-х разрядную шину	1010

Практическая работа №27. Синтез регистра сдвига.

Цель работы: разработать регистр сдвига, в который последовательно записывается информация (5 бит). Для синтеза схемы применяются D-триггеры и логические элементы базиса И-ИЛИ.

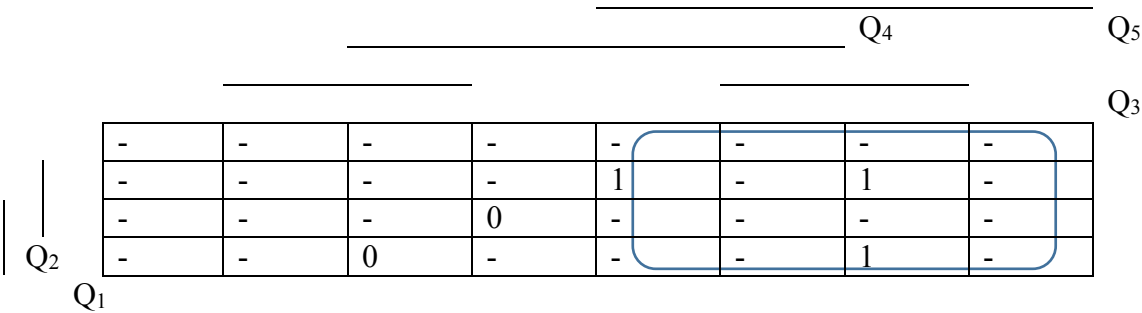
Теоретическая часть

Для данного регистра сдвига требуется 5 D-триггеров. Для синхронизации работы на синхронизирующие входы всех триггеров подается сигнал с генератора импульсов.

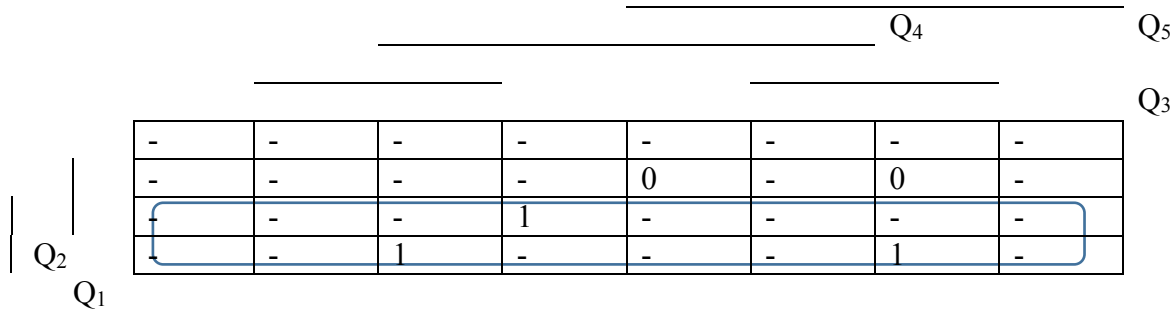
Пример прохождения числа 11010 через заданный кольцевой регистр сдвига можно продемонстрировать с помощью таблицы 1.

Таблица 1

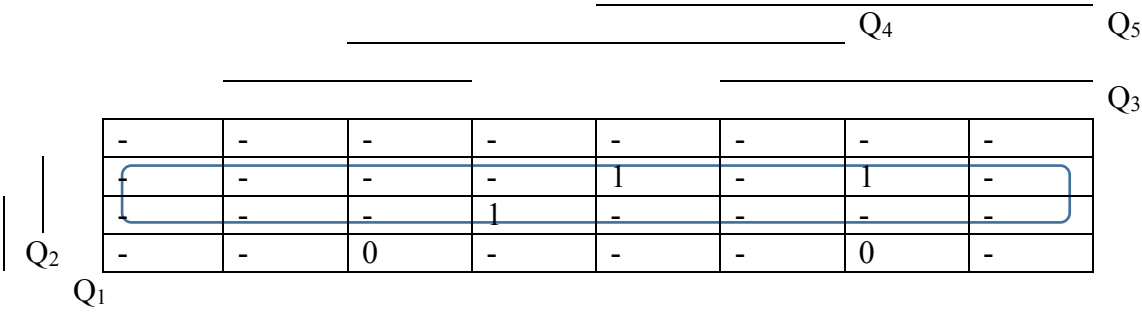
	Выходы					Состояния триггеров				
Такты	Q1	Q2	Q3	Q4	Q5	D1	D2	D3	D4	D5
1	1	1	0	1	0	0	1	1	0	1
2	0	1	1	0	1	1	0	1	1	0
3	1	0	1	1	0	0	1	0	1	1
4	0	1	0	1	1	1	0	1	0	1
5	1	0	1	0	1	1	1	0	1	0
6	1	1	0	1	0	0	1	1	0	1
7	0	1	1	0	1	1	0	1	1	0
8	1	0	1	1	0	0	1	0	1	1
9	0	1	0	1	1	1	0	1	0	1
10	1	0	1	0	1	1	1	0	1	0
11	1	1	0	1	0	0	1	1	0	1
12	0	1	1	0	1	1	0	1	1	0
13	1	0	1	1	0	0	1	0	1	1
14	0	1	0	1	1	1	0	1	0	1
15	1	0	1	0	1	1	1	0	1	0
16	1	1	0	1	0	0	1	1	0	1



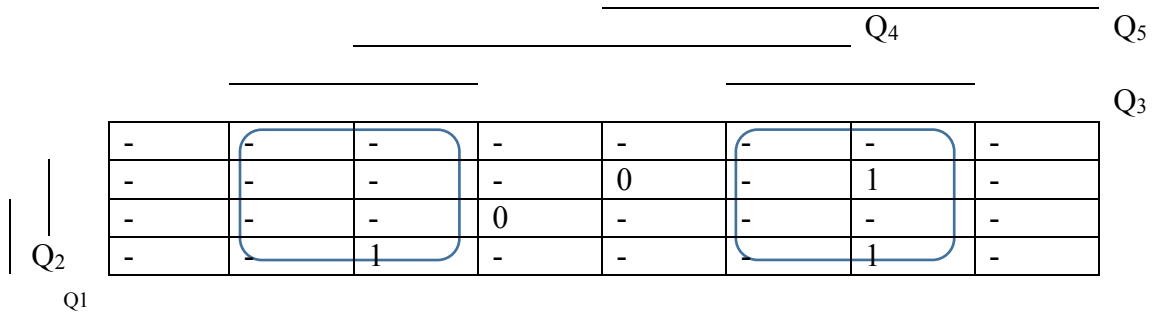
$D1 = Q5$



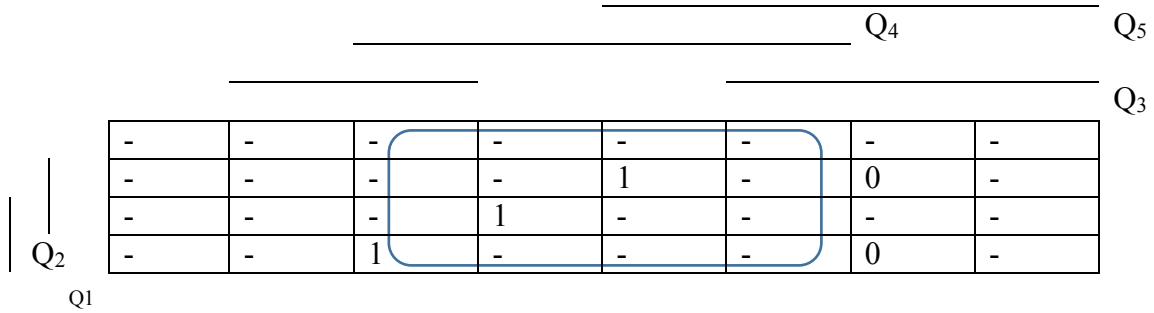
$D2 = Q1$



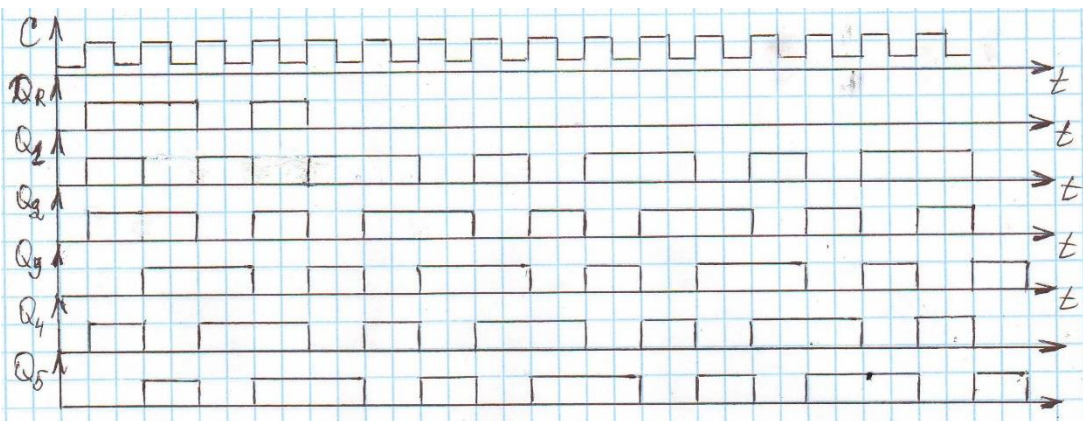
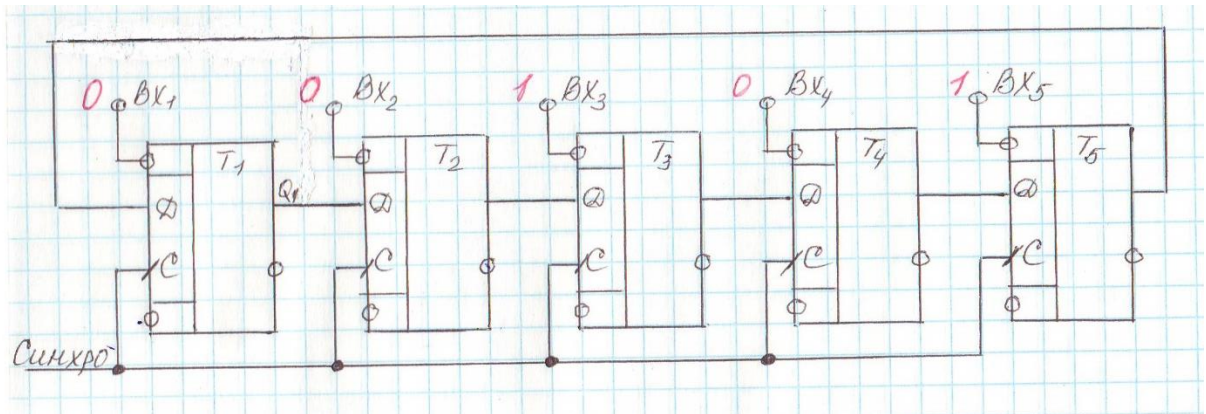
$D3 = Q2$



$$D4 = Q3$$



$$D5 = Q4$$



Задания на практическую работу

№ варианта	Сдвигаемое число
1.	10101
2.	11001
3.	01110
4.	11001
5.	10011
6.	10010
7.	01010
8.	01101

9.	10110
10.	01010
11.	01100
12.	10001
13.	11100
14.	00111
15.	01011
16.	00111
17.	11000
18.	10001
19.	00101
20.	10100

Практическая работа №28. Синтез суммирующего счетчика.

Цель работы: синтезировать схему суммирующего счетчика.

Пример выполнения практической работы

Модуль счёта равен двенадцати $K_{сч}=12$.

Характеристическая таблица JK триггера:

$Q_n \rightarrow Q_{n+1}$	J_n	K_n
$0 \rightarrow 0$	0	*
$0 \rightarrow 1$	1	*
$1 \rightarrow 0$	*	1
$1 \rightarrow 1$	*	0

Таблица переключений:

N	Q_4^n	Q_3^n	Q_2^n	Q_1^n	Q_4^{n+1}	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	J_4	K_4	J_3	K_3	J_2	K_2	J_1	K_1
0	0	0	0	0	0	0	0	1	0	*	0	*	0	*	1	*
1	0	0	0	1	0	0	1	0	0	*	0	*	1	*	*	1
2	0	0	1	0	0	0	1	1	0	*	0	*	*	0	1	*
3	0	0	1	1	0	1	0	0	0	*	1	*	*	1	*	1
4	0	1	0	0	0	1	0	1	0	*	*	0	0	*	1	*
5	0	1	0	1	0	1	1	0	0	*	*	0	1	*	*	1
6	0	1	1	0	0	1	1	1	0	*	*	0	*	0	1	*
7	0	1	1	1	1	0	0	0	1	*	*	1	*	1	*	1
8	1	0	0	0	1	0	0	1	*	0	0	*	0	*	1	*
9	1	0	0	1	1	0	1	0	*	0	0	*	1	*	*	1
10	1	0	1	0	1	0	1	1	*	0	0	*	*	0	1	*
11	1	0	1	1	1	1	0	0	*	0	1	*	*	1	*	1
12	1	1	0	0	0	0	0	0	*	1	*	1	0	*	0	*

Заполняем диаграммы Вейча для каждого из входов триггеров: J1, K1, J2, K2, J3, K3, J4, K4.
Шаблон диаграммы Вейча на 4 переменных:

	X2				
X1	1	1	0	0	X3
	1	1	0	1	
	1	0	0	1	
	1	0	0	0	
	X4				

J1		K1	
X	X	1	1
X	X	1	1
1	X	X	X
1	1	X	X

J2		K2	
X	X	1	1
X	X	1	X
X	X	0	X
X	X	0	X

J3		K3	
1	1	X	X
X	X	1	X
X	X	0	X
0	0	X	X

J4		K4	
0	0	X	X
1	X	X	X
0	X	X	1
0	0	X	X

J1		K1	
1	1	1	1
1	0	1	1
1	0	1	1
1	1	1	1

J2		K2	
1	1	1	1
1	1	1	1
0	0	0	0
0	0	0	0

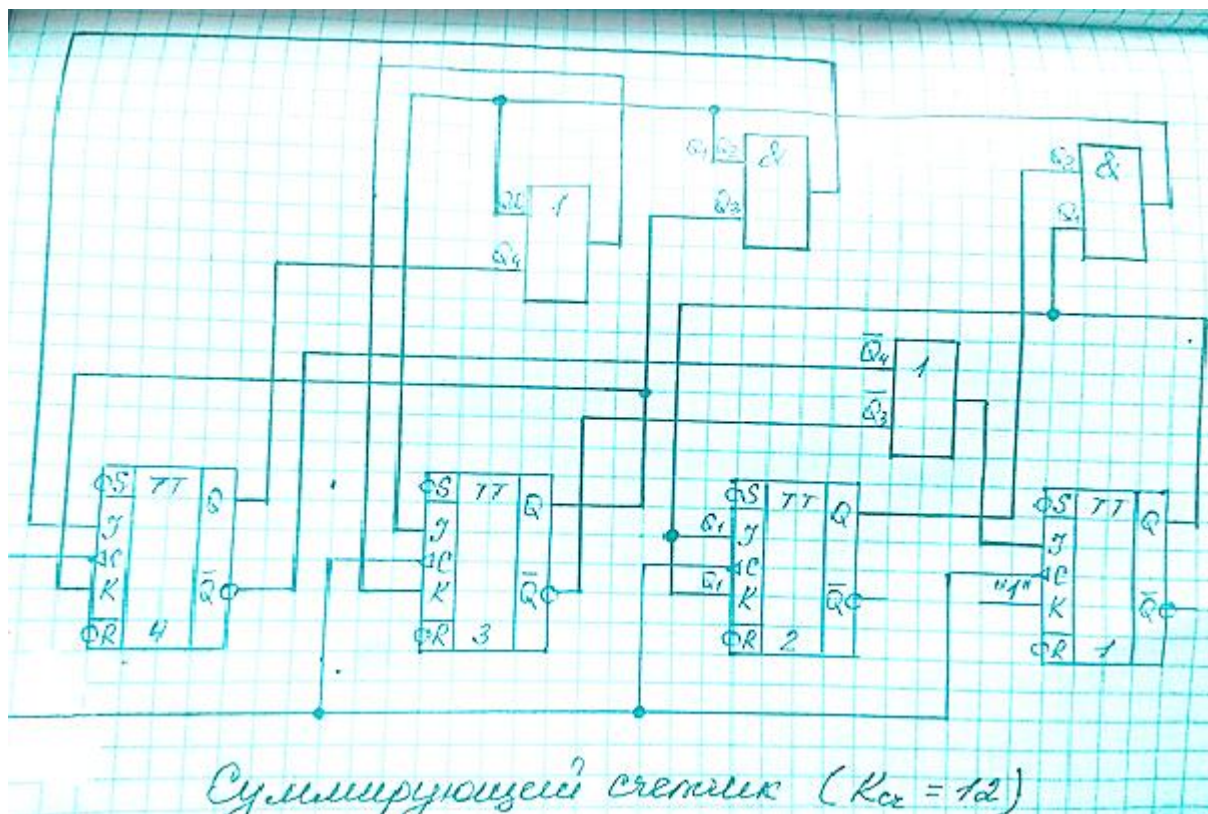
J3		K3	
1	1	1	1
1	1	1	1
0	0	0	1
0	0	0	1

J4		K4	
0	0	0	0
1	1	1	1
0	0	1	1
0	0	0	0

Производим минимизацию, получаем уравнения входов:

$$J_1 = \bar{Q}_4 \vee \bar{Q}_3; \quad K_1 = 1; \quad J_2 = Q_1; \quad K_2 = Q_1; \quad J_3 = Q_1 Q_2; \quad K_3 = Q_1 Q_2 \vee Q_4; \quad J_4 = Q_1 Q_2 Q_3; \quad K_4 = Q_3$$

Синтезируем схему суммирующего счетчика.



Задание на практическую работу

Последняя цифра студенческого билета	$K_{сч}$
0 или 4	9
1 или 5	10
2 или 6	11
3 или 7	13
4 или 8	14
5 или 9	15

Практическая работа №29. Синтез вычитающего счетчика.

Цель работы: синтезировать схему вычитающего счетчика.

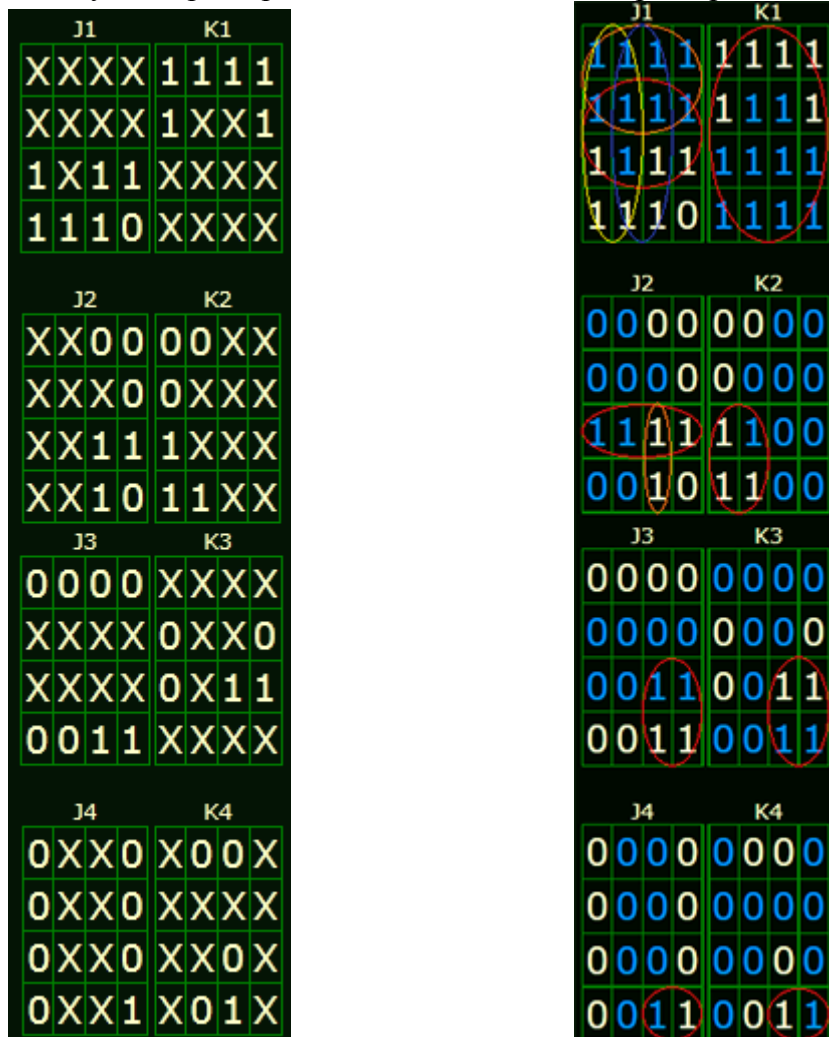
Пример выполнения практической работы

Как и в практической работе №28 составляем таблицу переключений (счет обратный от 12 (1100) до 0 (0000):

N	Q_4^n	Q_3^n	Q_2^n	Q_1^n	Q_4^{n+1}	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	J_4	K_4	J_3	K_3	J_2	K_2	J_1	K_1
0	1	1	0	0	1	0	1	1	*	0	*	1	1	*	*	0
1	1	0	1	1	1	0	1	0	*	0	0	*	*	0	*	1
2	1	0	1	0	1	0	0	1	*	0	0	*	*	1	*	0
3	1	0	0	1	1	0	0	0	*	0	0	*	0	*	*	1
4	1	0	0	0	0	1	1	1	*	1	1	*	1	*	1	*
5	0	1	1	1	0	1	1	0	0	*	*	0	*	0	0	*
6	0	1	1	0	0	1	0	1	0	*	*	0	*	1	1	*
7	0	1	0	1	0	1	0	0	0	*	*	0	0	*	0	*
8	0	1	0	0	0	0	1	1	0	*	*	1	1	*	1	*

9	0	0	1	1	0	0	1	0	0	*	0	*	*	0	0	*
10	0	0	1	0	0	0	0	1	0	*	0	*	*	1	1	*
11	0	0	0	1	0	0	0	0	0	*	0	*	0	*	0	*
12	0	0	0	0	1	1	0	0	1	*	1	*	0	*	*	1

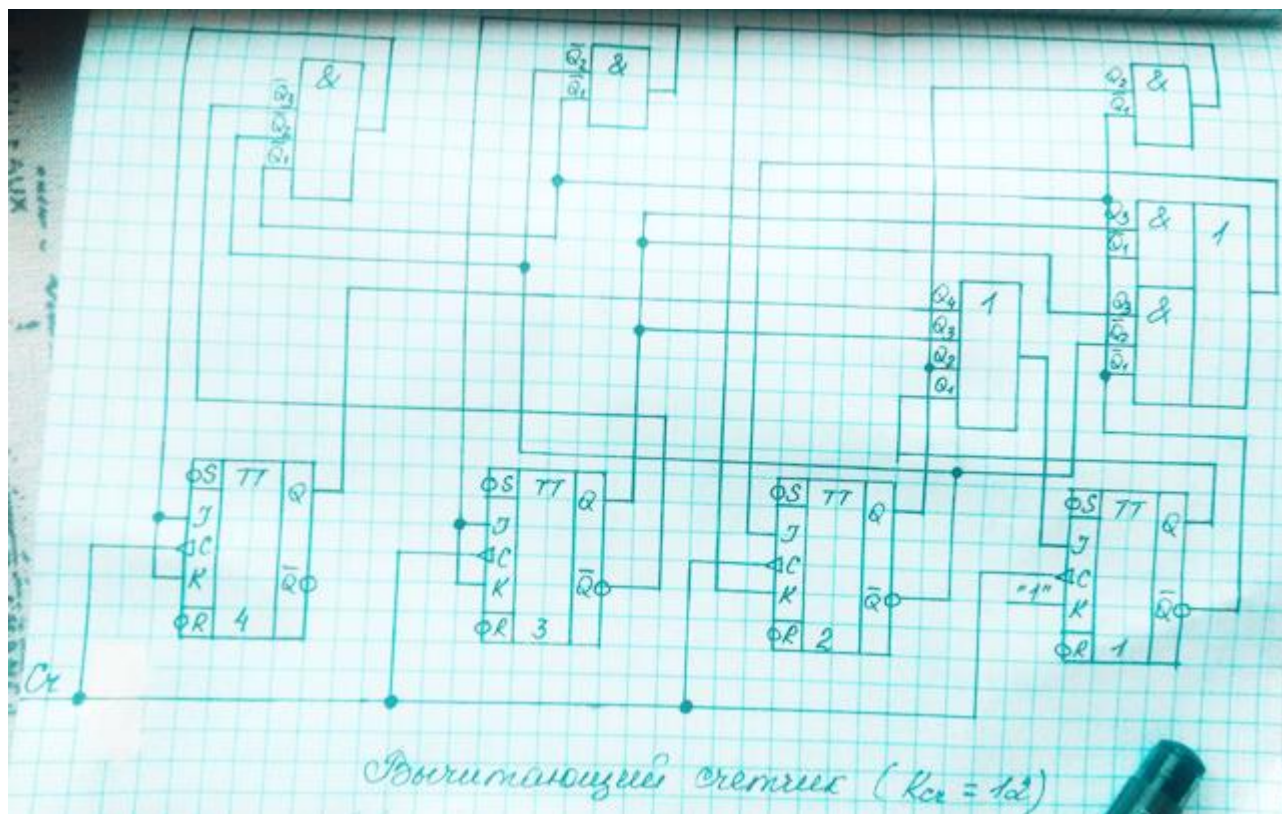
Пользуясь характеристической таблицей JK-триггера, заполняем диаграммы входов:



Производим минимизацию:

$$J_1 = Q_1 \vee Q_2 \vee Q_3 \vee Q_4; \quad K_1 = 1; \quad J_2 = \bar{Q}_1 Q_3 \vee \bar{Q}_1 \bar{Q}_2 Q_4; \quad K_2 = \bar{Q}_1 Q_2; \quad J_3 = K_3 = \bar{Q}_1 \bar{Q}_2; \quad J_4 = K_4 = \bar{Q}_1 \bar{Q}_2 \bar{Q}_3$$

Синтезируем схему:



Практическая работа №30. Синтез реверсивного счетчика.

Цель работы: синтезировать схему реверсивного счетчика.

Пример выполнения практической работы.

Счетчики, которые в процессе работы могут менять направление счета называются реверсивными.

Методика синтеза реверсивных счётчиков объединяет в себе методики синтеза суммирующего и вычитающего счётчиков, т.е. сначала синтезируются вычитающий и суммирующий счётчики, потом вводится сигнал, отвечающий за направление счёта и все уравнения объединяются.

Ранее в практических работах №28 и №29 мы получили уравнения входов для суммирующего и вычитающего счётчиков (+ суммирующий счетчик, - вычитающий):

(+)

$$J_1 = \bar{Q}_4 \vee \bar{Q}_3; \quad K_1 = 1; \quad J_2 = Q_1; \quad K_2 = Q_1; \quad J_3 = Q_1 Q_2; \quad K_3 = Q_1 Q_2 \vee Q_4; \quad J_4 = Q_1 Q_2 Q_3; \quad K_4 = Q_3$$

(-)

$$J_1 = Q_1 \vee Q_2 \vee Q_3 \vee Q_4; \quad K_1 = 1; \quad J_2 = \bar{Q}_1 Q_3 \vee \bar{Q}_1 \bar{Q}_2 Q_4; \quad K_2 = \bar{Q}_1 Q_2; \quad J_3 = K_3 = \bar{Q}_1 \bar{Q}_2; \quad J_4 = K_4 = \bar{Q}_1 \bar{Q}_2 \bar{Q}_3$$

Эти уравнения нужно объединить и ввести в них сигнал направления счёта P , от которого будет зависеть направление счёта: если $P=1$, то $J1=(+)J1$, а если $P=0$, то $J1=(-)J1$, реализуется это так:

$$J1 = (P \cdot (+)J1) \vee (\bar{P} \cdot (-)J1)$$

здесь $(+)J1$ это уравнение входа $J1$ для суммирующего счётчика, т.е. вот это $J_1 = \bar{Q}_4 \vee \bar{Q}_3$, а $(-)J1$ - уравнение входа $J1$ для вычитающего счётчика, т.е. вот это $J_1 = Q_1 \vee Q_2 \vee Q_3 \vee Q_4$.

Если простым языком, то мы просто берём и умножаем уравнения входов суммирующего счётчика на P , а уравнения вычитающего на инверсный $\bar{P}$, после чего складываем их.

Запишем полное уравнение входа $J1$ для реверсивного счётчика:

$$J_1 = ((\bar{Q}_4 \vee \bar{Q}_3) \cdot P) \vee ((Q_1 \vee Q_2 \vee Q_3 \vee Q_4) \cdot \bar{P})$$

Записываем аналогично для остальных входов:

$K_1 = 1$;

$$J_2 = (Q_1 \cdot P) \vee ((\bar{Q}_1 Q_3 \vee \bar{Q}_1 \bar{Q}_2 Q_4) \cdot \bar{P}); \quad K_2 = (Q_1 P) \vee (\bar{Q}_1 Q_2 \bar{P});$$

$$J_3 = (Q_1 Q_2 P) \vee (\bar{Q}_1 \bar{Q}_2 \bar{P}); \quad K_3 = ((Q_1 Q_2 \vee Q_4) \cdot P) \vee (\bar{Q}_1 \bar{Q}_2 \bar{P});$$

$$J_4 = (Q_1 Q_2 Q_3 P) \vee (\bar{Q}_1 \bar{Q}_2 \bar{Q}_3 \bar{P}); \quad K_4 = (Q_3 P) \vee (\bar{Q}_1 \bar{Q}_2 \bar{Q}_3 \bar{P})$$

Аналогично работам №29 и №30 собираем схему.

Практическая работа №31. Разработка принципиальной схемы генератора двоичных чисел на D-триггерах.

Цель работы: спроектировать синхронный счетчик, реализующий заданную последовательность смены состояний.

Исходная последовательность чисел: 0, 7, 14, 15, 8, 1, 2, 9, 10, 3, 11, 4, 12, 6, 5.

1. Составляем таблицы истинности (таблицы переходов триггеров).

Десятичное число	Q ₃	Q ₂	Q ₁	Q ₀	D ₃	D ₂	D ₁	D ₀
0	0	0	0	0	0	1	1	1
7	0	1	1	1	1	1	1	0
14	1	1	1	0	1	1	1	1
15	1	1	1	1	1	0	0	0
8	1	0	0	0	0	0	0	1
1	0	0	0	1	0	0	1	0
2	0	0	1	0	1	0	0	1
9	1	0	0	1	1	0	1	0
10	1	0	1	0	0	0	1	1
3	0	0	1	1	1	0	1	1
11	1	0	1	1	0	1	0	0
4	0	1	0	0	1	1	0	0
12	1	1	0	0	0	1	1	0
6	0	1	1	0	0	1	0	1
5	0	1	0	1	0	0	0	0

Q₀ ... Q₃ - входные переменные

D₀ ... D₃ - логические функции

2. Составляем карты Карно (минимальных логических функций).

Триггер D₃

Q ₃ Q ₂ \ Q ₁ Q ₀	00	01	11	10
00	0	0	1	1
01	1	0	1	0
11	0	1	1	1
10	0	1	0	0

$$D_3 = Q_1 \bar{Q}_2 \bar{Q}_3 + Q_0 Q_1 \bar{Q}_3 + Q_1 Q_2 Q_3 + Q_0 \bar{Q}_1 Q_3 + \bar{Q}_0 \bar{Q}_1 Q_2 \bar{Q}_3$$

Триггер D₂

$Q_3Q_2 \backslash Q_1Q_0$	00	01	11	10
00	1	0	0	0
01	1	0	1	1
11	1	0	0	1
10	0	0	1	0

$$D_2 = \bar{Q}_0 Q_1 Q_3 + Q_1 Q_2 \bar{Q}_3 + \bar{Q}_0 \bar{Q}_1 \bar{Q}_3 + Q_0 Q_1 \bar{Q}_2 Q_3$$

Триггер D_1

$Q_3Q_2 \backslash Q_1Q_0$	00	01	11	10
00	1	1	1	0
01	0	0	1	0
11	1	0	0	1
10	0	1	0	1

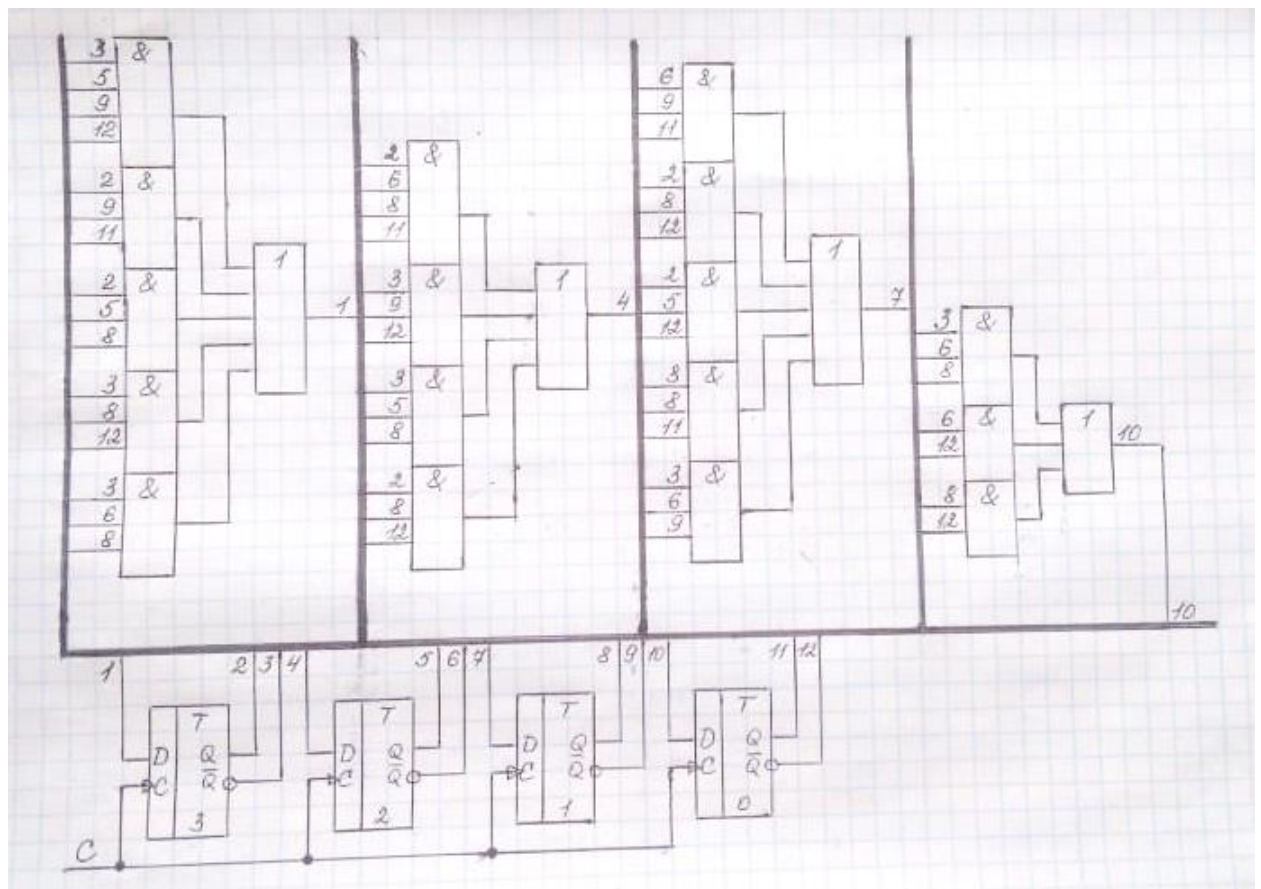
$$D_1 = \bar{Q}_1 \bar{Q}_2 \bar{Q}_3 + Q_0 Q_1 \bar{Q}_3 + \bar{Q}_0 Q_2 Q_3 + \bar{Q}_0 Q_1 Q_3 + Q_0 \bar{Q}_1 \bar{Q}_2$$

Триггер D_0

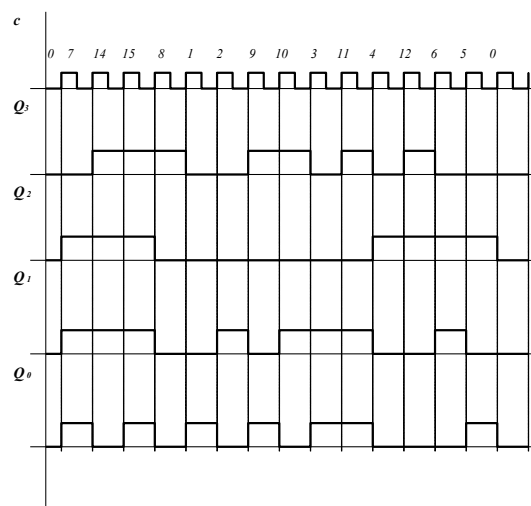
$Q_3Q_2 \backslash Q_1Q_0$	00	01	11	10
00	1	0	1	1
01	0	0	0	1
11	0	0	0	1
10	1	0	0	1

$$D_0 = \bar{Q}_0 Q_1 + \bar{Q}_0 \bar{Q}_2 + Q_1 \bar{Q}_2 \bar{Q}_3$$

3. Выполняем схему генератора



4. Строим временные диаграммы .



Варианты заданий на практическую работу

№ варианта	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Последовательности чисел	11	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8
	2	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0
	4	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9
	0	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1
	15	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10
	8	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2
	13	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11
	1	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3
	3	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12
	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4
	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13
	7	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5
	9	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6
	6	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Практическая работа №32. Исследование двойного преобразования (АЦП-ЦАП).

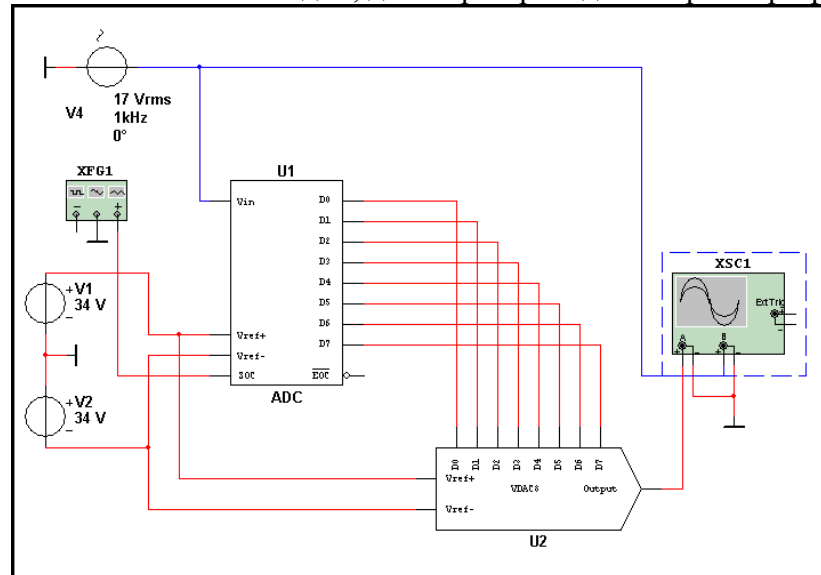
Цель работы: Освоить порядок моделирования цифроаналоговых и аналого- цифровых преобразователей

Практическая часть

Собрать схему, показанную на рисунке. Подать на вход АЦП синусоидальное напряжение, установив действующее напряжение на источнике U_{in} , равным номеру варианта. Опорные напряжения выбрать равными удвоенному значению номера варианта.

Временные диаграммы зарисовать в масштабе при различных частотах дискретизации: 4; 8; 16 и 32 кГц.

При частоте дискретизации 32 кГц дополнительно зарисовать еще три временные диаграммы при соединении между собой АЦП и ЦАП с помощью 1 разряда (D7), двух разрядов (D7, D6) и трех разрядов (D7, D6, D5). Другими словами, из восьми проводов, соединяющих АЦП и ЦАП, последовательно оставлять только один, два и три провода в старших разрядах.



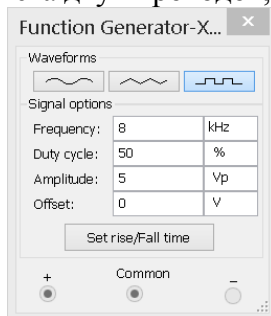
Методические указания

В предыдущей работе входное напряжение U_{in} вначале с помощью АЦП превращается в поток цифр. Затем цифры с помощью ЦАП обратно конвертируются в аналоговое напряжение. Таким образом, в идеальном случае после двойного преобразования выходное напряжение ЦАП должно полностью совпасть с входным напряжением U_{in} . Однако качество трансформации во многом определяется выбранной частотой дискретизации и разрядностью используемых преобразователей.

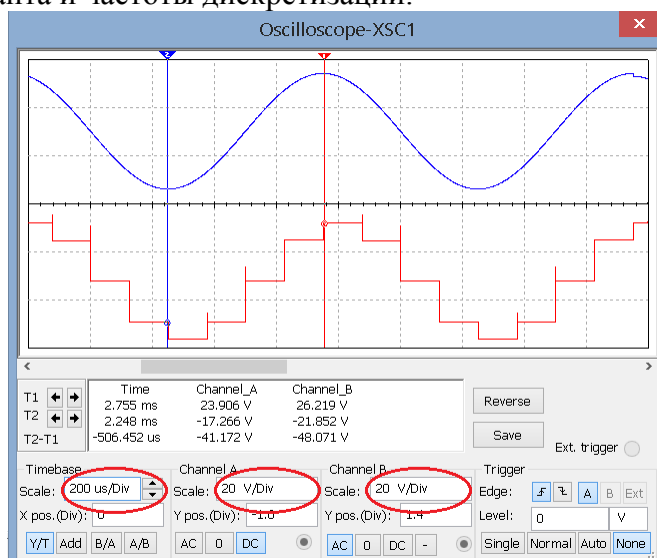
В процессе исследований данной схемы нужно дать качественную оценку точности преобразований в зависимости от частоты дискретизации и разрядности АЦП и ЦАП.

При проведении моделирования величины времени развертки и усиления по каждому каналу следует подобрать таким образом, чтобы полнее использовать весь экран осциллографа. Целесообразно время развертки установить таким, чтобы число периодов, укладывающихся на экране было 2...3. При исследовании двухканальных осциллограмм наглядность изображений можно повысить с помощью смещения графиков по вертикали (Y Position).

Во время моделирования осциллограф будет работать в двухканальном режиме. Важно, чтобы осциллограммы входного и выходного напряжений были изображены в одинаковых масштабах. Для повышения наглядности осциллограмм желательно использовать разноцветные изображения. Для этого нужно изменить цвета двух проводов, подходящих к осциллографу.



На следующем рисунке показаны настройки осциллографа для варианта 17 и тактовой частоте, равной 8 кГц. Овалами, отмечены органы управления, которые придется использовать при изменении номера варианта и частоты дискретизации.



Требования к отчету

Отчет подготавливается в электронном виде. Он должен содержать постановку задачи, скриншоты, которые показывают порядок решения задачи, схемы, таблицы с результатами расчетов и моделирования, временные диаграммы, необходимые комментарии и анализ полученных результатов.

Контрольные вопросы

1. Как определить напряжение наименьшей ступеньки восьмиразрядного ЦАП, если известны опорные напряжения?
2. Как с помощью осциллографа определить длительность одной ступеньки сигнала на выходе ЦАП?
3. Как с помощью осциллографа определить разность напряжений двух ступенек?
4. Какая частота дискретизации используется при записи (считывании) в оптических аудиодисках?

5. Сколько уровней квантования используется при записи (считывании) в оптических аудиодисках?
6. Сколько уровней квантования имеет 16-ти разрядный ЦАП?
7. Как с помощью теоремы Котельникова по известной максимальной частоте спектра определить необходимую частоту дискретизации АЦП?
8. Как по известной частоте дискретизации определить длительность одной ступеньки на выходе ЦАП?
9. Что называется, разрешением АЦП?
10. Приведите примеры аналоговых сигналов.
11. Почему внутри ЭВМ циркулируют преимущественно цифровые сигналы?
12. Приведите примеры цифроаналогового и аналого-цифрового преобразования.

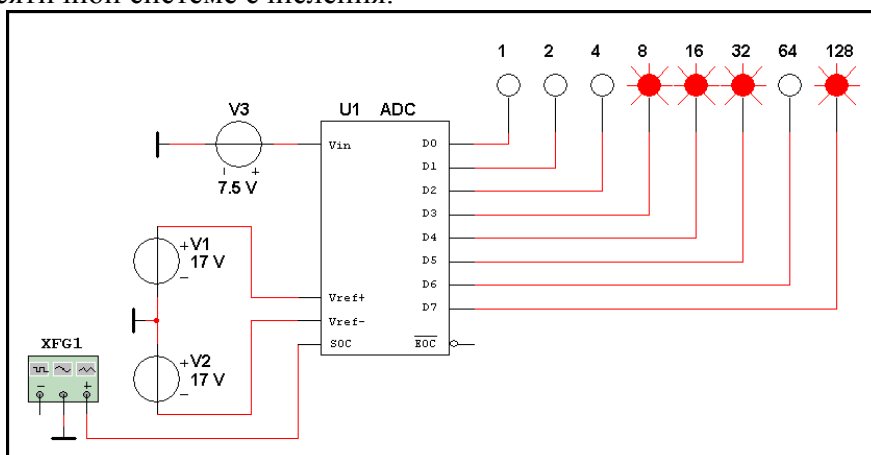
Практическая работа №33. Исследование аналого-цифровых преобразователей.

Цель работы: освоить порядок моделирования аналого-цифровых преобразователей.

Практическая часть

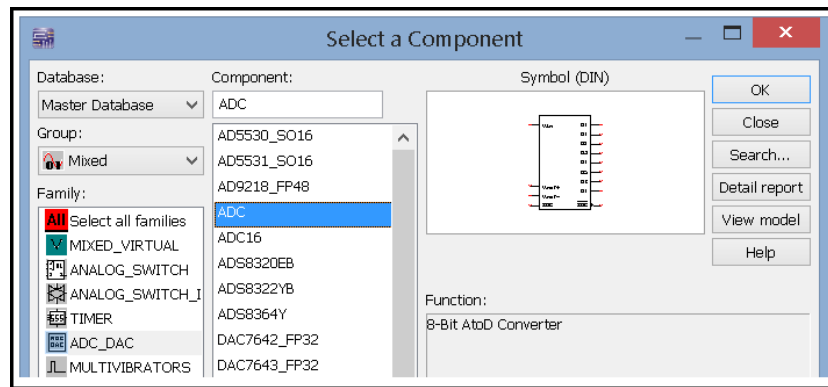
Собрать схему, показанную на рисунке. Установить на генераторе Function Generator частоту 1 кГц, прямоугольную форму колебаний (Waveforms) и амплитуду (Amplitude) в пределах 3...5 В. Выбрать опорные напряжения, равными номеру варианту. Поочередно установить семь значений входного напряжения U_{in} и определить выходной двоичный код. Результаты моделирования занести в таблицу.

В последнюю колонку таблицы поместить расчетное значение выходного кода, представленное в десятичной системе счисления.



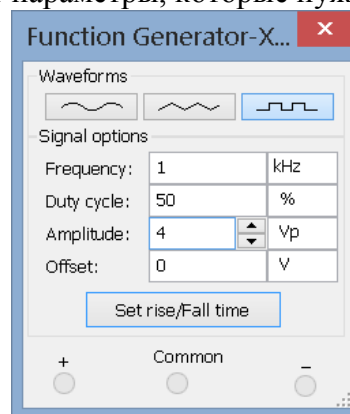
U_{in}, V	Показания индикаторов								D
	128	64	32	16	8	4	2	1	
$+U_{op}$									
$-U_{op}$									
$+U_{op}/2$									
$-U_{op}/2$									
$+U_{op}/10$									
$-U_{op}/10$									
0									

Для выбора АЦП из базы данных нужно пройти путь: панель **Components** группа **Mixed** семейство **ADC_DAC** компонент **ADC**.



Рассмотрим назначение элементов. На вход АЦП SOC подается сигнал от генератора Function Generator, который определяет частоту дискретизации. Так как в данном задании входное напряжение U_{in} постоянное, то не имеет значения, на какой частоте работает генератор (например, можно установить частоту 1 кГц).

Следующий рисунок показывает параметры, которые нужно установить на генераторе.



К выходам АЦП подключены индикаторы, которые обозначены в соответствии с их весовыми коэффициентами. Это делает удобным процедуру определения цифрового сигнала D , выраженного в десятичной системе счисления.

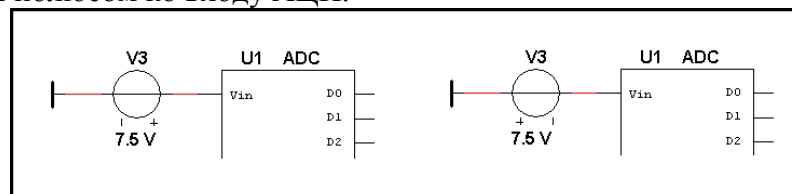
Выходной код АЦП рассчитывается по формуле:

$$D = 128 + U_{in} \cdot 128 / U_{op},$$

где U_{in} - входное аналоговое напряжение; D - выходное число, представленное в десятичной системе счисления.

Следует обратить внимание, что при $U_{in} = 0$ выходной десятичный код $D = 128$, при $U_{in} > 0$ выходной код $D > 128$, а при $U_{in} < 0$ выходной код $D < 128$.

Для смены полярности входного напряжения U_{in} необходимо отсоединить источник U_{in} и подключить его другим полюсом ко входу АЦП.



Требования к отчету

Отчет подготавливается в электронном виде. Он должен содержать постановки задач, скриншоты, которые показывают порядок решения задачи, схемы, таблицы с результатами расчетов и моделирования, временные диаграммы, необходимые комментарии и анализ полученных результатов.

Контрольные вопросы

1. Что называется разрешением АЦП?
2. Приведите примеры аналоговых сигналов.

3. Почему внутри ЭВМ циркулируют преимущественно цифровые сигналы?
4. Приведите примеры цифроаналогового и аналого-цифрового преобразования.

Практическая работа №34. Исследование цифроаналоговых преобразователей.

Цель работы: освоить порядок моделирования цифроаналоговых преобразователей.

Теоретическая часть

Различают две формы представления информации — непрерывную (аналоговую) и прерывистую (цифровую, дискретную). Непрерывная форма характеризует процесс, который не имеет перерывов и теоретически может изменяться в любой момент времени и на любую величину (например, крики животных, музыкальные произведения).

Все сигналы в живой и неживой природе существуют в аналоговой форме (влажность, давление, температура воздуха, освещенность, скорость ветра, нервные импульсы животных, речь человека и т.д.). Цифровые сигналы появились совсем недавно (середина прошлого века). Это искусственные сигналы, изобретенные людьми.

Цифровые сигналы могут изменяться лишь в определенные моменты времени и принимать лишь заранее обусловленные значения (например, только значения напряжений 0,2 и 3,5 В). Моменты времени возможного изменения уровня цифрового сигнала задает тактовый генератор конкретного цифрового устройства. Современные ЭВМ работают с цифровыми сигналами, поэтому необходимо уметь трансформировать аналоговые сигналы в цифровые (например, оцифровать изображение художественной картины, записать выступление музыкальной группы).

Для преобразования аналогового сигнала в цифровой сигнал требуется провести дискретизацию непрерывного сигнала во времени, квантование по уровню, а затем кодирование отобранных значений.

Дискретизация — замена непрерывного (аналогового) сигнала последовательностью отдельных во времени отсчетов (сэмплов) этого сигнала.

Квантование — разбиение диапазона уровней непрерывной величины на конечное число интервалов и округление входных уровней до разрешенных значений.

На рисунке схематично показан процесс преобразования аналогового сигнала в цифровой сигнал. Цифровой сигнал в данном случае может принимать лишь пять различных разрешенных уровней. Естественно, что качество такого преобразования невысокое. Из рисунка видно, что изменение цифрового сигнала происходит лишь в определенные моменты времени (в данном случае этих моментов одиннадцать).

Кодирование — представление результата измерения уровня в точке выборки (отсчета) двоичным числом.

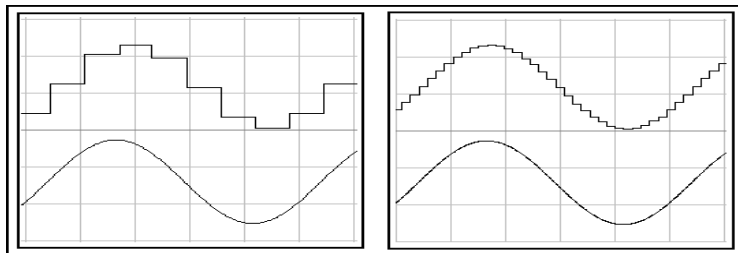
При преобразовании непрерывный сигнал заменяют последовательностью чисел. Показанный на рисунке непрерывный сигнал заменяется числами 2-3-4-4-4-3-2-2-3-4-4. Десятичная система счисления в рассматриваемом примере использована лишь для большей наглядности. Фактически аналоговый сигнал преобразуют в последовательность единиц и нулей. Результаты данного преобразования можно представить в виде таблицы.

Время	Десятичные числа	Двоичные числа
t_1	2	0010
t_2	3	0011
t_3	4	0100
t_4	4	0100
t_5	4	0100
t_6	3	0011
t_7	2	0010
t_8	2	0010
t_9	3	0011
t_{10}	4	0100

t11	4	0100
-----	---	------

Здесь цифровые сигналы представлены четырьмя разрядами двоичных чисел. Очевидно, что, чем больше разрядов у двоичных чисел (то есть чем больше число уровней квантования) и чем чаще во времени осуществляются отсчеты (выборки), тем точнее будет преобразован непрерывный сигнал в цифровой.

Образное представление о качестве оцифрованного сигнала дают кинофильмы, снятые с разной скоростью. Первые немые фильмы были сняты с частотой 16 кадров в секунду. Из-за низкой частоты съемки некоторые фазы движения объектов терялись, и перемещение персонажей на экране становилось комичным, дерганным. Переход на частоту 24 кадров в секунду сняло эту проблему, движение стало плавным.



Предыдущий рисунок показывает, как влияет частота дискретизации аналого-цифрового преобразования на качество оцифрованного сигнала. В нижней части диаграмм показан исходный аналоговый сигнал (синусоида частотой 1 кГц). В верхней части рисунков изображен сигнал после двойного преобразования при разных значениях частоты дискретизации. В данном случае аналоговый сигнал был вначале преобразован в цифровой, а затем цифровой сигнал обратно конвертирован в аналоговый. Рисунки отличаются использованной частотой дискретизации. В первом случае частота дискретизации составляла 8 кГц, во втором – 32 кГц. Из рисунка видно, что с ростом частоты дискретизации качество преобразования улучшается. На качество цифрового сигнала сильно влияет также его разрядность.

Необходимая частота дискретизации (при заданной наибольшей частоте в спектре преобразуемого сигнала) определяется с помощью **теоремы Котельникова** (в англоязычной литературе ее называют теоремой Найквиста — Шеннона или теоремой отсчетов). Теорема гласит: любой сигнал со спектром частот от 0 до $f_{\max}$ можно передать с необходимой точностью при помощи чисел, следующих друг за другом через интервалы времени, равные $1/2f_{\max}$.

Первое представление об аналоговом и цифровом способах хранения и распространения информации можно получить, рассматривая два способа записи звуковых сигналов: аналоговую и цифровую аудиозаписи. В аналоговой аудиозаписи непрерывный электрический сигнал, формируемый источником звука на выходе микрофона, с помощью магнитной головки наносится на движущуюся магнитную ленту. Недостатком аналогового способа обработки информации является то, что копия бывает всегда хуже оригинала, причем с увеличением числа перезаписей качество фонограммы постоянно ухудшается. Длительное хранение магнитной ленты приводит к ее размагничиванию и в итоге – к потере записи.

При цифровой аудиозаписи используется процесс выборки, заключающийся в периодическом измерении уровня (громкости, амплитуды) аналогового звукового сигнала (например, поступающего с выхода микрофона) и превращении полученного значения в последовательность двоичных чисел. Для преобразования аналогового сигнала в цифровой используется специальный конвертор, называемый аналогово-цифровой преобразователь (АЦП). В англоязычной литературе используется термин Analog Digital Converter (ADC). Сигнал на выходе АЦП представляет собой последовательность двоичных чисел, которая может быть обработана компьютером. Обратная конверсия цифрового сигнала в непрерывный сигнал осуществляется с помощью цифроаналогового преобразователя (ЦАП, DAC).



Качество аналогово-цифрового преобразования характеризует параметр, называемый разрешением. **Разрешение** — это количество уровней квантования, используемых для замены непрерывного аналогового сигнала цифровым сигналом. Восемьмиразрядная выборка позволяет получить только 256 различных уровней квантования цифрового сигнала, а шестнадцатиразрядная выборка — 65 536 уровней.

Еще один показатель качества трансформации непрерывного сигнала в цифровой сигнал — это частота дискретизации — количество преобразований аналог-цифра (выборок), производимое устройством в одну секунду. Этот показатель измеряют килогерцами (килогерц — тысяча выборок в секунду). Типичное значение частоты дискретизации лазерных (оптических) аудиодисков — 44,1 кГц.

Практическая часть

Задание 1. Исследование ЦАП

Собрать схему ЦАП, использующего взвешивающую резистивную матрицу. Установить напряжение на источнике напряжения V1 равным номеру варианта. Коммутируя ключи 1, 2, 3 и 4, сделать 16 измерений и заполнить таблицу 1. В последнюю колонку занести показания вольтметра U1.

На рисунке показан момент подачи на вход ЦАП комбинации цифровых сигналов 1011В (ключ старшего разряда S4 находится внизу).

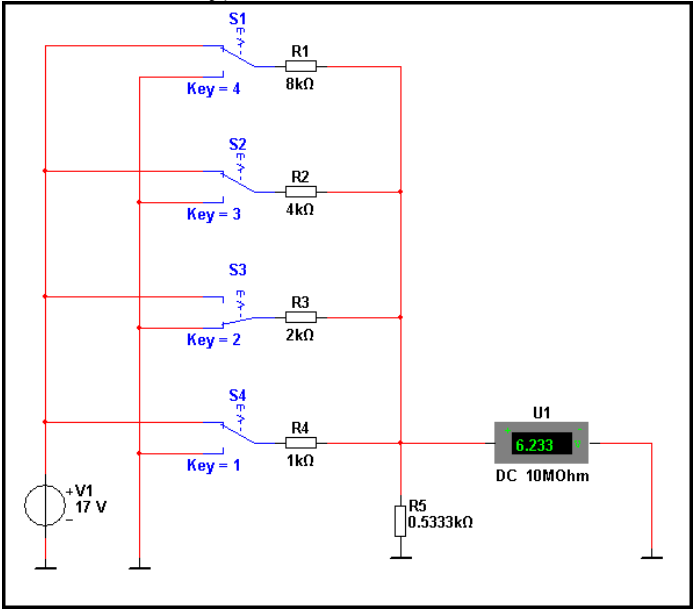
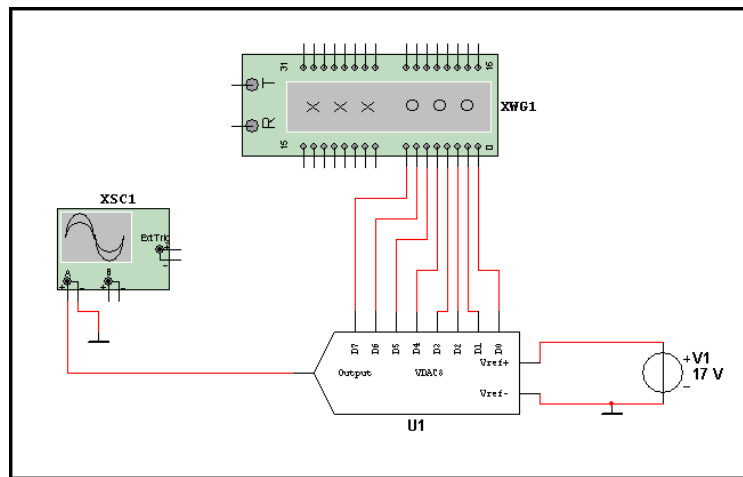


Таблица 1

Цифровой сигнал				Аналоговый сигнал U, В
Ключ 1	Ключ 2	Ключ 3	Ключ 4	
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
...	...	...	...	
1	1	1	1	

Задание 2. Исследование микросхемы ЦАП

Составить схему, показанную на следующем рисунке. Преобразовать 16 букв своей фамилии и имени (без пробелов) в десятичную систему счисления. Если число символов в фамилии и имени меньше 16, то следует использовать часть букв отчества. Первые буквы фамилии, имени и отчества должны быть заглавными, остальные – строчные. Для записи использовать кириллицу.



Ввести 16 чисел в генератор слов (Word Generator) и поочередно подать их на ЦАП 16. Поместить в отчет временные диаграммы, которые формируются на экране осциллографа в процессе моделирования ЦАП. С помощью осциллографа измерить аналоговое напряжение U , соответствующее каждому значению цифрового сигнала.

Опорное напряжение U_{op} в схеме должно быть установлено равным номеру варианта (на рисунке показано опорное напряжение для варианта 17). Рассчитать значение выходного напряжения U_p , которое соответствует каждой входной комбинации. Результаты исследований занести в таблицу.

Сопоставить результаты расчетов и измерений.

Буквы	Десятичный код, D	Аналоговое напряжение, В	
		Расчетное, U_p	Измеренное, U

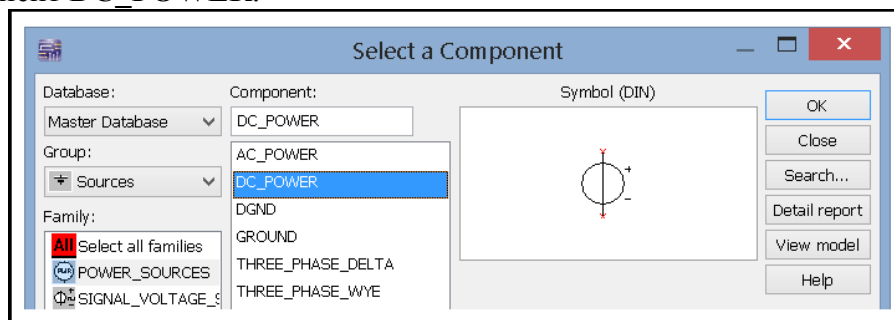
Порядок выполнения практической работы

Методические указания к заданию 1.

Принцип работы ЦАП, использующего взвешивающую резистивную матрицу, заключается в изменении тока, протекающего через резистор R_5 . Чем больше ключей формируют высокое напряжение, тем большее напряжение формируется на резисторе R_5 . Легко заметить, что вклад каждого ключа (каждого двоичного разряда) различен. Наибольший вклад в выходное напряжение дает ключ S_4 . Это объясняется тем, что через резистор R_4 протекает наибольший ток, так как номинал этого резистора наименьший. Минимальный квант выходного напряжения формируется ключом S_1 (младший разряд двоичного числа).

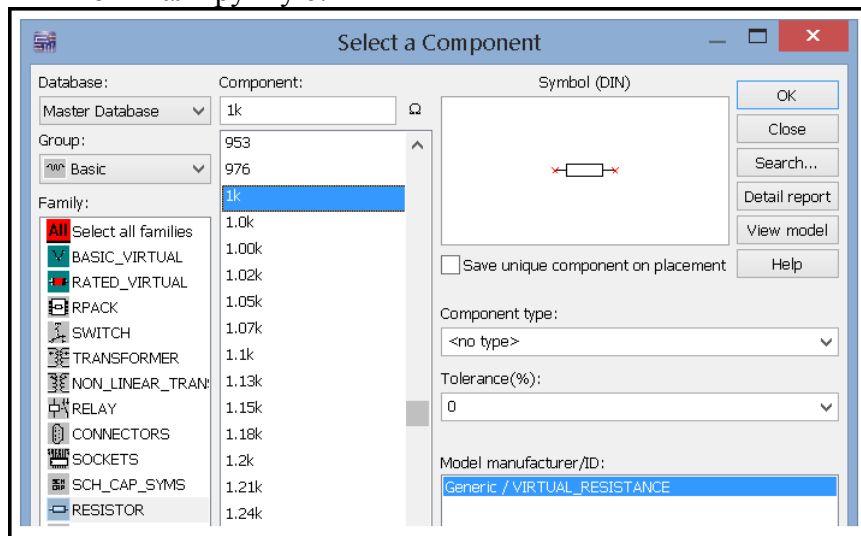
Еще одна конструктивная особенность взвешивающей резистивной матрицы состоит в том, что номиналы резисторов выбраны такими же, как и веса разрядов двоичных чисел: 1, 2, 4, 8. Это простейший способ построения ЦАП.

Извлечение источника напряжения из базы данных Multisim происходит в такой последовательности: панель **Components** группа **Sources** семейство **POWER_SOURCES** компонент **DC_POWER**.



Установка заданного напряжения происходит стандартно: двойной щелчок по установленному элементу открывает диалоговое окно, которое позволяет сделать необходимые корректировки параметров компонента.

Для установки в схему резисторов нужно выполнить команды: панель **Components** группа **Basic** семейство **RESISTOR** в списке компонентов выбрать нужный номинал резистора или установить номинал вручную.



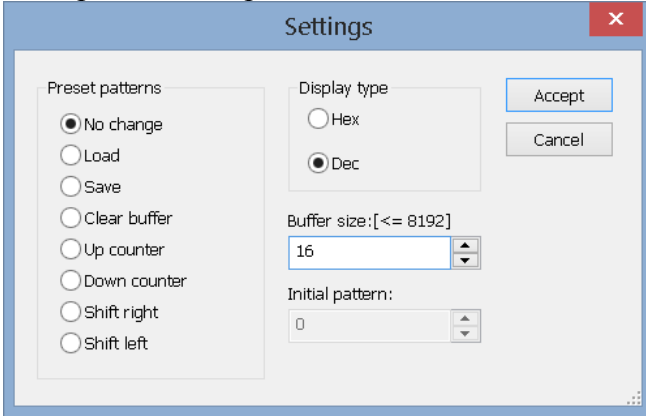
Методические указания к заданию 2

Микросхема ЦАП находится в базе данных Multisim. Для ее установки нужно пройти путь: панель **Components** группа **Mixed** семейство **ADC_DAC** компонент **VDAC**.

При исследовании ЦАП цифровой сигнал формируется с помощью генератора слов (Word Generator). В каждом варианте цифровые сигналы разные и определяются буквами фамилии, имени и отчества студента. Преобразование русских букв в десятичные числа производится на основании кодовой таблицы СР-1251.

пробел	32	!	33	"	34	#	35	\$	36
%	37	&	38	'	39	(	40	)	41
*	42	+	43	,	44	-	45	.	46
/	47	0	48	1	49	2	50	3	51
4	52	5	53	6	54	7	55	8	56
9	57	:	58	;	59	<	60	=	61
>	62	?	63	@	64	A	65	B	66
C	67	D	68	E	69	F	70	G	71
H	72	I	73	J	74	K	75	L	76
M	77	N	78	O	79	P	80	Q	81
R	82	S	83	T	84	U	85	V	86
W	87	X	88	Y	89	Z	90	[	91
\	92	]	93	^	94	_	95	`	96
a	97	b	98	c	99	d	100	e	101
f	102	g	103	h	104	i	105	j	106
k	107	l	108	m	109	n	110	o	111
p	112	q	113	r	114	s	115	t	116
u	117	v	118	w	119	x	120	y	121
z	122	A	192	Б	193	В	194	Г	195
Д	196	Е	197	Ж	198	З	199	И	200
Й	201	К	202	Л	203	М	204	Н	205
О	206	П	207	Р	208	С	209	Т	210
У	211	Ф	212	Х	213	Ц	214	Ч	215
Ш	216	Щ	217	Ъ	218	Ы	219	Ь	220
Э	221	Ю	222	Я	223	а	224	б	225

Для изменения объема буфера нужно вызвать диалоговое окно **Settings** (Установки). Вызов этого окна происходит при щелчке по кнопке **Set...**

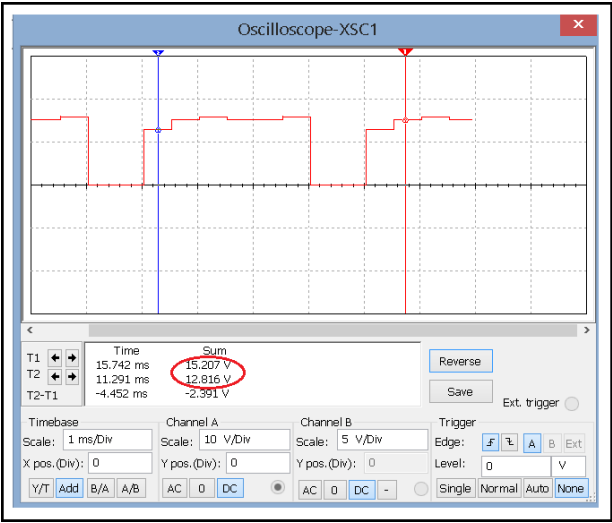


Выбрав подходящую систему счисления (Hex, Dec), нужно в окне **Buffer size** (Размер буфера) указать число используемых символов.

Временную развертку на осциллографе (Timebase) следует установить 1 ms/Div, а чувствительность 5 V/Div (для вариантов 1...8) или 10 V/Div (для вариантов 9...16).

При измерении аналогового сигнала с помощью осциллографа нужно открыть детальное изображение осциллографа (двойной щелчок по свернутому изображению). Считывание показаний осциллографа следует производить с помощью одного или двух цветных маркеров.

Пример настройки осциллографа (состояние управляющих элементов) приведен на следующем рисунке. Осциллограф в данном случае работает в одноканальном режиме и канал B (Channel B) не используется.



Овалом показана область вывода результатов измерения.

Выходное напряжение восьмиразрядного ЦАП рассчитывается по формуле:

$$U_p = D \cdot U_{op} / 256,$$

где D - десятичный код цифрового сигнала; U_{op} – опорное напряжение (определяется номером варианта).

Если $U_{op} = 17$ В, то приведённая формула будет иметь компактный вид, удобный для проведения расчётов в варианте 17:

$$U_p = D \cdot 17 / 256 = D \cdot 0,0664.$$

Сделанные округления приведут к появлению небольших различий между расчетными и измеренными значениями. В следующей таблице приведены результаты моделирования и расчетов для слова Бендер.

Буквы	Десятичный код, D	Аналоговое напряжение, В	
		Расчётное, U_p	Измеренное, U
Б	193	12,816	12,816

е	229	15,207	15,207
н	237	15,738	15,738
д	228	15,141	15,141
е	229	15,207	15,207
р	240	15,937	15,937

Анализ таблицы показывает, что наименьшее различие выходных напряжений (для букв «д» и «е») составляет 0,066 В.

Требования к отчету

Отчет подготавливается в электронном виде. Он должен содержать постановки задач, скриншоты, которые показывают порядок решения задачи, схемы, таблицы с результатами расчетов и моделирования, временные диаграммы, необходимые комментарии и анализ полученных результатов.

Контрольные вопросы

1. Как определить напряжение наименьшей ступеньки восьмиразрядного ЦАП, если известны опорные напряжения?
2. Как с помощью осциллографа определить длительность одной ступеньки сигнала на выходе ЦАП?
3. Как с помощью осциллографа определить разность напряжений двух ступенек?
4. Какая частота дискретизации используется при записи (считывании) в оптических аудиодисках?
5. Сколько уровней квантования используется при записи (считывании) в оптических аудиодисках?
6. Сколько уровней квантования имеет 16-ти разрядный ЦАП?
7. Как с помощью теоремы Котельникова по известной максимальной частоте спектра определить необходимую частоту дискретизации АЦП?
8. Как по известной частоте дискретизации определить длительность одной ступеньки на выходе ЦАП?

Практическая работа №35. Электронный усилитель. Биполярные транзисторы.

Цель работы: изучение принципа работы усилителя; изучение принципа действия транзисторов.

Краткие сведения из теории

Электронный усилитель (ЭУ) - усилитель электрических сигналов, в усилительных элементах которого используется явление электрической проводимости в газах, вакууме и полупроводниках. Электронный усилитель может представлять собой как самостоятельное устройство, так и блок (функциональный узел) в составе какой-либо аппаратуры - радиоприемника, магнитофона, измерительного прибора и т.д.

Задание на практическую работу

Задача 1. Определите недостающие параметры схемы, приведенной на рисунке 1. Исходные данные приведены в таблице 1.

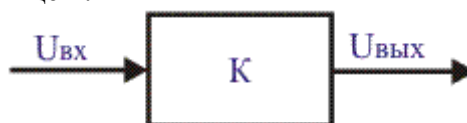


Рисунок 1 - Схема электронного усилителя

Таблица 1 - Исходные данные для задачи 1

№ варианта	$U_{вх}$, В	$U_{вых}$, В	К
01	5	?	10

02	5	25	?
03	2	10	?
04	?	25	10
05	?	15	10
06	10	15	?
07	12	30	?
08	?	45	25
09	10	?	15
10	8	16	2

Задача 2. Определите не достающие параметры схемы, приведенной на рисунке 2. Исходные данные приведены в таблице 1.2.

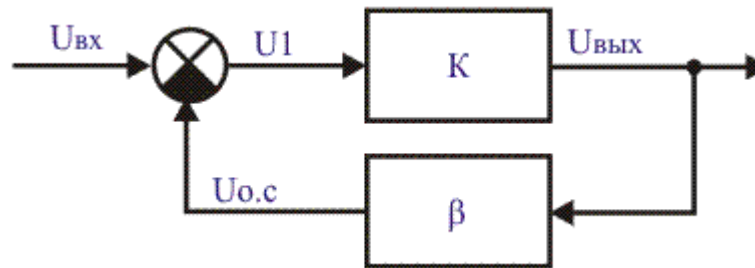


Рисунок 2 - Схема электронного усилителя

Таблица 2 - Исходные данные для задачи 2

№ варианта	$U_{вх}, В$	$U_{вых}, В$	$U1, В$	β	K
01	5	-3	10	?	?
02	?	?	3	5	30
03	10	?	3	?	2
04	?	25	10	?	5
05	20	15	?	?	?
06	?	15	?	5	25
07	12	30	?	?	?
08	10	45	?	7	?
09	?	30	15	?	20
10	?	?	5	10	15

Задача 3. Биполярный транзистор включен по схеме с общим эмиттером. По приведенным на рисунке 3 характеристикам определить недостающие и h – параметры транзистора. Варианты заданий приведены в таблице 3. Напряжения питания каскада для всех вариантов принять равным 30 В.

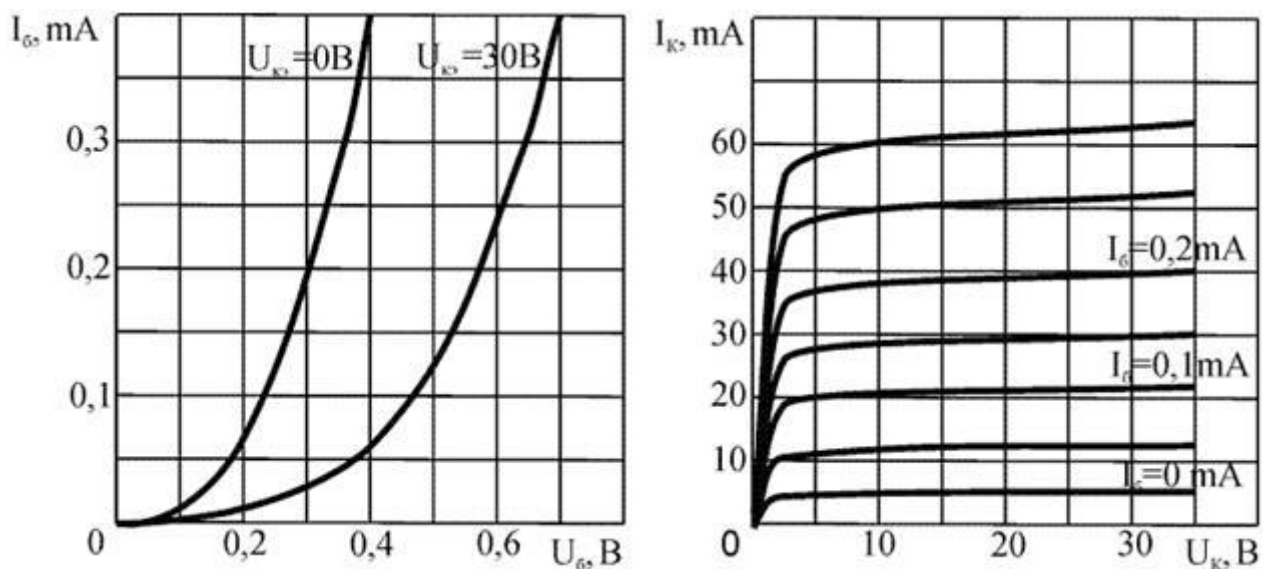


Рисунок 3 - Биполярный транзистор включенный по схеме с ОЭ

Таблица 3 - Исходные данные для задачи 3

Вариант №	$I_{\text{б}}$ мкА	$I_{\text{к}}$ мА	$I_{\text{с}}$ мА	α	$U_{\text{к}}$ В
1	150	?	?	?	?
2	?	40	?	?	?
3	?	?	?	?	15
4	100	?	?	?	?
5	?	25	?	?	?
6	?	?	?	?	20
7	50	?	?	?	?
8	?	50	?	?	?
9	?	?	?	?	25
10	200	?	?	?	?

Содержание отчета

- 1 Титульный лист с названием практической работы, номером варианта, фамилиями студентов и группы.
- 2 Цель работы и задание на практическую работу.
- 3 Результаты расчетов.
4. Выводы по полученным данным расчетов.

Контрольные вопросы

1. Что означает понятие идеальные операционные усилители?
2. Идеальный ОУ обладает какими характеристиками?
3. Особенности устройства СВЧ биполярных транзисторов (БТ).
4. Особенности устройства СВЧ полевых транзисторов (ПТ).

Практическая работа №36. Исследование параметров операционных усилителей

Цель работы – экспериментальное исследование параметров операционных усилителей (ОУ): напряжения смещения, АЧХ, ФЧХ и скорости нарастания выходного напряжения.

Экспериментальная часть

1. Создать модель ОУ в среде Electronics Workbench в соответствии с вариантом задания таблицы 1. Для этого щелкнуть два раза клавишей указания мыши на изображении ОУ и выбрать в появившемся окне **3 – Terminal Opamp Properties** в разделе **Library** библиотеку **default**, а затем в разделе **Model** – тип ОУ **ideal**. Выбрать последовательно клавишей указания мыши кнопки **Copy** и **Paste**, записать латинскими буквами в появившемся окне тип ОУ в соответствии с вариантом задания и нажать кнопку **OK**. В результате в разделе **Model** добавится новый тип ОУ. Для корректировки его параметров нажать кнопку **Edit** и установить значения параметров **Input offset voltage [VOS]** (напряжение смещения $U_{см}$), **Input resistance [RI]** (входное сопротивление $R_{вх}$), **Open-loop gain [A]** (коэффициент усиления K_y), **Slew rate [SR]** (скорость нарастания выходного напряжения $V_{Увых}$), **Unity-gain bandwidth [FU]** (частота единичного усиления f_1) и **Output resistance [RO]** (выходное сопротивление $R_{вых}$) в соответствии с таблицей 1. Значения других параметров оставить без изменения.

Таблица 1

№ варианта	Тип ОУ	$U_{см}$, мВ (VOS)	$R_{вх}$, МОм (RI)	K_y (A)	$V_{Увых}$, В/мкс (SR)	f_1 , МГц (FU)	$R_{вых}$, Ом (RO)
1	2	3	4	5	6	7	8
1	K140УД6А	5	3	70000	2,5	1	150
2	K140УД7	9	0,3	50000	0,3	0,8	150
3	K140УД8А	20	1000	50000	5	1	200
4	K140УД9	5	0,3	35000	0,5	1	200
5	K140УД10	5	1	50000	50	15	200
6	K140УД11	10	0,2	25000	20	5	200
7	K140УД12	6	50	50000	2	1	1000
8	K140УД14А	2	30	50000	0,05	0,3	300
9	K140УД17А	0,025	100	300000	0,1	0,4	200
10	K140УД20А	6	0,4	50000	0,3	0,8	150
11	K153УД1	7,5	0,5	50000	0,2	1	150
12	K153УД2	5	0,3	50000	0,5	1	300
13	K153УД3	2	0,1	25000	1	1	200
14	K153УД4	5	0,1	5000	0,12	0,3	200
15	K153УД5А	2,5	1	1000000	0,1	0,2	100
16	K154УД1	5	1	200000	10	1	300
17	K154УД2	2	5	100000	10	1	300
18	K157УД1	5	0,3	50000	0,5	0,5	20
19	K544УД1А	20	150000	50000	2	5	200
20	K544УД2А	20	300000	20000	20	15	200
21	K140УД20А	6	0,4	50000	0,3	0,8	150
22	K153УД2	5	0,3	50000	0,5	1	300
23	K154УД1	5	1	200000	10	1	300
24	K157УД1	5	0,3	50000	0,5	0,5	20

1. Собрать схему для исследования влияния напряжения смещения на выходное напряжение ОУ в прямом включении в соответствии с рисунком 1. Вольтметр включить в режиме измерения постоянного напряжения (**DC**).

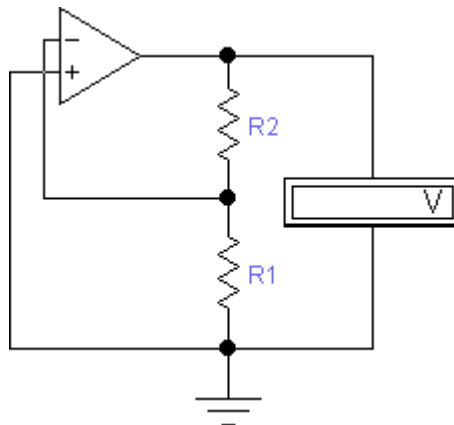


Рисунок 1

Измерить выходное напряжение ($U_{\text{вых}}$) ОУ и занести в таблицу для следующих значений $R1$ и $R2$, приведенных в таблице 2.

Таблица 2

$R1$, кОм	$R2$, кОм	$U_{\text{вых}}$, В	$U_{\text{см}}$, мВ
1	10		
1	100		

2. Собрать схему для исследования АЧХ и ФЧХ ОУ, включенного по схеме повторителя напряжения, в соответствии с рисунком 2.

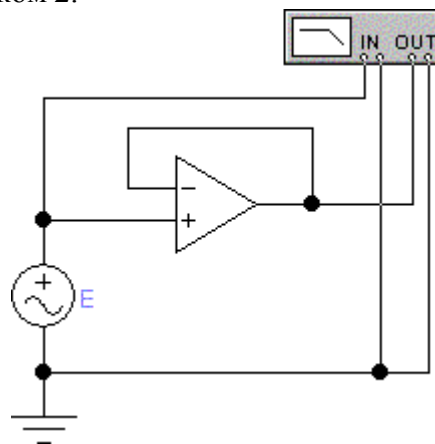


Рисунок 2

Установить напряжение источника ЭДС $E = 1$ В и частоту $f = 1$.

Установить **Code Plotter** в режим **Magnitude**. По вертикальной и горизонтальной осям выбрать логарифмический масштаб (**Log**).

Получить АЧХ ОУ на экране **Code Plotter**.

Установить **Code Plotter** в режим **Phase**. По вертикальной и горизонтальной осям выбрать логарифмический масштаб (**Log**).

Получить ФЧХ ОУ на экране **Code Plotter**.

3. Собрать схему инвертирующего усилителя на ОУ в соответствии с рисунком 3 и исследовать влияние сопротивления нагрузки $R3$ на величину выходного напряжения ОУ.

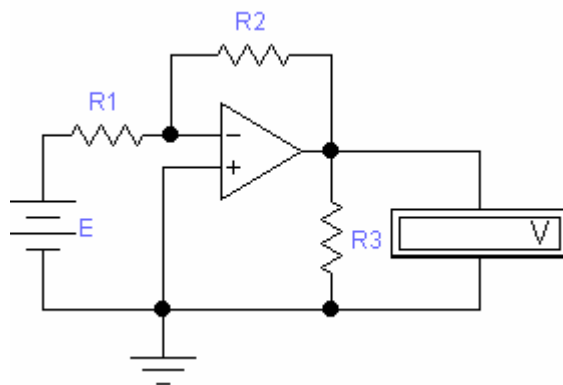


Рисунок 3

Последовательно устанавливая различные значения резисторов из таблицы 3, измерить выходное напряжение ОУ $U_{\text{вых}}$ и занести полученные значения в таблицу 3.

Таблица 3

R3, кОм	0,1	0,2	0,5	1	2	3	4	5
$U_{\text{вых}}$, В								

4. Собрать схему инвертирующего усилителя в соответствии с рисунком 4 для исследования переходных параметров ОУ.

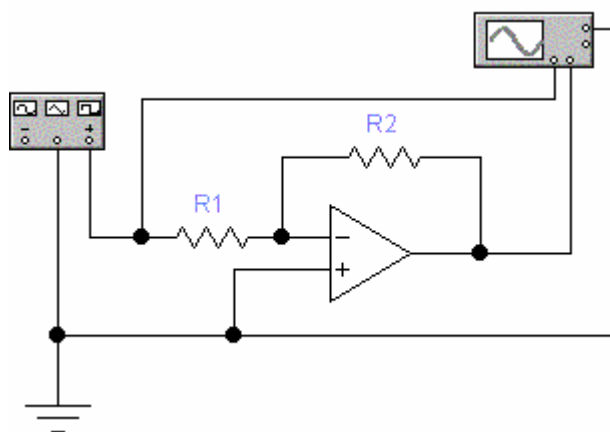


Рисунок 4

Последовательно устанавливая значение резистора R2, выходное напряжение $U_{\text{ги}}$ и частоту $f_{\text{ги}}$ генератора импульсов в соответствии с таблицей 4, зарисовать соответствующие осциллограммы и определить время установления ОУ $t_{\text{уст}}$ с точностью 5%.

Результаты занести в таблицу 4.

Таблица 4

R1, кОм	R2, кОм	$U_{\text{ги}}$, В	$f_{\text{ги}}$, кГц	$t_{\text{уст}}$, мкс	$V_{U_{\text{вых}}}$, В/мкс
10	100	0,5	10		
10	10	5,0	10		

Обработка экспериментальных данных

1. Вычислить и занести в таблицу 2 значения напряжения смещения ОУ $U_{\text{см}}$ по формуле:

$$U_{\text{см}} = U_{\text{вых}}/K,$$

где $K=1+(R2/R1)$.

Сравнить полученные значения с соответствующими значениями из таблицы 1 и сделать выводы.

2. Определить по полученной АЧХ (п.2.6) граничную частоту $f_{\text{гр}}$ для исследуемого ОУ.

Построить график зависимости $U_{\text{вых}}(R_n)$ по значениям табл.3.

3. Оформить отчет, включив в него все исследуемые схемы, графики и таблицы с результатами экспериментов и расчетов, и сделать выводы.

Контрольные вопросы

1. Какова область применения ОУ?
2. Понятие об идеальном ОУ и классификация ОУ.
3. При каких условиях реальный ОУ можно считать идеальным?
4. Привести входные параметры ОУ.
5. Привести передаточные параметры ОУ.
6. Привести выходные параметры ОУ.
7. Привести переходные параметры ОУ.
8. Что показывает амплитудно-частотная характеристика ОУ?
9. В чем заключаются основные достоинства ОУ, благодаря которым они широко применяются?

Практическая работа №37. Исследование статических свойств операционных усилителей

Цель работы: познакомиться с основными свойствами операционных усилителей, работающих при постоянном входном напряжении в установившемся режиме. Изучить влияние параметров операционных усилителей на работу измерительных узлов, построенных на их основе.

Порядок выполнения работы

В ходе работы необходимо провести исследование двух типов ОУ, указанных преподавателем. Предстоит измерить входное напряжение смещения, входные токи и коэффициент ослабления синфазного сигнала. Перед началом работы необходимо зарисовать в тетради по две таблицы идентичные таблице 1. В них будут заноситься результаты измерений для двух ОУ.

1. Измерение напряжения смещения операционного усилителя.

1.1. Собрать схему, показанную на рисунке 1, а. Желательно проследить, чтобы обозначения на схеме ($R_1 \dots R_4$, $J_1 \dots J_3$) совпадали с рисунком. На этом этапе параметры элементов, такие как сопротивления резисторов, не менять.

1.2. Установить сопротивления резисторов и перевести ключи в положение согласно таблице 2.

Таблица 1 – Результаты измерений параметров ОУ

Пункт	Измеряемый параметр	Измеренное значение	R_1 , кОм	R_2 , кОм	R_3 , кОм
1.4	$U_{см\ вых}$, мВ (до балансировки)				
1.5	$U_{см}$, мВ (расчетное)				
1.7	$U_{см\ вых}$, мВ (после балансировки)				
2.1	$I_{вх\ и}$, нА (расчетное)				
2.2	$I_{вх\ н}$, нА (расчетное)				

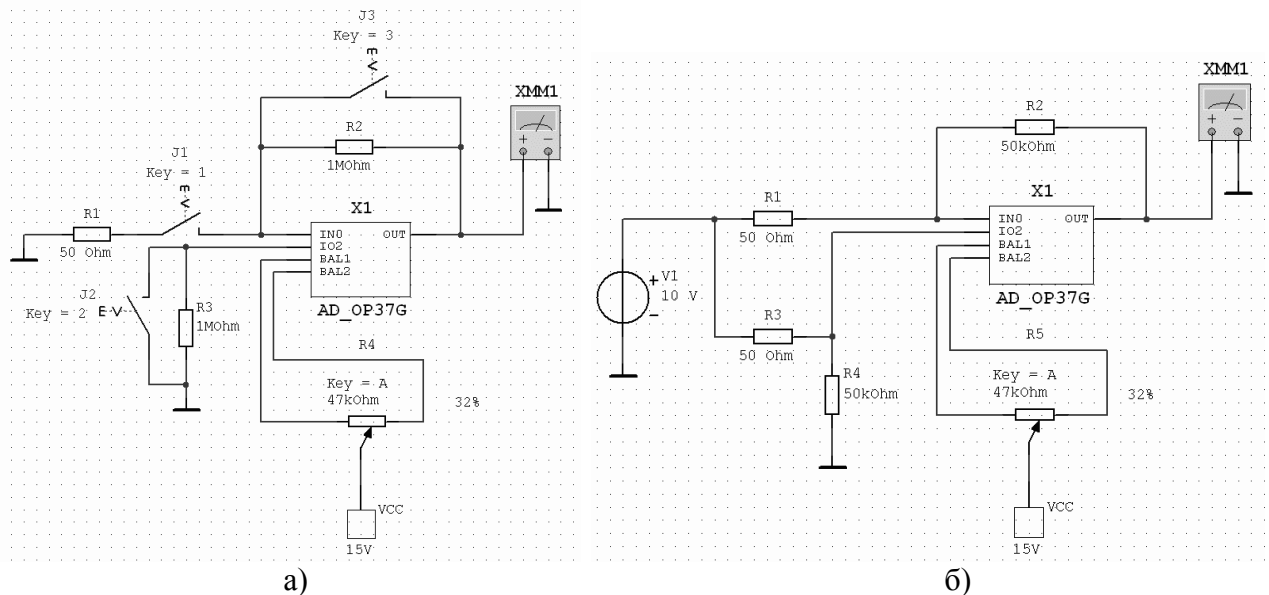


Рисунок 1 – Схемы для измерения напряжения смещения и входных токов (а) и коэффициента подавления синфазного сигнала (б).

Таблица 2 – Данные для экспериментов п. п. 1–3

Измеряемый параметр	Положение ключей			Рекомендуемые значения сопротивлений			Расчетная формула
	J_1	J_2	J_3	R_1	R_2	R_3	
$U_{см}$				100 Ом	100 кОм	–	$U_{см} = \frac{U_{вхл} R_1}{R_1 + R_2}$
$I_{вх и}$				–	1 МОм	–	$I_{вх и} = U_{вхл} / R_2$
$I_{вх н}$				–	–	1 МОм	$I_{вх н} = U_{вхл} / R_3$
$\Delta I_{вх и}$				–	1 МОм	1 МОм	$\Delta I_{вх и} = U_{вхл} / R, R=R_2=R_3$

1.3. Сопротивление балансирующего потенциометра выбрать из диапазона 10...100 кОм. Перед началом измерений убедиться, что его движок установлен в положение 50%.

1.4. При помощи мультиметра, подключенного к выходу ОУ, зафиксировать выходное напряжение $U_{см вых}$, занести в таблицу 2.1. Указать значения сопротивлений R_1, R_2 .

Замечание. Для удобства строки таблицы пронумерованы в соответствии с пунктами лабораторной работы.

1.5. Рассчитать входное напряжение смещения $U_{см}$ по формуле в таблице 2, результат записать в таблицу 1.

1.6. Отбалансировать операционный усилитель. Для этого, изменяя положение движка потенциометра, добиться максимально близкого к нулю (по модулю) выходного напряжения ОУ.

1.7. Зафиксировать выходное напряжение $U_{см вых}$ после балансировки, занести в таблицу 1. Записать сопротивление и положение движка потенциометра в процентах (в отчет не включать).

2. Измерение входных токов и разности входных токов.

2.1. Провести измерение входного тока инвертирующего входа ОУ. Для этого установить сопротивления резисторов и положение ключей согласно таблице 2. По измеренному $U_{вых}$ рассчитать $I_{вх и}$, по соответствующей формуле из таблицы. Результат занести в таблицу 1.

2.2. Аналогичным образом определить входной ток неинвертирующего входа и разность входных токов.

3. Устранение влияния входных токов при помощи корректирующего сопротивления.

3.1. Установить положение ключей так, как это было сделано для измерения выходного напряжения смещения. Установить $R_1=R_2=500$ кОм.

3.2. Зафиксировать выходное напряжение $U_{вых}$, занести в таблицу.

2.1. Указать значения сопротивлений R_1, R_2 .

3.3. Рассчитать сопротивление корректирующего резистора $R_{кор}$ по формуле

$$R_{кор} = \frac{R_1 R_2}{R_1 + R_2}.$$

3.4. Установить $R_3=R_{кор}$. Разомкнуть ключ J_2 . Зафиксировать выходное напряжение $U_{вых}$, занести в таблицу 1. Указать значения сопротивления R_3 .

3.5. Повторить измерения по пунктам 1–3 для второго типа ОУ.

4. Измерение коэффициента подавления синфазного сигнала. Схема эксперимента показана на рисунке 1, б).

4.1. Установить сопротивления резисторов так, чтобы получить коэффициент усиления порядка 1000, причем $R_1=R_3$ и $R_2=R_4$. ОУ следует тщательно отбалансировать. Можно воспользоваться записанным ранее положением движка потенциометра (см. пункт 1.6).

4.2. Установить входное напряжение порядка 10 В.

4.3. Измерить и записать выходное напряжение. Рассчитать $K_{осс}$ по формуле

$$K_{осс} = \frac{U_{вх}}{U_{вых}} \cdot \frac{R_1 + R_2}{R_1}$$

и выразить результат в децибелах. Записать условия эксперимента: значения сопротивлений резисторов и входное напряжение.

4.4. Повторить измерения для второго ОУ.

В отчет включить сводную таблицу измеренных параметров (таблица 3). Сделать вывод о соответствии результатов лабораторной работы справочным данным (таблица 1.1), классифицировать исследованные ОУ согласно параграфу 1.2.

Таблица 3 – Параметры исследованных ОУ

Тип ОУ	$U_{см}$, мВ	$I_{вх}$, нА	$\Delta I_{вх}$, нА	$K_{осс}$, дБ
ОУ №1				
ОУ №2				

Контрольные вопросы

1. Какими свойствами должен обладать идеальный усилитель?
2. Перечислите основные параметры ОУ, поясните смысл каждого из них, укажите диапазон типовых значений. Поясните суть их влияния на качество усилителя при использовании ОУ в измерительных цепях.
3. Изобразите схемы измерения основных параметров ОУ. Запишите формулы для расчета параметров по результатам измерений.
4. Изобразите основные схемы включения ОУ. Запишите приближенную и точную формулы, выражающие коэффициент передачи и входное сопротивление схемы.
5. Дайте сравнительную характеристику инвертирующей и неинвертирующей схем усиления.
6. Приведите классификацию ОУ. Каковы диапазоны параметров, характерные для каждой классификационной группы? В чем заключаются достоинства и недостатки каждой группы?

Практическая работа №38. Исследование динамических свойств операционных усилителей

Цель работы: познакомиться с основными свойствами операционных усилителей, работающих при переменном входном напряжении и в переходных процессах. Изучить влияние параметров операционных усилителей на работу измерительных узлов, построенных на их основе.

Порядок выполнения работы

1. Исследование АЧХ усилителя.

Схема эксперимента показана на рисунке 1.

Выбрав сопротивления резисторов R_1 , R_2 , установить коэффициент усиления $K=10$.

Установить частоту входного сигнала 50...100 Гц; амплитуду 1...10 мВ; форма — синусоидальная. Амплитуда должна выбираться так, чтобы ОУ не насыщался. Если ОУ обладает большим смещением, его необходимо предварительно отбалансировать.

Записать в таблицу 1 входное напряжение $U_{вх}$, измерить и записать выходное напряжение $U_{вых}$.

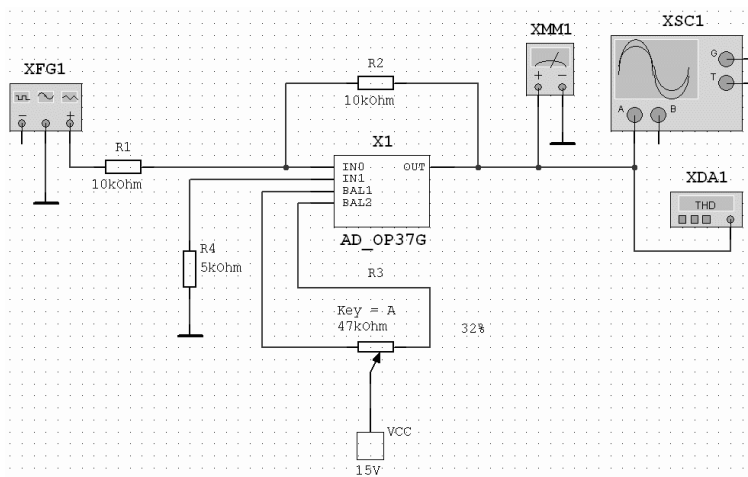


Рисунок 1 – Схема для измерения частоты единичного усиления и скорости нарастания выходного напряжения.

Таблица – Результаты исследования АЧХ характеристики ОУ

	$K=10$	$K=100$	$K=1000$
$U_{вх}$, В ($f=50...100$ Гц)			
$U_{вых}$, В ($f=50...100$ Гц)			
$K_{НЧ}=U_{вых}/U_{вх}$			
$U_{вых} \approx U_{вх}$, В ($K \approx 1$)			
f_1 , МГц ($K \approx 1$)			

Определить реальный коэффициент передачи схемы в области низких частот $K_{НЧ}=U_{вых}/U_{вх}$. Результат записать в таблицу 1. Необходимо учитывать, что у функционального генератора задается амплитуда напряжения, а мультиметр измеряет действующее значение.

Увеличивать частоту входного напряжения до тех пор, пока коэффициент передачи не станет равен единице. Частоту единичного усиления можно искать, ориентируясь на справочные данные. Записать значение этой частоты и выходное напряжение в таблицу 1.

Провести измерения еще для двух коэффициентов усиления: 100 и 1000.

По результатам построить асимптотические ЛАЧХ усилителей в одной системе координат. При построении считать, что в области низких частот ЛАЧХ имеет горизонтальный участок на уровне $K_{НЧ}$. Наклонный участок провести через точку f_1 с наклоном -20 дБ/дек.

Измерение скорости нарастания выходного напряжения.

Изменить форму напряжения функционального генератора на прямоугольную. Амплитуду напряжения установить равную 10 В, частоту примерно равную $f_1/100$, коэффициент усиления — единичный.

Увеличивая частоту, добиться того, чтобы форма выходного напряжения стала треугольной или трапецидальной с пологими фронтами. Определить скорость нарастания по осциллографу на линейном участке выходного сигнала как отношение приращения напряжения к приращению времени. Записать условия эксперимента $U_{вх}$, R_1 , R_2 , f , и результаты измерений: ΔU (В), Δt (мкс), $V=\Delta U/\Delta t$ (В/мкс).

Снять зависимость коэффициента нелинейных искажений (КНИ) от частоты.

Схема эксперимента та же. Рекомендуемый коэффициент передачи усилителя 1...10, амплитуда входного напряжения 1...10 В. Во время эксперимента форма сигнала должна быть синусоидальной.

Установить частоту сигнала $f \approx V/(4\pi A)$, где V — скорость нарастания выходного напряжения ОУ; A — амплитуда выходного напряжения. Установить частоту первой гармоники (Fundamental Frequency) измерителя НИ равной частоте входного сигнала. Записать КНИ в таблицу 3.2.

Таблица 2 – Зависимость КНИ от частоты

f , кГц						
K_G , %						

3.3. Постепенно изменяя частоту примерно до $V/(\pi A)$, снять зависимость КНИ от частоты. При каждом измерении останавливать измеритель НИ кнопкой Stop, устанавливать частоту первой гармоники, равную частоте сигнала, запускать измеритель НИ, записывать результат в таблицу.

3.4. Начертить график зависимости КНИ от частоты.

3.5. Повторить измерения по пунктам 1–3 для второго типа ОУ.

Контрольные вопросы

1. К какой классификационной группе относятся исследованные вами операционные усилители?

2. В одной системе координат постройте семейство ЛАЧХ усилителей с заданными коэффициентами передачи (например, $K=10, 100, 1000$) и ЛАЧХ разомкнутого усилителя с коэффициентом усиления K_{OU} .

3. Какие из параметров ОУ особенно важны при обработке постоянного, переменного, импульсного напряжений?

4. Какие из параметров ОУ влияют на аддитивную, а какие на мультипликативную составляющие погрешности усилительного устройства?

5. Какие из параметров ОУ определяют основную, а какие дополнительную погрешности?

6. Что называется коэффициентом нелинейных искажений?

Практическая работа №39. Расчет режимов работы транзисторного ключа.

Цель работы: изучение и исследование основных схемотехнических принципов построения логических элементов цифровых устройств преобразования сигналов. Определить продолжительность стадии переключения транзисторного ключа, в соответствии с заданием.

Теоретическая часть

Резисторно-транзисторная логика (РТЛ) — технология построения логических электронных схем на базе простых транзисторных ключей. На начальной стадии развития техники цифровых ИС прежде всего использовались схемы, которые легко можно было реализовать и свойства которых уже были известны (на основе аналогии с обычными схемами, собранными из дискретных компонентов). Простейшими из этих схем являются схемы резисторно-транзисторной логики (РТЛ), которые под разными названиями выпускались несколькими фирмами. На рис. 1 показана базовая РТЛ-схема, в которой транзистор находится в режиме насыщения, если на одном или нескольких входах имеется напряжение высокого уровня (Н). При этом выходное напряжение схемы равно 0 В (L). Таким образом, в рамках положительной логики эта РТЛ-схема является схемой ИЛИ-НЕ с функцией $F = \overline{A \cup B \cup C}$. Выходное напряжение схемы имеет высокий уровень (Н), если все три входных напряжения равны нулю (L).

C	B	A	F
L	L	L	H
L	L	H	L
L	H	L	L
L	H	H	L
H	L	L	L
H	L	H	L
H	L	H	L
H	H	H	L

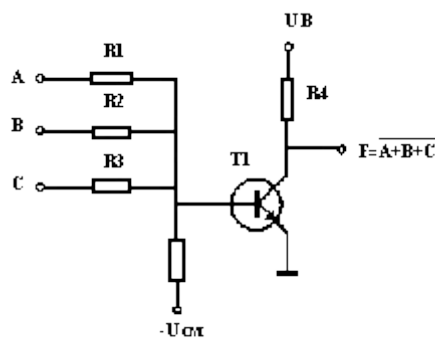


Рисунок 1. Базовая РТЛ-схема с таблицей истинности.

В таблице применяются обозначения МЭК для уровней напряжения.

В режиме с низким уровнем на входе (0 В, отрицательная логика), схема ведет себя как схема И-НЕ с функцией $F = \overline{ABC}$.

В ключевом каскаде транзистор обычно включается по схеме с общим эмиттером. Ключевой каскад (рис.3) может находиться в одном из двух стационарных состояний.

В соответствии с функциями ключа, транзистор может находиться в одном из двух статических режимов: режиме отсечки и режиме насыщения. Активный режим работы обусловлен переходом из одного статического режима в другой.

Режим отсечки (транзистор закрыт). На входе действует напряжение $U_{вх} = U^0$ В этом режиме ток коллектора $I_K = I_{КБО}$, где $I_{КБО}$ — обратный ток коллекторного перехода. Напряжение на выходе ключа практически равно напряжению источника питания. Рабочая точка находится в точке А на нагрузочной прямой. Для обеспечения такого режима необходимо выполнить условие $U_{п}U_{Б}$, $U_{Б} = U_0 + I_{КБО} \cdot R$, и условие отсечки определяется: $U_{п} U_0 + I_{КБО} \cdot R$.

Активный режим (транзистор открыт, но не насыщен). Напряжение на входе лежит в пределах $U_{п} U_{вх} U^1$. В этом режиме транзистор находится короткое время, равное времени переключения из одного статического состояния в другое. Через электроды транзистора протекают прямые токи $I_Э$, I_K , $I_Б$. При изменении $U_{вх}$ меняется ток базы $I_Б$ и рабочая точка

перемещается по нагрузочной прямой от точки А к Б. В активном режиме $I_{КБ0}$ I_K $I_{Кнас}$. И напряжение на базе U_B $U_{п}$.

Режим насыщения (транзистор открыт и насыщен). В режиме насыщения на входе действует напряжение $U_{вх} = U_1$, которое вызывает появление тока, втекающего в базу I_B $I_{Бнас}$. $I_{Бнас}$ соответствует границе между активным режимом и режимом насыщения (точка Б). В этой точке ток базы ещё связан линейной зависимостью с током коллектора, т.е. $I_{Бнас} = I_{Кнас} / h_{21Э}$.

В режиме насыщения ток коллектора $I_{Кнас}$ $I_B \cdot h_{21Э}$. Тогда условие насыщения транзистора $U_1 / R_{Eк} / h_{21Э} \cdot R_k$.

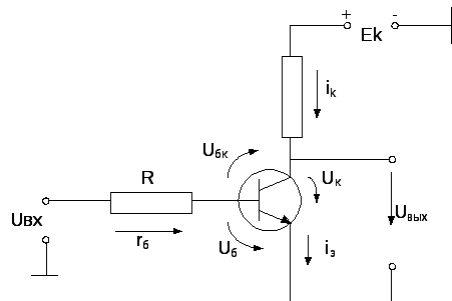


Рисунок 2. Схема транзисторного ключа.

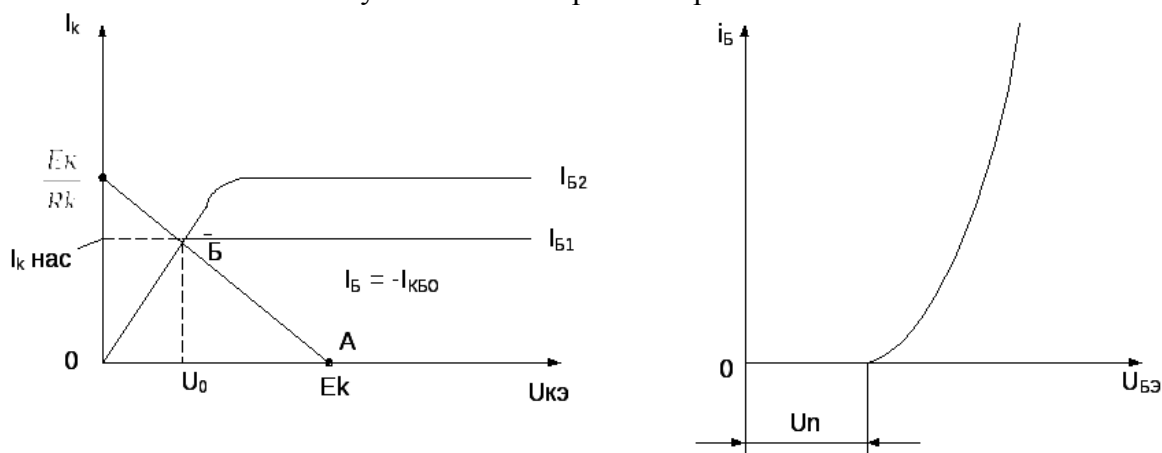


Рисунок 3. Входная и выходная характеристики транзисторного ключа.

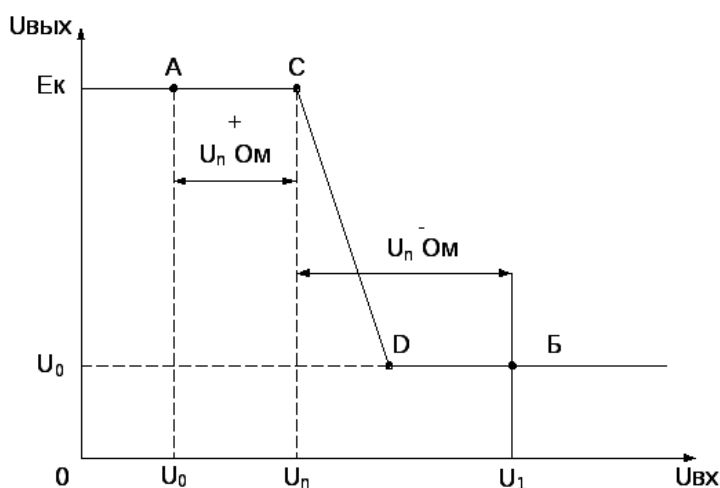


Рисунок 4. Передаточная характеристика транзисторного ключа.

Практическая часть

Задача: определить продолжительность стадий переключения транзисторного ключа, если $E_k = 5 \text{ В}$, $R_k = 1 \text{ кОм}$, $|U_m| = 1 \text{ В}$, $\tau_B = 100 \text{ нс}$, $\beta = 50$, $C_k = 5 \text{ пФ}$, $C_э = 5 \text{ пФ}$, $U_{пор} = 0$.

Пример выполнения:

1. Задержка включения транзистора при $U_{пор} = 0$

$$t_{з.вкл} = 0,7(C_k + C_э) \cdot R_б = 0,7 \cdot 10 \cdot 10^{-12} \cdot 2 \cdot 10^3 = 14 \text{ нс}$$

2. Для определения длительности фронта включения находим эквивалентную постоянную транзистора

$$\tau_{\text{ЭК}} = \tau_{\beta} + R_k \cdot C_k (\beta + 1) = 100 \cdot 10^{-9} + 1 \cdot 10^3 \cdot 5 \cdot 10^{-12} \cdot 51 = 355 \text{ нс}$$

и степень насыщения

$$S = I_{\bar{6}} \cdot \beta / I_{\text{КН}}; \quad I_{\text{КН}} = E_k / R_k = 5 / 1 \cdot 10^3 = 5 \text{ мА};$$

$$I_{\bar{6}}^+ = I_{\bar{6}}^- = |U_m| / R_{\bar{6}} = 1 / (2 \cdot 10^3) = 0,5 \text{ мА};$$

$$S = (0,5 \cdot 10^{-3} \cdot 50) / (5 \cdot 10^{-3}) = 5.$$

Так как $S \gg 1$, то величину $t_{\text{ф.вкл}}$ можно определить по выражению

$$t_{\text{ф.вкл}} = \tau_{\text{ЭК}} / S = (355 \cdot 10^{-9}) / 5 = 71 \text{ нс}.$$

Общее время включения транзистора

$$t_{\text{вкл}} = t_{\text{з.вкл}} + t_{\text{ф.вкл}} = 14 \cdot 10^{-9} + 71 \cdot 10^{-9} = 85 \text{ нс}.$$

3. Задержка включения в соответствии с выражением

$$t_{\text{з.вык}} = 2,3 \cdot \tau_{\text{ЭК}} \lg[\beta(I_{\bar{6}} + I_{\text{КН}})] / \beta \cdot I_{\bar{6}} + I_{\text{КН}} =$$

$$= 2,3 \cdot 355 \cdot 10^{-9} \lg[50(0,5 \cdot 10^{-3} + 0,5 \cdot 10^{-3})] / 50 \cdot 0,5 \cdot 10^{-3} + 5 \cdot 10^{-3} \approx 180 \text{ нс}$$

4. Длительность фронта включения

$$t_{\text{ф.вык}} = 2,3 \cdot \tau_{\text{ЭК}} \lg(I_{\text{КН}} + \beta I_{\bar{6}}) / \beta \cdot I_{\bar{6}} =$$

$$= 2,3 \cdot 355 \cdot 10^{-9} \lg(0,5 \cdot 10^{-3} + 50 \cdot 0,5 \cdot 10^{-3}) / 50 \cdot 0,5 \cdot 10^{-3} = 49 \text{ нс}$$

Общее время выключения транзистора

$$t_{\text{вык}} = t_{\text{з.вык}} + t_{\text{ф.вык}} = 180 \cdot 10^{-9} + 49 \cdot 10^{-9} = 229 \text{ нс}$$

Из приведенного примера следует, что задержка выключения, связанная с насыщением транзистора, является самой длительной стадией переходных процессов.

Исходные данные для выполнения практической работы

Вариант	R_k , кОм	$R_{\bar{6}}$, кОм	U_m , В	τ_{β} , нс	β	C_k , пФ	$C_{\bar{6}}$, пФ	$U_{\text{пор}}$
1	2	3	1	100	50	5	5	0
2	1	2	1	90	50	5	6	0
3	1	1	1	80	60	6	5	0
4	1	4	1	70	50	4	5	0
5	3	2	1	75	50	3	4	0
6	2	3	1	100	60	5	3	0
7	1	3	1	60	70	5	5	0
8	2	1	1	95	55	6	4	0
9	3	5	1	100	60	4	6	0
10	1	1	1	100	50	5	6	0
11	2	2	1	100	60	6	6	0
12	3	3	1	80	50	5	4	0
13	1	2	1	100	55	5	6	0
14	2	2	1	85	50	4	5	0
15	2	3	1	85	65	5	5	0
16	1	4	1	90	50	5	6	0
17	3	2	1	100	50	6	6	0
18	2	2	1	90	55	5	4	0
19	3	2	1	80	50	4	4	0
20	2	4	1	100	70	5	5	0
21	3	2	1	85	50	5	5	0
22	3	3	1	90	50	5	5	0
23	2	1	1	100	65	4	5	0
24	3	3	1	95	55	6	4	0
25	4	4	1	100	50	6	4	0

26	3	3	1	100	55	5	7	0
27	5	5	1	75	65	6	4	0
28	3	3	1	100	50	5	5	0
29	4	3	1	80	65	4	6	0
30	2	2	1	80	50	5	6	0

Содержание отчета

1. Расчет транзисторного ключа
2. Вывод по сделанным расчетам
3. Схема транзисторного ключа

Контрольные вопросы

1. Назначение транзисторных ключей.
2. Режимы работы транзисторного ключа.
3. Состояние транзистора при режиме отсечки.
4. Состояние транзистора при режиме насыщения.
5. Что такое активный режим.
6. Какое напряжение на выходе соответствует режиму отсечки.
7. Какое напряжение на входе соответствует режиму насыщения.
8. Какое напряжение на входе соответствует режиму отсечки.
9. Какое напряжение на выходе соответствует режиму насыщения.
10. Нарисовать входную характеристику транзисторного ключа, объяснить особенности входной характеристики.
11. Нарисовать передаточную характеристику, объяснить по ней режимы работы транзисторного ключа.
12. Нарисовать выходную характеристику транзисторного ключа, объяснить по ней режим насыщения.

Практическая работа №40. Исследование логического элемента ТТЛ.

Цель работы: Приобретение практических навыков работы в программной среде NI Multisim, построение логического элемента ТТЛ, определение уровней выходного сигнала.

Ход работы

Собрать схему согласно рис.1, измерить $U_{вых}^0$ (нулевой выходной сигнал) и $U_{вых}^1$ (логическая единица).

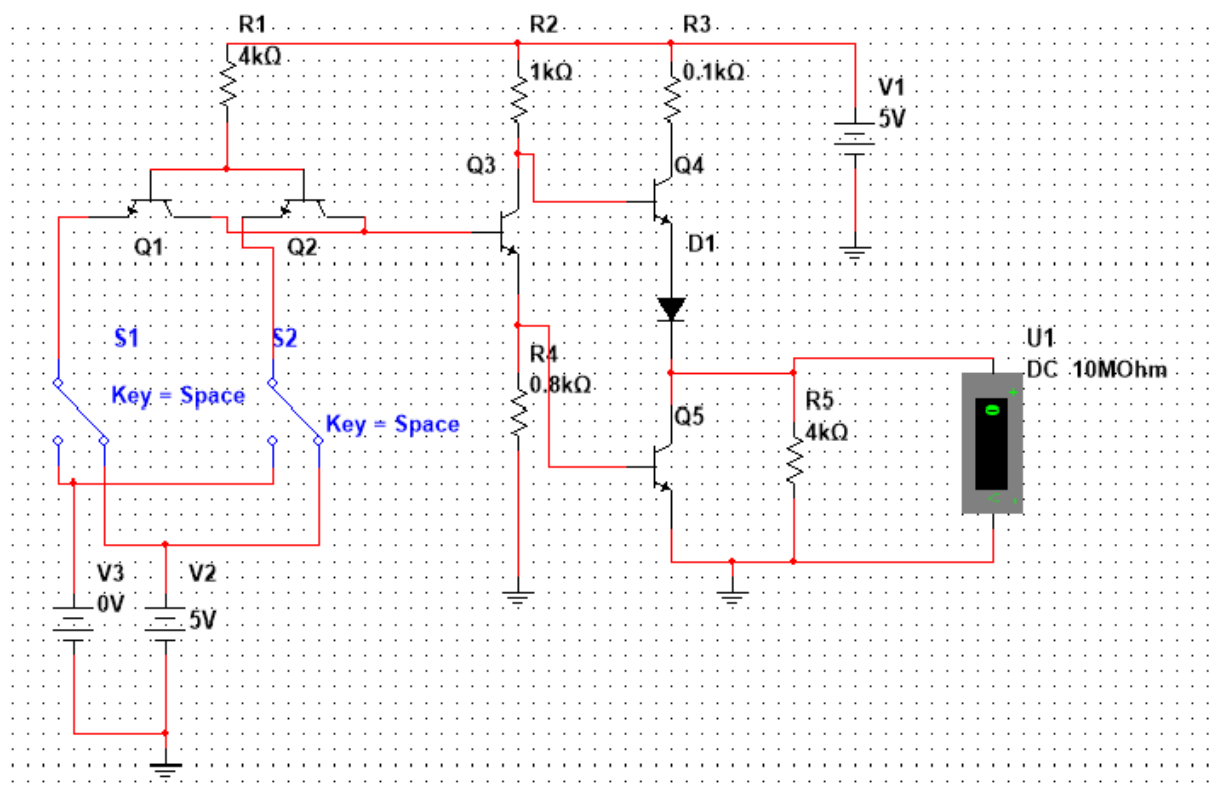


Рис.1 Электрическая принципиальная схема элемента ТТЛ И-НЕ

Оформить отчет и сделать выводы о проделанной работе.

Практическая работа №41. Микросхемы эмиттерно–связанной логики (ЭСЛ).

Цель работы: Изучить схему базового логического элемента ЭСЛ, принцип работы, основные параметры и характеристики, стандартные серии и номенклатуру микросхем ЭСЛ.

Ход работы

Задание 1. Зарисовать схему базового логического элемента ЭСЛ, пояснить принцип работы.

Задание 2. Записать основные параметры и характеристики микросхем ЭСЛ.

Задание 3. Перечислить стандартные серии микросхем ЭСЛ.

Задание 4.

1. Привести функциональное обозначение заданной микросхемы.
2. Пояснить выполняемую функцию заданной микросхемы.
3. Указать состав микросхемы, наименование и назначение входов и выходов.
4. Характерные особенности применения заданной микросхемы.
5. Указать стандартные серии для данного типа микросхемы.
6. Записать логические выражения для выходов микросхемы.

№ вар	1	2	3	4	5	6	7	8	9	10
Тип ИС	ЛЕ106	ЛЕ111	ЛЕ123	ЛК117	ЛК118	ЛК121	ЛЛ110	ЛМ101	ЛМ102	ЛМ105
№ вар	11	12	13	14	15	16	17	18	19	20
Тип ИС	ЛМ109	ЛП107	ЛП112	ЛК121	ЛЕ106	ЛЕ106	ЛЕ111	ЛЕ123	ЛК117	ЛК118
№ вар	21	22	23	24	25	26	27	28	29	30
Тип ИС	ЛЕ123	ЛК117	ЛК118	ЛК121	ЛЛ110	ЛМ101	ЛЕ106	ЛЕ111	ЛМ102	ЛМ105

Задание 5. Для вариантов 1, 5, 6, 11, 16, 21, 26.

1. Перечислить номенклатурный ряд микросхем типа ЛЕ.
2. Перечислить стандартные серии микросхем типа ЛЕ

3. Привести пример микросхемы типа ЛЕ(функциональное обозначение, состав, назначение входов/выходов).

4. Характерные особенности применения микросхем типа ЛЕ.

Задание 5. Для вариантов 2, 7, 10, 12, 17, 22, 27

1. Перечислить номенклатурный ряд микросхем типа ЛК.

2.Перечислить стандартные серии микросхем типа ЛК.

3. Привести пример микросхемы типа ЛК (функциональное обозначение, состав, назначение входов/выходов).

4. Характерные особенности применения микросхем типа ЛК.

Задание 5. Для вариантов 3, 8, 13, 15, 18, 23, 28.

1. Перечислить номенклатурный ряд микросхем типа ЛМ.

2.Перечислить стандартные серии микросхем типа ЛМ.

3. Привести пример микросхемы типа ЛМ (функциональное обозначение, состав, назначение входов/выходов).

4. Характерные особенности применения микросхем типа ЛМ.

Задание 5. Для вариантов 4, 9, 14, 19, 20, 24, 29, 30.

1. Перечислить номенклатурный ряд микросхем типа ЛП.

2.Перечислить стандартные серии микросхем типа ЛП.

3. Привести пример микросхемы типа ЛП(функциональное обозначение, состав, назначение входов/выходов).

4. Характерные особенности применения микросхем типа ЛП.

Практическая работа №42. Микросхемы на КМОП (КМДП) – транзисторах.

Цель работы: Изучить схему базового логического элемента КМОП, принцип работы, основные параметры и характеристики, стандартные серии и номенклатуру микросхем КМОП.

Ход работы

Задание 1. Зарисовать схему базового логического элемента КМОП (КМДП) структуры, пояснить принцип работы.

Задание 2. Записать основные параметры и характеристики микросхем КМОП (КМДП) структуры.

Задание 3. Перечислить стандартные серии микросхем КМОП (КМДП) структуры.

Задание 4.

1. Привести функциональное обозначение заданной микросхемы.

2. Пояснить выполняемую функцию заданной микросхемы.

3. Указать состав микросхемы, наименование и назначение входов и выходов.

4. Характерные особенности применения заданной микросхемы.

5. Указать стандартные серии для данного типа микросхемы.

6. Записать логические выражения для выходов микросхемы.

№ вар	1	2	3	4	5	6	7	8	9	10
Тип ИС	ЛА7	ЛА8	ЛА9	ЛЕ5	ЛЕ6	ЛЕ10	ЛИ1	ЛИ2	ЛН1	ЛН2
№ вар	11	12	13	14	15	16	17	18	19	20
Тип ИС	ЛП2	ЛП4	ЛП11	ЛП13	ЛА7	ЛА9	ЛА8	ЛА9	ЛЕ5	ЛЕ6
№ вар	21	22	23	24	25	26	27	28	29	30
Тип ИС	ЛЕ10	ЛИ1	ЛИ2	ЛН1	ЛН2	ЛП2	ЛП4	ЛП11	ЛП13	ЛЕ6

Задание 5. Для вариантов 1, 5, 6, 11, 16, 21, 26.

1. Перечислить номенклатурный ряд микросхем типа ЛА.

2. Перечислить стандартные серии микросхем типа ЛА

3. Привести пример микросхемы типа **ЛА**(функциональное обозначение, состав, назначение входов/выходов).

4. Характерные особенности применения микросхем типа **ЛА**.

Задание 5. Для вариантов 2, 7, 10, 12, 17, 22, 27

1. Перечислить номенклатурный ряд микросхем типа **ЛЕ**.

2. Перечислить стандартные серии микросхем типа **ЛЕ**.

3. Привести пример микросхемы типа **ЛЕ** (функциональное обозначение, состав, назначение входов/выходов).

4. Характерные особенности применения микросхем типа **ЛЕ**.

Задание 5. Для вариантов 3, 8, 13, 18, 20, 23, 28.

1. Перечислить номенклатурный ряд микросхем типа **ЛИ**.

2. Перечислить стандартные серии микросхем типа **ЛИ**.

3. Привести пример микросхемы типа **ЛИ** (функциональное обозначение, состав, назначение входов/выходов).

4. Характерные особенности применения микросхем типа **ЛИ**.

Задание 5. Для вариантов 4, 9, 14, 19, 24, 25, 29, 30.

1. Перечислить номенклатурный ряд микросхем типа **ЛП**.

2. Перечислить стандартные серии микросхем типа **ЛП**.

3. Привести пример микросхемы типа **ЛП**(функциональное обозначение, состав, назначение входов/выходов).

4. Характерные особенности применения микросхем типа **ЛП**.

Практическая работа №43. Разработка электронной схемы программируемой логической матрицы.

Цель работы: изучение элемента проектирования – программируемых логических матриц(ПЛМ).

Теоретические сведения

Из формальной логики известно, что любую функцию, заданную таблицей истинности, можно представить в дизъюнктивной нормальной форме – ДНФ (в виде дизъюнкции конъюнкций) и, следовательно, реализовать в схеме, используя вентили И и ИЛИ. Для вычисления логических функций в форме дизъюнкции конъюнкций служит так называемая программируемая логическая матрица (ПЛМ).

Обобщенная структура ПЛМ приведена на рисунке 1.

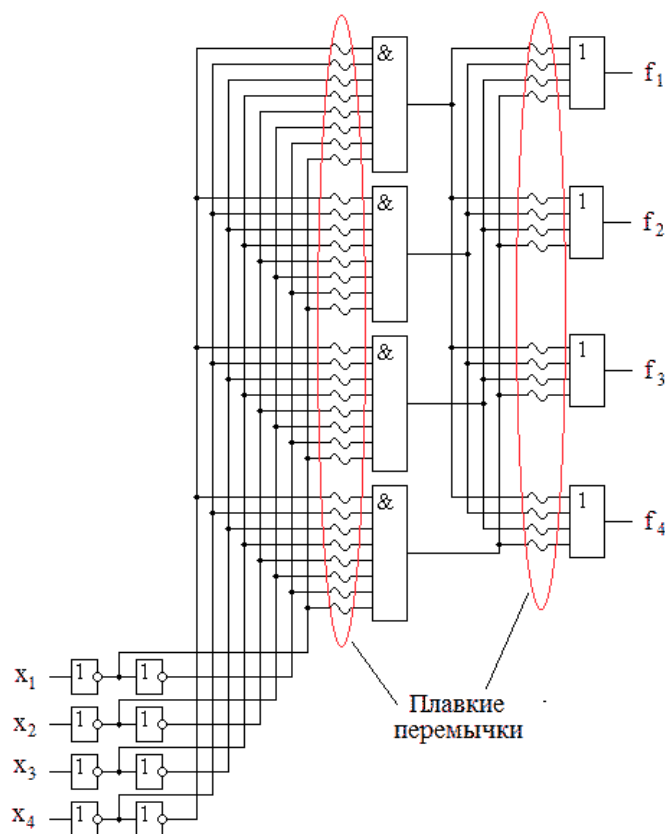


Рисунок 1. Обобщенная структура программируемых логических матриц (ПЛМ)

Основная идея работы ПЛМ заключается в реализации логической функции, представленной в СДНФ — дизъюнктивной нормальной форме. На рисунке четко прослеживаются логические элементы "И", способные реализовать любой минтерм СДНФ и логические элементы "ИЛИ", осуществляющие суммирование термов, требующихся по логическому выражению СДНФ. В схеме ПЛМ, приведенной на рисунке 1, ранг терма ограничен количеством входов и равен четырем, количество термов тоже равно четырем. В реально выпускавшихся микросхемах программируемых логических матриц (ПЛМ) количество входов было равно шестнадцати (максимальный ранг минтерма — 16), количество термов равно 32 и количество выходов микросхемы — 8.

Следует отметить, что полная принципиальная схема ПЛМ получается достаточно громоздкой (см. рисунок 1). Поэтому обычно применяется шинное представление проводников. Логический элемент "И", реализующий минтерм СДНФ, изображается как одиночная горизонтальная строка с условно-графическим обозначением схемы "И". Ко входам этого элемента подводится многоразрядная шина, а на выходе подключен одиночный проводник. Если входной проводник подключается ко входу логического элемента "И" (перемычка сохранена), то это место обозначается крестиком 'x', а если соединение отсутствует (перемычка сожжена), то крестик не проставляется. Аналогично обозначаются и многовходовые элементы "ИЛИ". Пример подобного изображения схемы ПЛМ (PLA) приведен на рисунке 2.

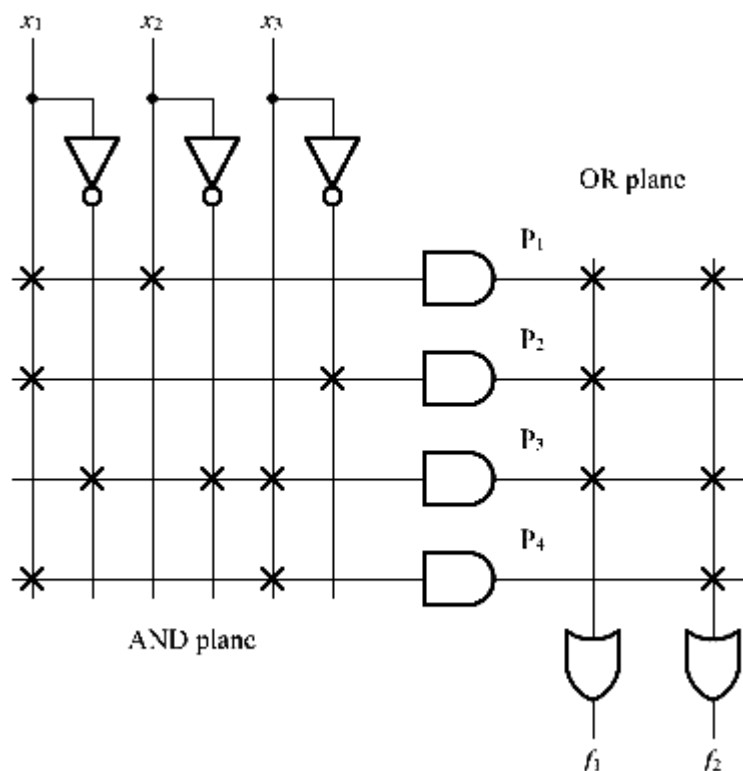


Рисунок 2. Представление внутренней структуры схем ПЛМ.

По схеме четко можно восстановить реализуемую им логическую функцию. На рисунке 2 реализованы две логические функции f_1 и f_2 :

$$f_1 = x_1x_2 + x_1\bar{x}_3 + \bar{x}_1\bar{x}_2x_3$$

$$f_2 = x_1x_2 + \bar{x}_1\bar{x}_2x_3 + x_1x_3$$

Практическая часть

Задание. Разработайте электронную логическую схему программируемой логической матрицы, реализующую логические функции, заданные в таблице соответствия:

X ₁	X ₂	X ₃	f ₁	f ₂	f ₃	f ₄	f ₅	f ₆
0	0	0	0	1	1	0	1	0
0	0	1	1	1	0	0	1	1
0	1	0	1	0	1	1	0	1
0	1	1	0	0	0	1	0	1
1	0	0	0	1	0	1	0	1
1	0	1	1	1	1	0	1	0
1	1	0	1	0	1	1	1	0
1	1	1	1	0	0	0	1	0

Содержание отчета

Отчет должен содержать:

1. Цель работы.
2. Словесную постановку задачи создания электронной логической схемы программируемой логической матрицы.
3. Электронную логическую схему программируемой логической матрицы.

Цель работы: изучение элемента проектирования – программируемых логических матриц (ПЛМ).

Теоретические положения и пример решения задачи.

Программируемые логические матрицы (ПЛМ) – большие интегральные схемы, современная база проектирования схем. ПЛМ характеризуются параметрами: n , q , m , где n – число входных переменных, m – число выходных переменных, q – число конъюнкций, из которых строятся булевы функции, реализуемые ПЛМ. Различаются следующие уровни ПЛМ:

- Средняя степень интеграции, $n+m < 14$, $q = 102$;
- Большой уровень интеграции, $n+m < 42$, $q=104$;
- Сверхбольшой уровень интеграции, ПЛМ находится внутри схемы и $n+m < 102$, $q=105$ –

106.

ПЛМ реализуют схемы с памятью, но мы ограничимся схемами комбинационного типа, и более того будем рассматривать ПЛМ с параметрами $n=4$, $m=2$ и $q=10$.

Общая схема ПЛМ имеет следующий вид:

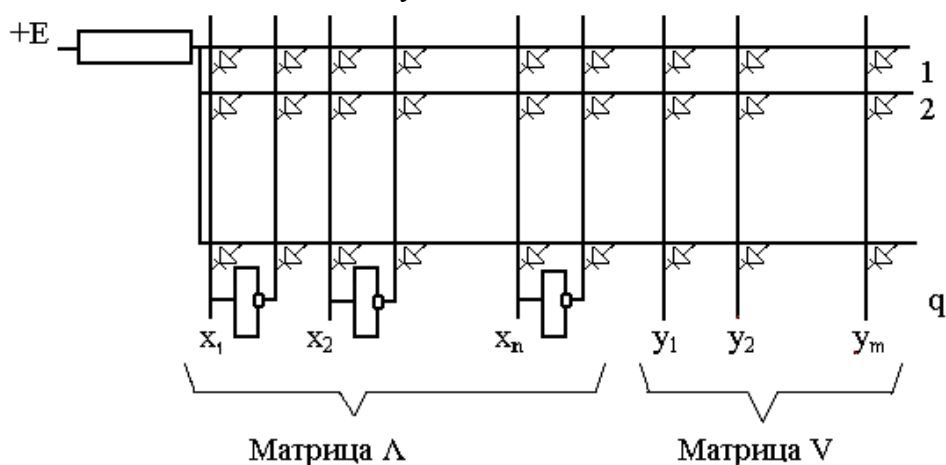


Рисунок 1. ПЛМ на диодах.

На каждом пересечении горизонтальной и вертикальной линий стоит цепочка последовательно соединенных диодного элемента и плавкого предохранителя. Такое соединение мы будем обозначать жирной точкой:



Рисунок 2. Обозначение соединения диодом.

Тогда наши функции будут реализованы следующей ПЛМ:

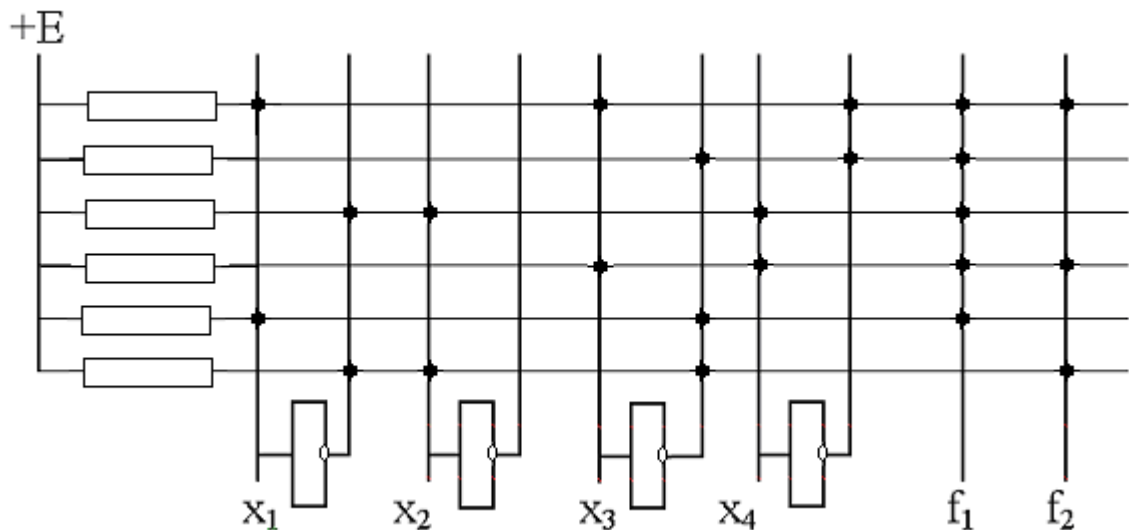


Рисунок 3. Реализация функций f_1 и f_2 на одной ПЛМ.

Теперь займемся ограничениями. Рассмотрим самое интересное из них: $n_{ПЛМ} < n_{бф}$. Т. е. число переменных на входе ПЛМ меньше, чем число переменных булевой функции.

Для нашего примера $n_{бф}=4$, пусть $n_{ПЛМ}=3$. Рассматриваем первую конъюнкцию u_1 . Ограничение сразу помогает выбрать первый интервал в первую ПЛМ. Задействованы сразу все переменные ПЛМ. Число интервалов не ограничено, поэтому добавляем в первую ПЛМ еще интервалы u_2 , u_4 и u_5 , зависящие от тех же переменных. Следующий интервал u_3 , который задействует тоже сразу все три переменных ПЛМ2. И, наконец ПЛМ3 реализует оставшийся интервал u_6 .

Внешняя схема соединения ПЛМ выглядит следующим образом:

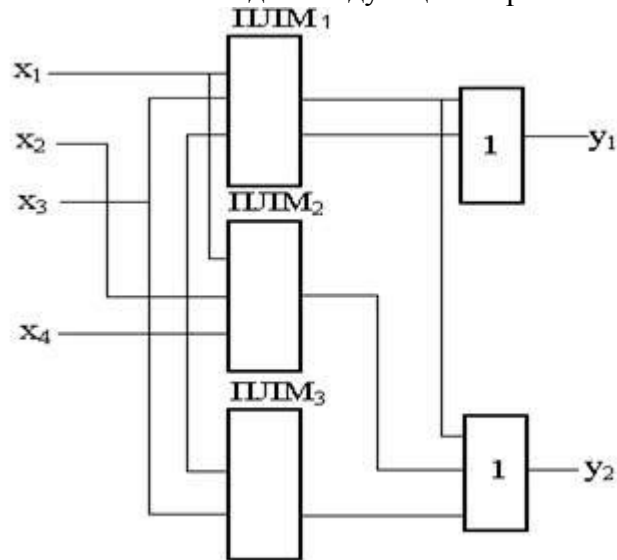


Рисунок 4. Схема двух булевых функций с учетом ограничения на число входов ПЛМ.

Реализуем схему на ПЛМ, немного изменив булевы функции.

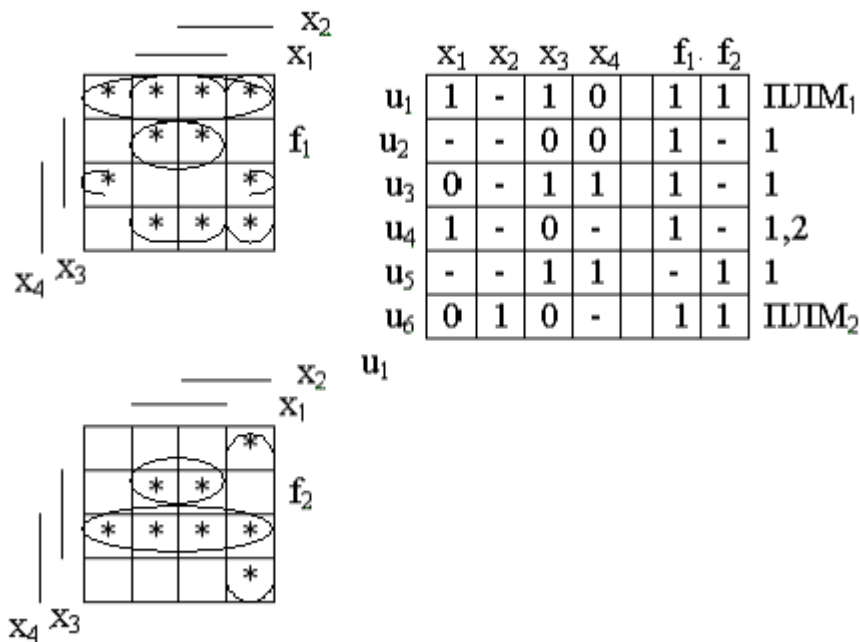


Рисунок 5. Система двух булевых функций, реализация на ПЛМ с учетом ограничения на число входов ПЛМ.

Задание на практическую работу

Построить схему на одной ПЛМ. Функции f_{i+1} , f_{i+2} реализовать схемой на ПЛМ с учетом ограничения при $n_{ПЛМ}=3$.

Практическое занятие №45. Запоминающие устройства

Цель работы: изучить временные диаграммы работы запоминающих устройств, построенных по различным схемам.

Порядок выполнения работы

1. Изучите до начала выполнения лабораторной работы методики построения оперативных и постоянных запоминающих устройств.

2. Изучите методику создания при помощи утилиты MegaWizard Plug-in Manager компонентов и их использования в САПР Quartus II.

3. Соберите схему для исследования постоянного запоминающего устройства в САПР Quartus II согласно варианта задания.

Для создания постоянного запоминающего устройства в САПР Quartus II при помощи утилиты MegaWizard Plug-in Manager следует выполнить такие операции:

- создайте новый проект;
- выберите в качестве элементной базы для реализации любую из микросхем ПЛИС, обязательно принадлежащих только семейству Cyclone (например, EP1C3T100A8);
- создайте новый Memory Initialization File (выберите необходимый пункт из меню File -> New);
- задайте необходимые значения для запрашиваемых параметров (например: Word size: 8, Number of words: 32);
- заполните открывшийся файл необходимыми данными, которые будут содержаться в ячейках ПЗУ (в качестве примера приведено заполнение файла LpmRom.mif);
- сохраните заполненный файл инициализации памяти (рекомендуется располагать его в каталоге проекта) и закройте его;
- создайте новый Block Diagram/Schematic File (выберите необходимый пункт из меню File -> New);

- выберите пункт Insert Symbol из меню Edit (аналогичное действие производится по двойному нажатию мышки в любом пустом месте Block Diagram/Schematic File);
- найдите и выберите из раскрывающегося списка в левой части появившегося окна параметризованную мегафункцию lpm_rom, которая находится в разделе megafunctions -> storage стандартных библиотек САПР Quartus II);
- завершите подтверждение выбора нажатием кнопки OK;

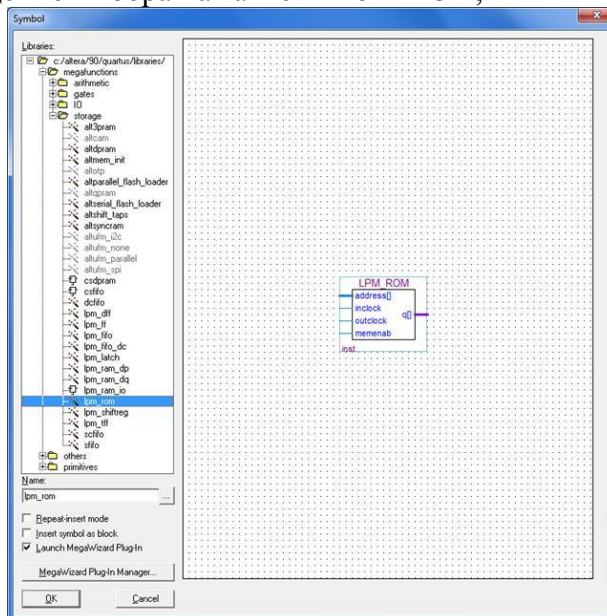


Рисунок 1. Выбор параметризованной мегафункции lpm_rom

- задайте имя создаваемого компонента, соблюдая правила именования, в открывшемся окне утилиты MegaWizard Plug-in Manager при необходимости и в случае, если САПР автоматически не выполнила эту операцию (например, не задавала имя по умолчанию lpm_rom0);
- установите необходимое значение в списке выбора языка описания компонента (настоятельно рекомендуется использовать язык VHDL);
- нажмите кнопку Next >;

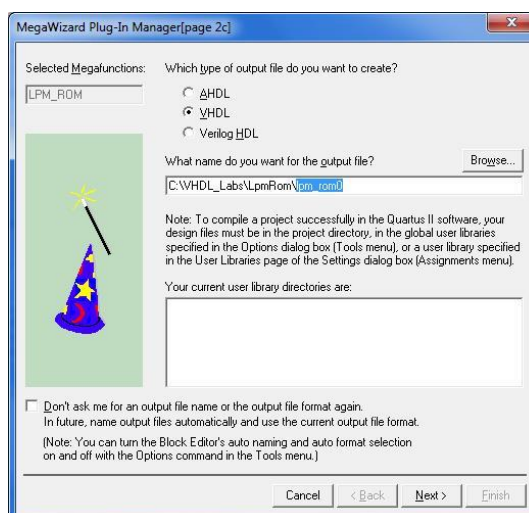


Рисунок 2. Первый шаг создания ПЗУ

- установите необходимое значение в списке выбора разрядности выходной шины данных ПЗУ (в качестве примера было выбрано значение 8 бит);
- установите необходимое значение в списке выбора общего объема памяти, которое задаётся как количество N-битных слов, где N – разрядность выходной шины данных ПЗУ (в качестве примера было выбрано значение 32);
- выполните проверку соответствия между заданными значениями разрядности и объема памяти, которые указывались при создании Memory Initialization File, и теми, которые установлены в окне утилиты MegaWizard Plug-in Manager;

– внесите необходимые изменения содержимого Memory Initialization File, в случае возникновения указанных выше несоответствий параметров, поскольку при этом не гарантируется корректная работа ПЗУ (причём, САПР Quartus II не генерирует какие-либо сообщения об ошибках или предупреждения);

– установите значение Auto в списке выбора типа блока памяти (в этом случае компилятор и трассировщик самостоятельно учтут все особенности внутренней реализации ПЗУ для выбранной микросхемы ПЛИС);

– установите необходимое значение в списке выбора способа тактирования операций с памятью: Single clock – общий тактовый сигнал, по которому осуществляется защёлкивание входного адреса ячейки ПЗУ и чтение данных из ячейки с заданным адресом; input, output clocks – отдельные сигналы для защёлкивания адреса ячейки и чтения данных соответственно (в качестве примера был выбран второй способ тактирования операций);

– выполните повторную проверку соответствия всех заданных параметров;

– нажмите кнопку Next >;

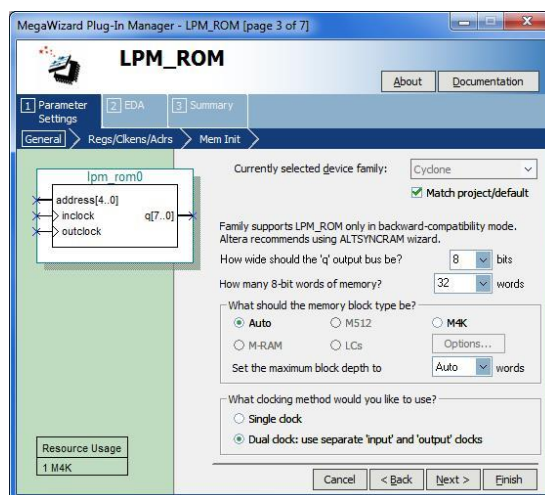


Рисунок 3. Второй шаг создания ПЗУ

– установите соответствующее значение в списке выбора вариантов установки триггеров-защёлок для выходных портов сигналов (создаваемое в качестве примера ПЗУ снабжено соответствующими триггерами), для входных портов значение данной опции задано по умолчанию, его изменять нельзя, поэтому все входные сигналы имеют соответствующие триггеры-защёлки;

– нажмите кнопку Next >;

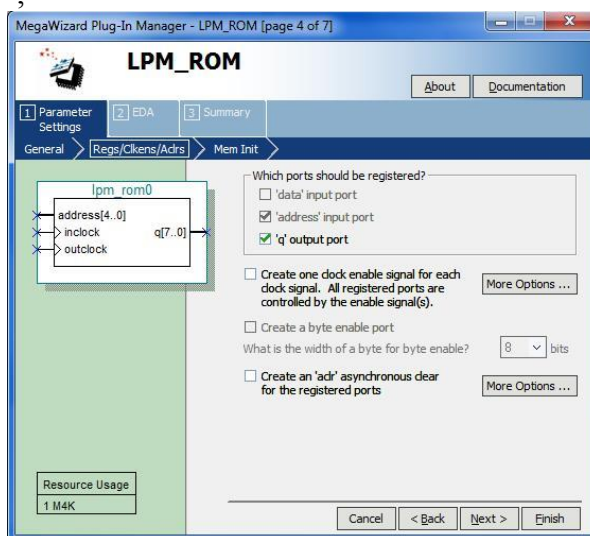


Рисунок 4. Третий шаг создания ПЗУ

– укажите в поле File name: путь, по которому размещён созданный ранее Memory Initialization File (./LpmRom.mif означает, что этот файл находится в каталоге проекта);

– выберите из соответствующего списка расширение файла *.mif (а не *.hex!) в процессе поиска пути к Memory Initialization File;

- нажмите кнопку Next >;
- нажмите кнопку Next > внизу появившегося окна (не следует вносить сюда никаких изменений);
- нажмите кнопку Finish, после появления окна, для завершения процесса создания компонента ПЗУ с помощью утилиты MegaWizard Plug-in Manager в САПР Quartus II.

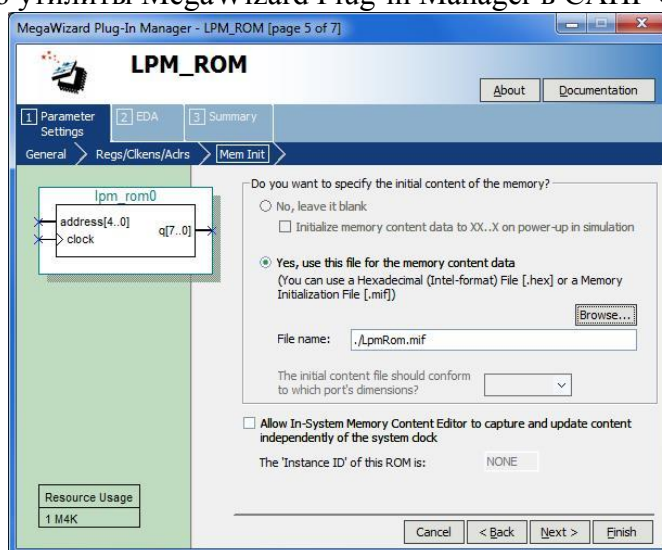


Рисунок 5. Четвёртый шаг создания ПЗУ

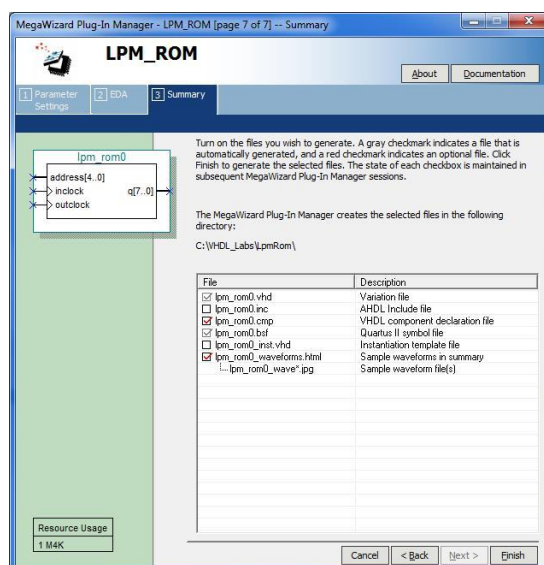


Рисунок 6. Итоговое окно создания ПЗУ

На рисунке 7 в качестве примера приведена схема для исследования синтезированного при помощи утилиты MegaWizard Plug-in Manager постоянного запоминающего устройства на базе параметризируемой мегафункции lpm_rom.

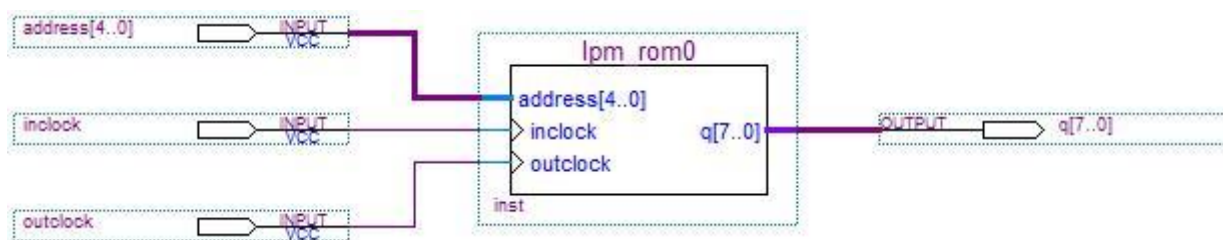


Рисунок 7. Схема исследования созданного ПЗУ

4. Исследуйте временные диаграммы работы синтезированного ПЗУ при произвольных фазовых соотношениях входных сигналов.

На рисунке 8 в качестве примера приведены временные диаграммы работы ПЗУ, процесс создания которого был описан выше.

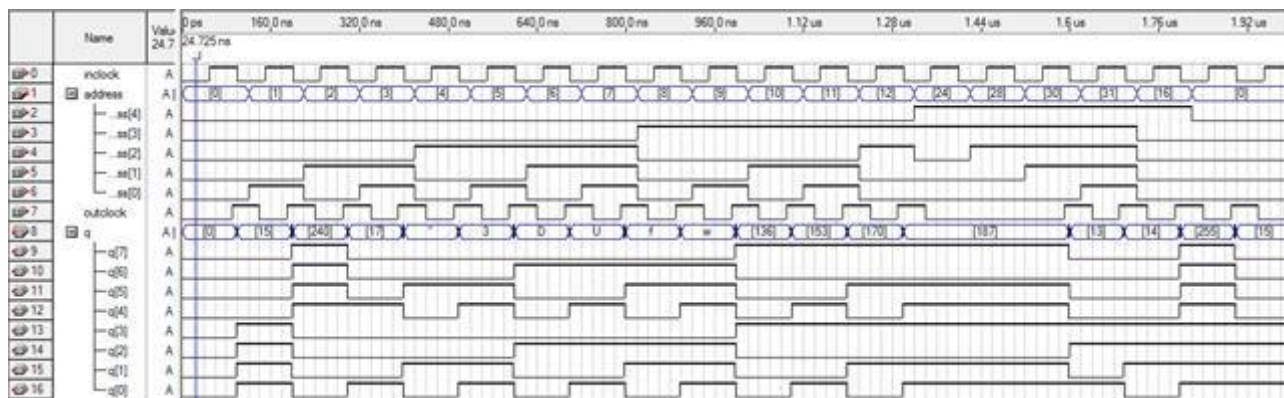


Рисунок 8. Временные диаграммы работы синтезированного ПЗУ

У компонента ПЗУ, созданного в качестве примера, выходная шина данных имеет соответствующие триггеры-защёлки. Это означает, что чтение данных из ячейки ПЗУ с заданным адресом будет выполняться за 2 такта (при отсутствии выходных триггеров чтение выбранной ячейки памяти происходит за 1 такт): в первом такте происходит выборка данных из ячейки ПЗУ, а во втором – эти данные защёлкиваются выходными триггерами. Сказанное выше касается так же и ОЗУ. Однако запись данных в ОЗУ происходит всегда за 1 такт соответствующего сигнала и не зависит от наличия регистров защёлки входных сигналов.

Из анализа временных диаграмм можно сделать вывод, что по каждому переднему фронту импульса на входе inclock происходит защёлкивание адреса ячейки памяти, который установлен на шине address[4..0]. По каждому переднему фронту импульса на входе outclock происходит защёлкивание данных из соответствующей ячейки (значения ячеек памяти заданы при помощи LpmRom.mif) выходными триггерами-защёлками, т.е., по сути, происходит чтение данных. Поэтому при отсутствии импульсов на входе outclock данные на выходе ПЗУ не изменяются (даже при наличии сигналов защёлки адреса).

5. Соберите схему для исследования однопортового оперативного запоминающего устройства на основании параметризируемой мегафункции lpm_ram_dp.

6. Исследуйте временные диаграммы работы синтезированного ОЗУ при произвольных фазовых соотношениях входных сигналов.

7. Соберите схему для исследования оперативного запоминающего устройства на основании параметризируемой мегафункции lpm_ram_dq.

8. Исследуйте временные диаграммы работы синтезированного ОЗУ при произвольных фазовых соотношениях входных сигналов.

Контрольные вопросы

1. Дайте определение запоминающего устройства.
2. Какие виды запоминающих устройств Вы знаете?
3. Дайте определение постоянного запоминающего устройства. Приведите примеры их применения в технике.
4. Дайте определение оперативного запоминающего устройства. Приведите примеры их применения в технике.

Практическое занятие №46. Исследование режимов работы статического ОЗУ.

Цель: Изучение назначения и функций оперативного запоминающего устройства. Знакомство с принципом работы оперативного запоминающего устройства.

Теоретическая часть

По выполняемым функциям различают следующие типы запоминающих устройств:
Оперативное запоминающее устройство (ОЗУ).

Постоянное запоминающее устройство (ПЗУ).

Перепрограммируемое запоминающее устройство (ППЗУ).

ОЗУ используется при необходимости выбрать и обновить хранимую информацию в высоком темпе работы процессора цифрового устройства. В ОЗУ предусматриваются 3 режима работы:

1 – хранение при отсутствии обращения к ЗУ.

2 – режим чтения хранимых слов.

3 – режим записи (время чтения и записи слова в ОЗУ составляют доли миллисекунды).

ОЗУ используются для хранения исходных данных промежуточных, промежуточных и конечных результатов обработки данных. Представляет из себя набор регистров.

ОЗУ строятся из набора однотипных микросхем и являются неотъемлемой частью микропроцессорных систем различного назначения, они делятся на 2 класса: статические и динамические. В статических ОЗУ запоминание информации производится на триггерах, а в динамических – на конденсаторах емкостью 0,5 пФ. Конструктивно ОЗУ выполнено в виде двух блоков – матрицы запоминающих элементов и дешифратора адреса. Матрица состоит из элементов памяти, расположенных вдоль строк и столбцов, и имеет двухкоординатную дешифрацию адреса – по строкам и столбцам. При этом в выбранной ячейке памяти срабатывает двухходовой элемент И и подготавливает цепи чтения записи информации на входных или выходных разрядных шинах.

Оперативные запоминающие устройства (ОЗУ) являются неотъемлемой частью микропроцессорных систем различного назначения. ОЗУ делятся на два класса: статические и динамические. В статических ОЗУ запоминание информации производится на триггерах, а в динамических – на конденсаторах емкостью 0,5 пФ. Длительность хранения информации в статических ОЗУ не ограничена, тогда как в динамических ОЗУ она ограничена временем саморазряда конденсатора, что требует специальных средств регенерации и дополнительных затрат времени на этот процесс.

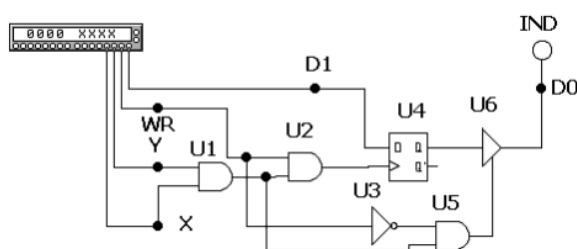


Рисунок 1.

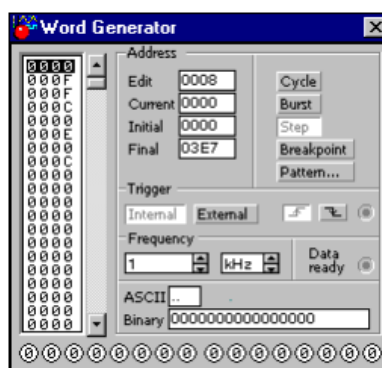


Рисунок 2.

На рис. 1 показана ячейка статического ОЗУ на D-триггере и вспомогательных логических элементах. Информационный вход ячейки подключен к шине данных D1 одного из разрядов, ее выход – к соответствующей шине D0 через элемент с тремя состояниями U6. Ячейка выбирается сигналами Y=1, X=1, поступающими с дешифратора адреса. При записи в ячейку памяти на D1 устанавливается 1 или 0, на входе WR/RD' – сигнал 1, в результате чего срабатывают элементы 2И U1, U2. Положительный перепад сигнала с элемента U2 поступает на тактовый вход D-триггера U4 и в нем записывается 1 или 0 в зависимости от уровня сигнала на его D-входе. При чтении на входе WR/RD' устанавливается 0, при этом срабатывают элементы U1, U3, U5 и на вход

РАЗРЕШЕНИЕ ВЫХОДА буферного элемента U6 поступает разрешающий сигнал, в результате чего сигнал с Q-выхода D-триггера передается на разрядную шину D0, состояние которой индицируется логическим пробником IND. Для проверки функционирования ячейки памяти используется генератор слова (рис. 2), выходной код которого соответствует указанным режимам работы ячейки.

Заметим, что запоминающие устройства статического типа отличаются высоким быстродействием и в компьютерах используются в качестве так называемой кэш-памяти.

Контрольные вопросы и задания

Вопросы:

1. Какие типы памяти существуют?
2. Чем отличается динамическая память от статической?
3. Назовите основные параметры запоминающих устройств.
4. В чем состоит назначение ROM?
5. Какие элементы памяти используются в различных типах ROM?
6. Как и кем программируются PROM?
7. В чем состоит различие EPROM и EEPROM?
8. Какая схемотехника элементов памяти обеспечивает максимальное быстродействие RAM?
9. В чем состоит принципы построения RAM с одномерной и двумерной адресацией?
10. Сравните преимущества и недостатки структур 2D и 3D.
11. Объясните принципы осуществления структуры памяти 2DM и ее основные достоинства по сравнению с остальными структурами.

Задание:

1. Смоделируйте и проанализируйте работу ОЗУ, схема которого приведена выше. Сделайте выводы по работе.

Практическое занятие №47. Изучение структуры и принципов работы постоянного запоминающего устройства (ПЗУ) с неизменяемой информацией.

Цель работы: изучение общих принципов построения ROM; приобретение навыков программирования PROM.

Практические сведения и методические рекомендации

Первое задание практической работы посвящено изучению структуры и принципов работы постоянного запоминающего устройства (ПЗУ) с неизменяемой информацией для отображения шестнадцатеричных символов на 7-сегментном индикаторе (см. рисунок 1.1).

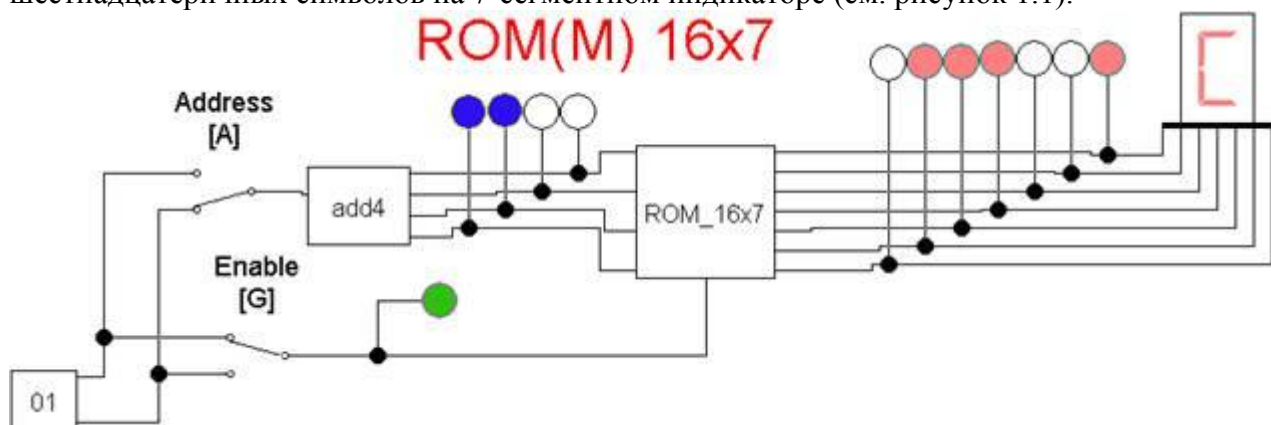


Рисунок 1.1

Блок ПЗУ (подсхема ROM_16x7) состоит из дешифратора на 16 выходов (микросхема 74154) и матрицы памяти с организацией 16x7, оформленных в виде подсхем с соответствующими наименованиями. Разрешение на работу дешифратора обеспечивается положительным уровнем

в исходном состоянии имеются все перемычки в пересечениях шестнадцати горизонтальных выходных линии дешифратора) и семи вертикалей (разрядных линии).

Write/Read [W] FROM TOX7

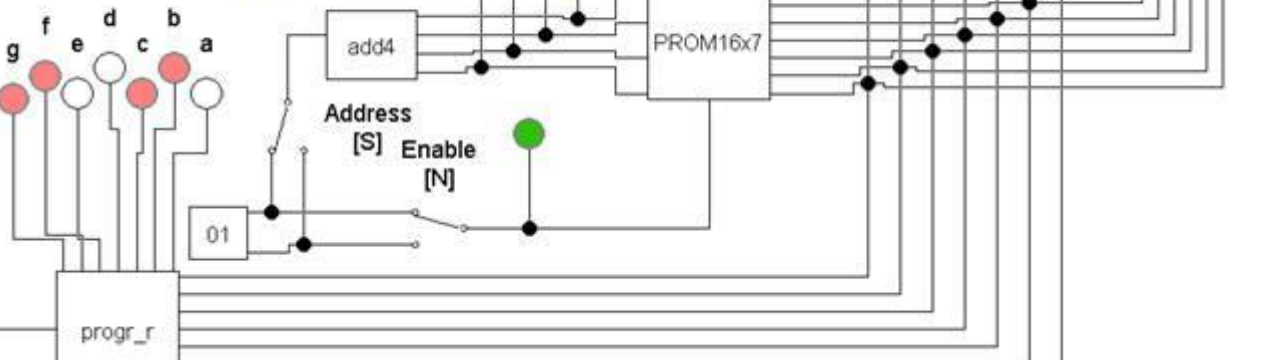


Рисунок 2

Программирование микросхемы осуществляется в следующем порядке:

- нажатием кнопки пуска (Activate simulation), активизировать схему;
- клавишей W (Write/Read) перевести установку в режим программирования (светится соответствующий светодиод);
- установить адрес программируемой строки матрицы памяти;
- подготовить информацию для программирования нажатием клавиш, соответствующих ликвидируемым сегментам предполагаемого к размещению на этой строке символа;
- кратковременным нажатием клавиши P (programming) обеспечить программирование строки (прошивку информации);
- восстановить исходные состояния клавиш сегментов;
- клавишей W (Write/Read) перевести установку в режим чтения информации и проверить прошитые в строках матрицы памяти информацию.

Следующие задания лабораторной работы, посвященные изучению различных структур систем памяти (2D, 3D, 2DM) и принципов их работы, выполняются на соответствующих моделях (см. рисунки 1.3-1.5).

Схема для исследования системы памяти по структуре 2D (см. рисунок 1.3) состоит из следующих блоков, оформленных в виде соответствующих подсхем:

- трехразрядный формирователь адреса (add3) на основе счетчика;
- четырехразрядный источник данных (data) на основе счетчика;
- система памяти RAM 8x4, в состав которой входят дешифратор на восемь выходов (dc8)

и матрица запоминающих ячеек (8x4), содержащая 8 ячеек (rg4) на основе регистров. Каждый из ячеек матрицы памяти снабжен индикатором, что позволяет вести наблюдение за выбором ячейки памяти во время эксперимента.

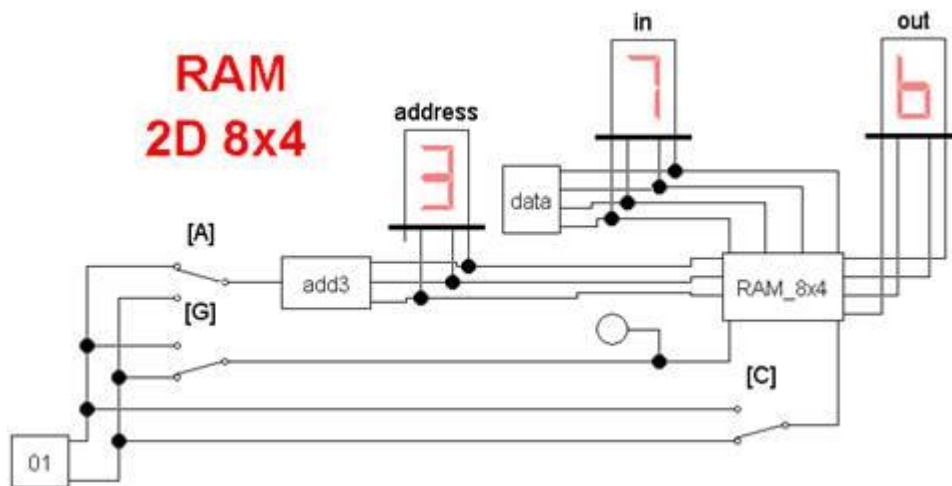


Рисунок 3

Схема для исследования системы памяти по структуре 3D (см. рисунок 1.4) состоит из следующих блоков, оформленных в виде соответствующих подсхем:

- четырехразрядный формирователь адреса (add4) на основе счетчика;
- четырехразрядный источник данных (data) на основе счетчика;
- система памяти RAM_16x4, в состав которой входят двоянный дешифратор (2dc4) и матрица запоминающих ячеек (16x4), содержащая 16 ячеек (rg4) на основе регистров.

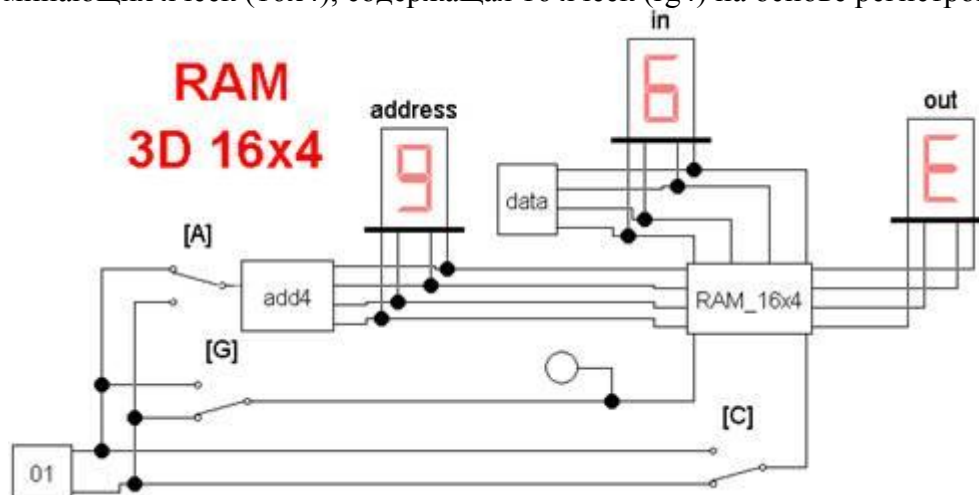


Рисунок 4

Схема для исследования системы памяти по структуре 2DM (см. рисунок 1.5) содержит в своем составе следующие блоки:

- блок памяти RAM_32x4 по структуре 2DM;
- блок подготовки данных Prepare;
- блок управления Control.

Блок памяти RAM_32x4 состоит из следующих подблоков:

- матрица элементов памяти 32x4;
- блок дешифрации строки dc8;
- блок выбора ячейки из строки (при чтении) mux4.

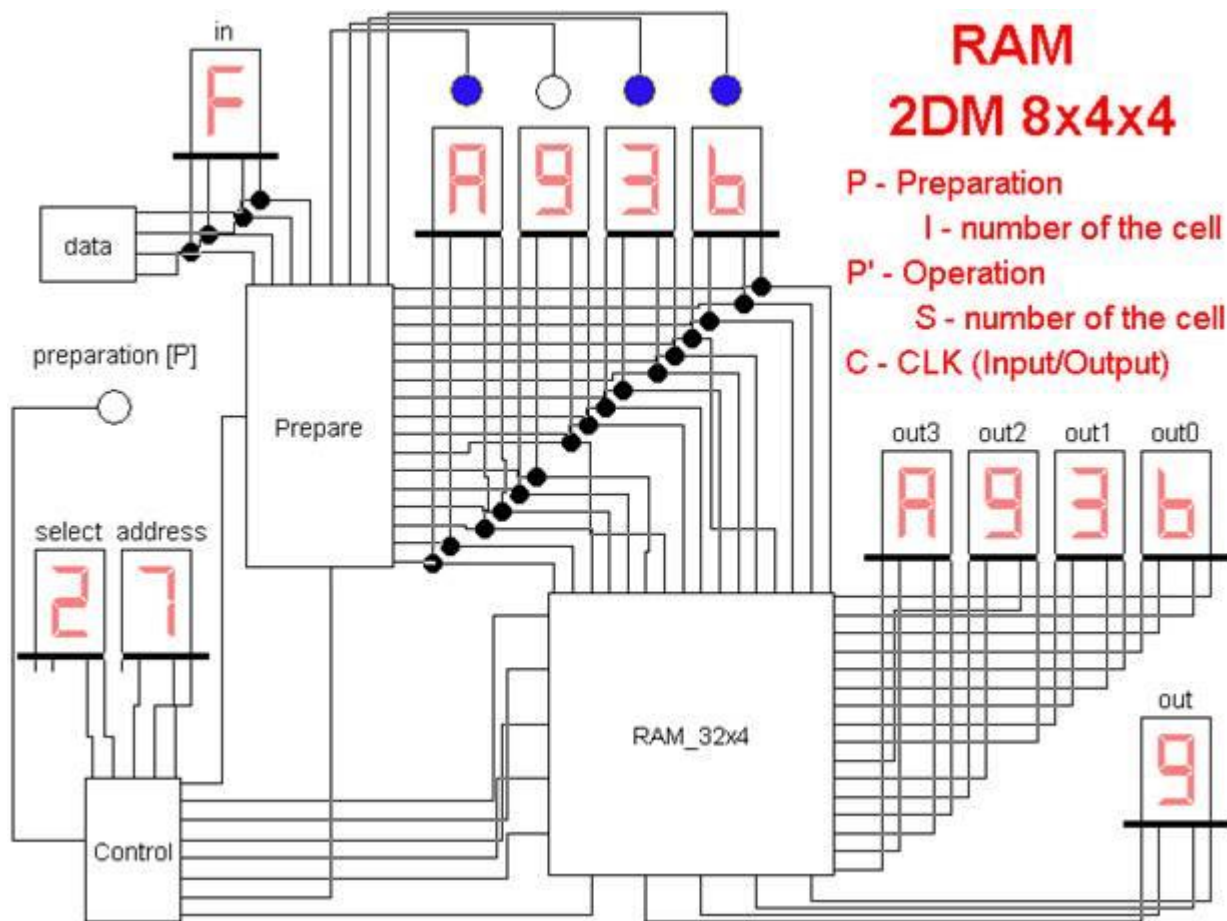


Рисунок 5

Матрица памяти 32x4 представляют собой (логически) 8 строк, в каждой из которых размещен блок cell_4x4, состоящая из четырех 4-разрядных ячеек памяти rg4. Физически блок памяти 32x4 реализован (с целью удобства соединения и обзора) в виде объединения двух блоков 16x4, составленных из четырех блоков cell_4x4. Каждый из внутренних блоков матрицы памяти 32x4 снабжен индикатором, что позволяет вести наблюдение за выбором строки и ячейки памяти во время эксперимента.

Блок dc8, предназначенный для выбора строки памяти, представляет собой дешифратор на восемь выходов.

Блок тих4, собранный на двух сдвоенных 4-входовых мультиплексорах, позволяет выбрать ячейку из активизированной строки.

Блок Prepare предназначен для предварительной подготовки четырех 4-разрядных слов, предполагаемых для одновременной записи в одну из строк матрицы памяти. На этапе подготовки эти слова временно сохраняются во внутреннем блоке in_4, составленного из четырех 4-разрядных ячеек памяти in. Выбор одной из этих ячеек осуществляется с помощью дешифраторного блока dc4 и блока его адресации 0_3.

В блоке Control объединены управляющие клавиши и дополнительные элементы, обеспечивающие режимы работы устройства. Устройство может работать в трех режимах: подготовки данных для записи, записи и чтения.

Режим подготовки данных осуществляется следующим образом:

- клавишей P (Prepare) перевести устройство в режим подготовки данных (светится соответствующий индикатор);
- клавишей I (In) выбрать одну из четырех ячеек блока in_4 (указывает соответствующий индикатор);
- клавишей C (CLK) записать в эту ячейку одну из чисел, формируемых блоком данных data.

Перевод в режим записи/чтения осуществляется повторным нажатием клавиши P (индикатор гаснет).

В режиме записи:

- клавишей A выбрать необходимую строку матрицы памяти (указывает соответствующий 7-сегментный индикатор);
- клавишей C (CLK) записать подготовленные данные в выбранную строку матрицы памяти.

В режиме чтения:

- клавишей A выбрать необходимую строку матрицы памяти (указывает соответствующий 7-сегментный индикатор);
- клавишей S (Select) выбрать определенную ячейку из активизированной строки матрицы памяти (указывает соответствующий 7-сегментный индикатор).

Рабочее задание

1. Изучить принципы построения ROM и порядок работы с ним:
 - открыть файл ROM(M) 16x7.ewb, раскрыть подсистемы различных блоков (см. рисунок 4.1) и изучить их структуру;
 - изменяя адреса ячеек памяти клавишей A (Address), проверить записанную в соответствующих строках памяти информацию.
2. Изучить принципы построения PROM и порядок его программирования:
 - открыть файл PROM 16x7.ewb (см. рисунок 4.2), изучить структуру матрицы chip16x7 и устройство программатора (programmer), раскрыв соответствующие подсистемы;
 - запрограммировать строки матрицы памяти для отображения символов из следующего списка: L, J, H, P, S, U, -, _, e, n, q, r и др.
3. Изучить принципы построения системы памяти по структуре 2D:
 - открыть файл RAM 2D_8x4.ewb, где представлена схема для исследования системы памяти по структуре 2D и изучения принципа ее работы (см. рисунок 4.3). Раскрыть подсистемы блоков и изучить их структуру;
 - исследовать работу представленной системы памяти, записывая в ячейки определенную информацию и проверяя затем их сохранность.
4. Изучить принципы построения системы памяти по структуре 3D:
 - открыть файл RAM 3D_16x4.ewb, где представлена схема для исследования системы памяти по структуре 3D и изучения принципа ее работы (см. рисунок 4.4). Раскрыть подсистемы блоков и изучить их структуру;
 - исследовать работу представленной системы памяти, записывая в ячейки определенную информацию и проверяя затем их сохранность.
5. Изучить принципы построения системы памяти по структуре 2DM:
 - открыть файл RAM 2DM_32x4.ewb, где представлена схема исследования блока памяти по структуре 2DM (см. рисунок 2.3) и изучить структуру ее составных блоков, раскрывая их и изучая соответствующий пояснительный текст;
 - исследовать работу устройства памяти RAM_2DM, записав информацию во все строки матрицы, а затем проверить возможность считывания из конкретной ячейки памяти.

Контрольные вопросы

1. Назовите основные параметры запоминающих устройств.
2. В чем состоит назначение ROM?
3. Какие элементы памяти используются в различных типах ROM?
4. Как и кем программируются PROM?
5. В чем состоит различие EPROM и EEPROM?
6. Какая схемотехника элементов памяти обеспечивает максимальное быстродействие RAM?
7. В чем состоит принцип построения RAM с одномерной и двумерной адресацией?
8. Сравните преимущества и недостатки структур 2D и 3D.
9. Объясните принципы осуществления структуры памяти 2DM и ее основные достоинства по сравнению с остальными структурами.

Практическое занятие №48. Исследование работы репрограммируемого постоянного запоминающего устройства

Цель работы: исследование особенностей функционирования больших интегральных схем (БИС) репрограммируемых постоянных запоминающих устройств (РПЗУ) в режиме записи и считывания информации.

Основные теоретические положения

ПЗУ предназначены для длительного хранения информации многократного использования (константы, таблицы данных, стандартные программы и т.д.). Запись информации в ПЗУ производится в процессе их изготовления. ПЗУ функционируют только в режиме считывания и сохраняют информацию при отключении питания.

В отличие от ПЗУ программируемые ПЗУ позволяют пользователю производить однократную запись (программирование) информации по каждому адресу. Основным режимом работы ППЗУ также является режим считывания информации.

Исследуемые в настоящей работе РПЗУ сохраняют информацию при отключении источников питания, а также допускают возможность ее многократной перезаписи электрическими сигналами непосредственно самим пользователем, что имеет принципиальное значение при отладке тех или иных систем. В отличие от ОЗУ быстродействие этих устройств в режиме записи информации значительно ниже, чем в режиме считывания информации. В связи с этим можно считать, что основным режимом работы РПЗУ является режим считывания информации.

Основными определяющими параметрами запоминающих устройств являются информационная емкость и быстродействие. В качестве единицы измерения информационной емкости используются бит, представляющий собой один (любой) разряд двоичного числа. Часто используются производные единицы:

байт (1 байт = 8 бит);

Кбайт (1 Кбайт = 2^{10} байт);

Мбайт (1 Мбайт = 2^{20} байт) и др.

Информационная емкость записывается, как правило, в виде произведения

$C_{\text{инф}} = n \times m$, где

n - число двоичных слов;

m - разрядность слова.

Например, емкость ОЗУ типа K155PY1 составляет

$C_{\text{инф}} = 16 \times 1 \text{ бит} = 16 \text{ бит}$.

Емкость ППЗУ типа K155PE3 равна

$C_{\text{инф}} = 32 \times 8 \text{ бит} = 256 \text{ бит} = 32 \text{ байта}$.

Такая форма записи характеризует также и организацию памяти. Так, в приведенном примере ОЗУ типа K155PY1 содержит 16 слов с разрядностью 1, а ППЗУ типа K155PE3 содержит 32 слова с разрядностью 8.

Быстродействие запоминающего устройства характеризуется величиной времени обращения. Время обращения - это интервал времени от момента подачи сигнала записи или считывания информации до момента завершения операции, т.е. минимальный интервал времени между двумя последовательными сигналами обращения к запоминающему устройству. Это время может составлять от долей до единиц микросекунд в зависимости от типа устройства.

В качестве примера запоминающего устройства рассмотрим БИС РПЗУ типа KP1601PP1 информационной емкостью

$C_{\text{инф}} 1K \times 4 = 4 \text{ Кбит}$ ($1K = 2^{10} = 1024$).

Условно-графическое обозначение микросхемы приведено на рис.1.

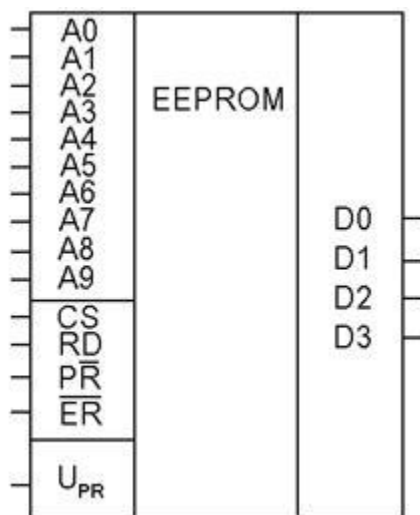


Рис.1

На рисунке использованы следующие обозначения:

A0...A9 - входы адреса

D0...D3 - входы / выходы данных

CS - выбор кристалла

RD - вход сигнала считывания

PR - вход сигнала программирования

ER - вход сигнала стирания

U_{PR} -вход напряжения программирования

Режимы работы микросхемы представлены в таблице 1.

Таблица 1

CS	ER	PR	RD	A0, A9	U _{PR}	D1/0	Режим
0	X	X	X	X	X	R _{off}	Хранение
1	0	1	0	X	-33, -31 В	X	Общее стирание
1	0	0	0	A	—//—	X	Избирательное стирание
1	1	0	0	A	—//—	D1	Запись данных
1	1	1	1	A	-33,5 В	D0	Считывание

В режиме хранения на вход С подается логический "0", при этом независимо от характера сигналов на других управляющих и адресных входах на выходах данных устанавливается высокоомное состояние (R_{off}).

При подаче CS = 1, ER = 0, PR = 1 и RD = 0 происходит стирание информации во всех ячейках памяти микросхемы, что соответствует для данной микросхемы установление всех ячеек в состояние логической "1".

При подаче сигналов CS = 1, ER = RD = 0 происходит избирательное стирание информации только по одному адресу A, установленному на входах A0 , A9 .

Для программирования ППЗУ на вход подается сигналы CS = 1 и PR = 0. При этом обеспечивается запись по заданному адресу A информации, поступившей на входы D0 , D3.

Для считывания информации по адресу A на вход микросхемы подаются сигналы CS = RD = 1. Считываемая информация поступает на выходы D0 , D3 микросхемы.

В режиме стирания и программирования на вход U_{PR} подается повышенное напряжение -33 , -31 В. В режиме считывания это напряжение может иметь любое значение в интервале от -33 В до 5 В.

Описание объекта и средств исследования

Функциональная схема исследуемого устройства представлена на рис.2.

Исследуемая микросхема запоминающего устройства ДД2 представляет собой ППЗУ с электрическим стиранием информации типа КР1601ПП1, рассмотренное выше.

Для задания кода адреса РПЗУ используются десять кнопок с фиксацией SA7...SA16. Отжатому состоянию кнопки соответствует сигнал логического "0", нажатому состоянию - сигнал логической "1" (при этом загорается соответствующий светодиод).

Данные для записи в РПЗУ формируются с помощью генератора пачки импульсов и счетчика СТ (ДД1). Число импульсов задается с помощью четырех кнопок с фиксацией на блоке К32 под надписью "Программатор СИ". Генератор запускается путем нажатия поочередно кнопок "Устан.0" и "Пуск". Число импульсов подсчитывается счетчиком, собранном на микросхеме типа К155ИЕ5, и в двоичном коде через шинный формирователь ВД подается на вход данных РПЗУ. При необходимости счетчик СТ может быть обнулен с помощью кнопки SA6.

Шинный формирователь ДДЗ выполняет функцию коммутатора, обеспечивающего заданную пересылку четырехразрядных слов данных. С этой целью в микросхеме ДДЗ предусмотрены три различные группы входов / выходов.

Входы DI предназначены для приема данных от внешних устройств (например, счетчика импульсов) и пересылки их в РПЗУ.

Выходы DO предназначены для передачи считываемых данных на блок индикации БИ2.

Выводы DI/O представляют собой входы или выходы микросхемы в зависимости от направления передачи данных.

При подаче на управляющий вход шинного формирователя Е сигнала логического "0" данные с входов DI подаются на выходы DI/O. При подаче на вход Е сигнала логической "1" данные с входов DI/O передаются на выход DO.

Блок формирования импульсов управления представляет собой устройство, формирующее сигнал управления работой РПЗУ.

В режиме "Общее стирание" БФИ формирует на входе ER РПЗУ сигнал логического "0". Сигнал формируется с помощью кнопки SA1 на блоке К32 путем перевода ее в нажатое состояние и обратно.

В режиме "Избирательное стирание" БФИ формирует на входах EP и RP РПЗУ сигналы логического "0". Сигналы формируются с помощью кнопки SA2 путем перевода ее в нажатое состояние и обратно.

В режиме "Запись информации" БФИ формирует сигналы логического "0" на входе PR РПЗУ и на входе Е шинного формирователя. Сигналы формируются с помощью кнопки SA3 путем перевода ее в нажатое состояние и обратно. Указанные сигналы формируются при условии, что одна из кнопок SA1 или SA2 находится в отжатом состоянии.

В режиме "Считывание информации" БФИ формирует сигнал логической "1" на входе RD РПЗУ и на входе Е шинного формирователя. Сигналы формируются с помощью кнопки SA4 путем перевода ее в нажатое состояние и обратно. Считывание информации производится из ячейки памяти с заданным адресом А. После считывания данные через шинный формирователь поступают на блок индикации БИ2.

Блок индикации БИ1, расположенный слева на передней панели блока К32, регистрирует число, находящееся в счетчике СТ2 (ДД1). Число представляется в десятичной форме с помощью двух семисегментных индикаторов (третьего и четвертого). Кнопка " IO | 2", расположенная под индикатором, должна находиться в отжатом состоянии.

Блок индикации БИ2, расположенный на панели справа, регистрирует данные, считываемые из РПЗУ. Информация на блоке индикации может быть представлена как в двоичной, так и в десятичной форме,

Вышеуказанный ряд питающих напряжений, необходимый для функционирования исследуемого устройства, формируется с помощью блоков питания стенда. Для подачи необходимых напряжений соответствующие кнопки питания должны находиться в нажатом состоянии, что сопровождается свечением индикаторов "+5", "+15", "-15", "-30".

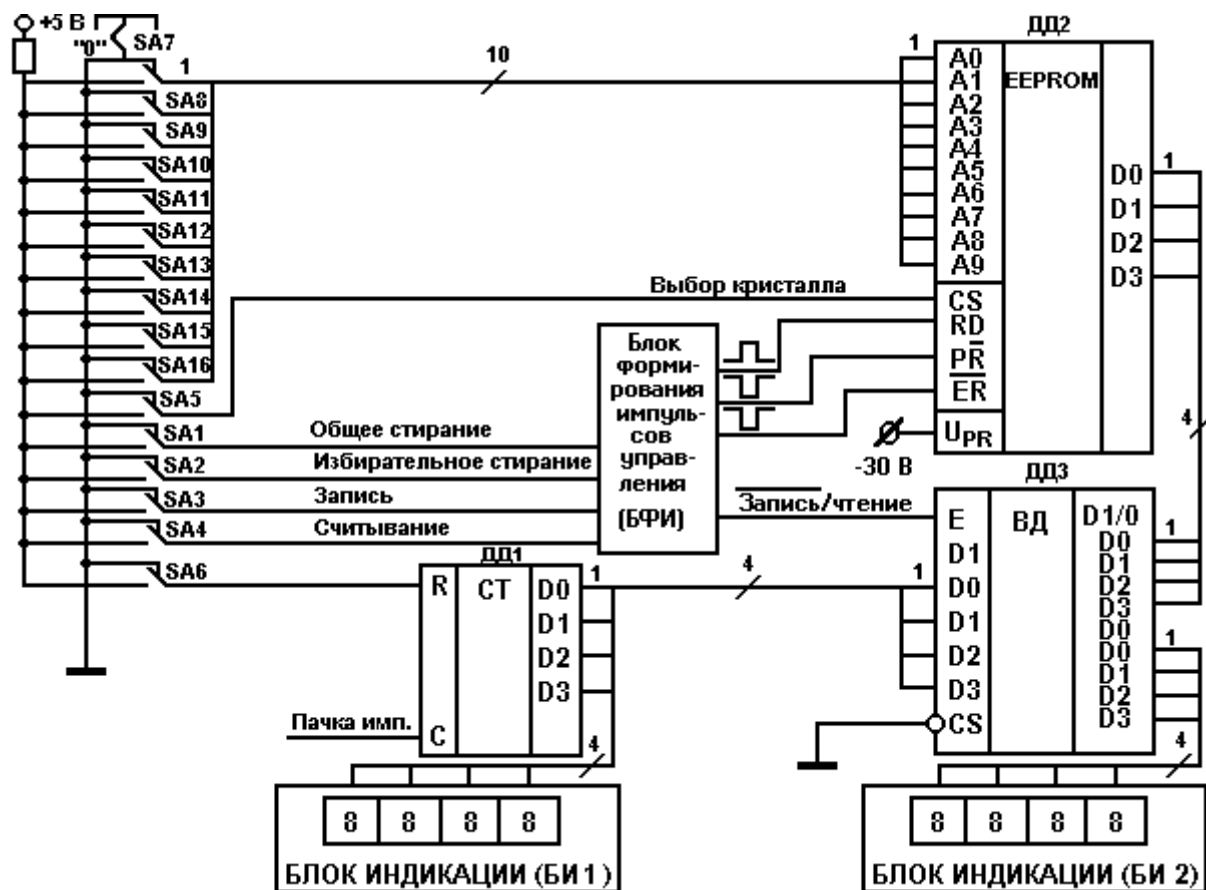


Рис.2

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Для исследования режимов работы РПЗУ подготовить исходную информацию в виде блока данных в двоичном коде и занести эти данные в таблицу (табл.2). Значения данных в десятичном коде предварительно согласовать с преподавателем.

Исследовать работу РПЗУ в режиме общего стирания информации.

Выполнить операции, указанные выше, и провести общее стирание информации в РПЗУ.

Провести считывание информации из РПЗУ по 8 последовательно расположенным адресам, начиная с адреса $A = 1$. Результаты измерений занести в таблицу (табл.2). Сделать выводы о работе РПЗУ в данном режиме.

Исследовать работу РПЗУ в режиме записи информации.

Выполнить операции, указанные ранее, и провести запись исходных данных по 8 последовательно расположенным адресам, начиная с адреса $A = 1$ в соответствии с табл.2

Таблица 2

№ п/п	Адрес	Исходные данные	Общее стир.	Запись	Избир. стир.	Общее стир.
1	0001					
2	0010					
3	0011					
4	0100					
5	0101					
6	0110					
7	0111					
8	1000					

Выполнить операции, указанные ранее, и провести считывание записанной в РПЗУ информации. Результаты измерений занести в таблицу (табл.2). Провести сравнение результатов записи с исходной информацией.

Исследовать работу РПЗУ в режиме избирательного стирания.

Выполнить операции, указанные ранее для первых 4-х адресов, начиная с адреса $A = 1$, проведя избирательное стирание информации по указанным адресам.

Провести считывание всего блока из 8-ми данных. Результаты считывания занести в таблицу (табл.2). Сделать выводы о работе РПЗУ в режиме избирательного стирания информации.

Провести общее стирание информации в РПЗУ, а затем повторное считывание исходного блока данных, начиная с адреса $A = 1$. Убедитесь, что информация в заданном массиве соответствует исходному состоянию и РПЗУ подготовлено к повторному программированию.

СОДЕРЖАНИЕ ОТЧЕТА

1. Название и цель работы.
2. Основные характеристики исследуемого РПЗУ.
3. Функциональная схема исследуемого устройства.
4. Таблица по п.4 и выводы о работе РПЗУ.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Дайте определение основных видов запоминающих устройств.
2. Назовите отличительные особенности ОЗУ, ПЗУ, ППЗУ и РПЗУ.
3. Приведите основные параметры запоминающих устройств и единицы их измерения.
4. Объясните основные режимы работы РПЗУ.

Практическое занятие №49. Диагностика и обслуживание FLASH накопителей.

Цель работы: формирование умений нахождения неисправностей флэш – накопителя.

Теоретический материал

Флэш-память - разновидность ЭСППЗУ, ее полное название - Flash Erase EEPROM (Electrically Erasable Programmable ROM) - можно перевести как "электрически стираемое программируемое постоянное запоминающее устройство". Другими словами, флэш-память - это энергонезависимая (т.е. не потребляющая энергии при хранении данных) перезаписываемая (т.е. данные можно стереть и записать заново при помощи электрического тока) память, содержимое которой можно быстро стереть (Flash Erase).

Флэш-память - это полупроводниковая память. Ее элементарная ячейка, в которой хранится один бит информации, представляет собой не конденсатор, а полевой транзистор со специальной электрически изолированной областью, которую называют "плавающим затвором" (floating gate). Электрический заряд, помещенный в эту область, способен сохраняться в течение многих лет. При записи одного бита данных ячейка заряжается - заряд помещается на плавающий затвор, при стирании - заряд снимается с плавающего затвора и ячейка разряжается.

Преимущества флэш-памяти по сравнению с другими средствами переноса и хранения данных очевидны: высокая надежность и удар прочность (результат отсутствия движущихся компонентов и простоты механической конструкции носителей и накопителей), малое энергопотребление, компактность. Однако у нее есть недостатки - ограниченное количество циклов перезаписи (от 10 тыс. до 1 млн.) и относительно медленная работа.

Флэш-память имеет несколько типов организации массива. Наибольшее распространение получила память типов И-НЕ (NAND), ИЛИ-НЕ (NOR). По архитектуре эти два типа имеют существенные различия.

Тип ИЛИ-НЕ (NOR) содержит ячейки, включенные параллельно друг другу и обеспечивает относительно быстрый произвольный доступ к данным, возможность побайтной записи информации. Однако, этот тип архитектуры имеет ячейки относительно большого размера, потому плохо масштабируется. Время стирания или записи гораздо больше, чем у других типов флэш-памяти.

Идеально подходит для хранения программ (BIOS, ПЗУ сотовых телефонов и т.д.), а так же для замены микросхем EEPROM.

Тип И-НЕ (NAND), содержит ячейки, включенные последовательно (гирляндой), между двумя линиями выборки. Группы ячеек объединяются в страницы. Страницы в блоки. Стоки

разных транзисторов такой гирлянды находятся на разных страницах. Поэтому произвольный доступ к ячейкам не возможен. Чтение, запись осуществляются одновременно только в пределах одной страницы, а стирание, осуществляются одновременно только в пределах одного или нескольких блоков. Однако, осуществляется быстрее, чем у типа ИЛИ-НЕ. Стирание/запись блока происходят так же достаточно быстро.

Устройство. Основные элементы ФЛЭШ – установлены на многослойной плате РСВ

- USB разъём тип А;
- стабилизатор питания контроллера и флэш из 5 в 3,3 вольт;
- микросхема контроллера;
- микросхема энергонезависимой NAND памяти;
- кварцевый резонатор, обычно на 12 Mhz.



Рисунок 1 - Расположение основных элементов флэш-накопителя

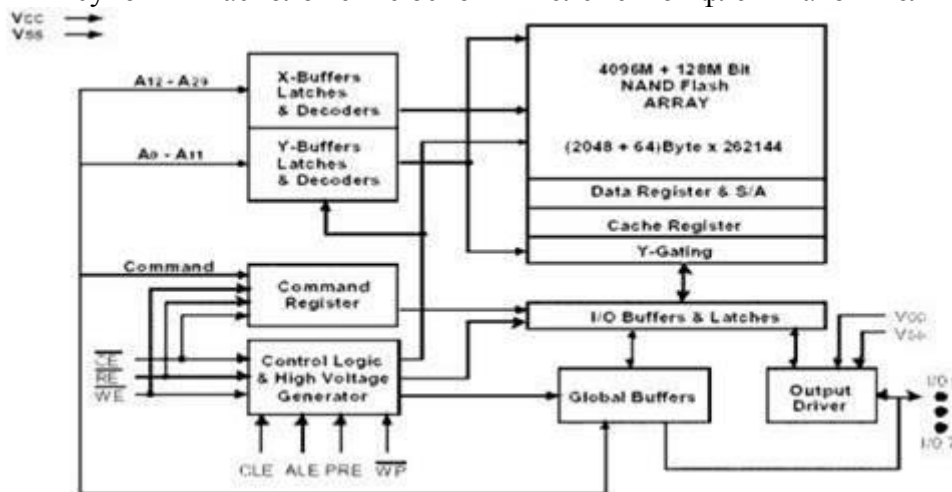


Рисунок 2 - Блок-схема flash накопителя

Элементы флэш-памяти назначение и симптомы их неисправности.

РСВ-многослойная печатная плата, на которой устанавливаются все элементы флэш. Типичные неисправности: некачественная пайка, внутренние обрывы проводников при механическом повреждении, удар, изгиб. Симптомы: нестабильная работа флэш.

USB разъём – некачественная пайка контактов.

Симптомы: флэш периодически не определяется.

Стабилизатор – конвертирует и стабилизирует напряжение поступающие с компьютера в напряжение необходимое для работы контроллера и флэш-памяти.

Симптомы: флэш не определяется совсем, или видно в системе как неопознанное устройство. Часто выходит из строя при переполюсовке USB разъёма.

NAND микросхема – энергонезависимая память. Симптомы: повреждение отдельных блоков памяти в связи со старением или по другим причинам, невозможность записи или чтения, лечится переформатированием фирменной утилитой с уменьшением общего размера флэш.

Контроллер – микросхема управления NAND памятью и передачи данных. В ней хранятся данные о типе микросхемы NAND, производителе и другая служебная информация необходимая для функционирования флэш накопителя. Симптомы: флэш определяется как неизвестное

устройство, нулевой или заниженный объем флэш памяти. Часто выходит из строя при «горячем» извлечении флэш. Обычно помогает перепрошивка контроллера фирменными утилитами.

Кварцевый резонатор – формирует опорную частоту для функционирования логики контроллера и флэш памяти. При поломке (что бывает крайне редко), флэш не определяется в системе.

Все неисправности флэш накопителей делятся на две группы:

- аппаратные
- программные

Аппаратные представлены как правило двумя основными видами:

Наиболее часто встречаются неисправности, связанные с механическими или электрическими повреждениями, вызванными нарушением правил эксплуатации устройства. Установив флэш в разъем на корпусе системного блока, пользователь по неосторожности зацепив ее рукой ломает разъем, зачастую вместе с печатной платой на которой распаяны микросхемы контроллера и собственно памяти. Печатная плата, несмотря на свою миниатюрность, четырехслойная и восстановить поврежденные дорожки во внутренних слоях невозможно.



Рисунок 3 - Внешний вид флэш-накопителя со сломанным разъемом

Вторая неисправность, которая часто встречается - это выгорание предохранителя или микросхемы стабилизатора питания. Происходит это чаще всего тогда, когда накопитель подключают к неправильно подключенному на материнскую плату USB-разъему, находящемуся на передней панели корпуса. Если вы не уверены в работоспособности этого разъема (на чужом компьютере, например), то лучше не лениться и подключить накопитель к разъему, распаянному непосредственно на материнскую плату сзади системного блока.



Рисунок 4 - Внешний вид флэш-накопителя со сгоревшим стабилизатором

Программные неисправности или так называемые «софтовые» проблемы. Это когда на накопителе нет видимых механических или электрических повреждений, но флэш определяется как неизвестное устройство, компьютер зависает при обращении к диску, неправильно определяется объем. Здесь опять, в первую очередь, надо определиться что важнее: исправность накопителя или данные на нем. Если система определяет накопитель как USB Storage device, но показывает, что его объем равен нулю или диск неотформатирован, или вместо списка файлов вы видите мешанину из символов, имеет смысл сначала воспользоваться программами восстановления данных, которые применяются для жестких дисков. Такими как R-Studio, Get Data Back или аналогичных. Как правило, если неприятность связана с некорректной таблицей разделов или ошибками файловой системы, эти программы позволяют скопировать информацию. Если же такой способ не дал результатов, а данные жизненно необходимы, то имеет смысл задуматься о пересадке микросхемы памяти на другой исправный носитель. Неправильная запись (логические нарушения) происходит из-за сбоев компьютера, неправильного извлечения устройства или истощения ресурса на запись микросхемы flash.

Задания на практическую работу

1. Описать характеристики флэш-накопителя

Таблица 1

Характеристики флэш-накопителя

Характеристика	Значение
Объём flash накопителя	
Скорость записи данных	
Скорость чтения данных	

2. Напишите типы и классы данных флэш-накопителей:

SD (Secure Digital)-miniSD и microSD (TransFlash)

SDHC (Secure Digital High Capacity)

SDXC (Secure Digital eXtended Capacity)

Контрольные вопросы

1. Дайте определение назначению флэш-накопителей?

2. Из каких частей состоит флэш-накопитель?

3. Как работает флэш-память?

4. Какие основные неисправности вы знаете?

Практическое занятие №50. Принципы работы кэш-памяти.

Цель работы: проверить работу различных алгоритмов замещения при различных режимах записи.

Задание

В качестве задания предлагается некоторая короткая "программа" (табл. 2), которую необходимо выполнить с подключенной кэш-памятью (размером 4 и 8 ячеек) в шаговом режиме для следующих двух вариантов алгоритмов замещения (табл. 1).

Таблица 1

Номера вариантов	Режим записи	Алгоритм замещения
1, 7, 11	Сквозная	СЗ, без учета бита записи
	Обратная	О, с учетом бита записи
2, 5, 9	Сквозная	БИ, без учета бита записи
	Обратная	О, с учетом бита записи
3, 6, 12	Сквозная	О, с учетом бита записи
	Обратная	СЗ, с учетом бита записи
4, 8, 10	Сквозная	БИ, без учета бита записи
	Обратная	БИ, без учета бита записи

Таблица 2

№ варианта	Номера команд программы						
	1	2	3	4	5	6	7
1	RD #12	WR 10	WR @10	ADD 12	WR R0	SUB 10	PUSH R0
2	RD #65	WRR2	MOV R4, R2	WR 14	PUSH R3	POP R3	CALL 002

3	RD #16	SUB #5	WR 9	WR @9	WR R3	PUSH R3	POP R4
4	RD #99	WR R6	MOV R7, R6	ADD R7	PUSH R7	CALL 006	POP R8
5	RD #11	WR R2	WE-@R2	PUSH R2	CALL 005	POP R3	RET
6	RD #19	SUB #10	WR9	ADD #3	WR ©9	CALL 006	POPR4
7	RD #6	CALL 006	WR11	WRR2	PUSH R2	RET	JMP 002
8	RD #8	WRR2	WR@R+	PUSH R2	POP R3	WR-@R3	CALL 003
9	RD #13	WR14	WR@14	WR@13	ADD 13	CALL 006	RET
10	RD #42	SUB #54	WR16	WR@16	WRR1	ADD @R1+	PUSH R1
11	RD #10	WRR5	ADD R5	WRR6	CALL 005	PUSH R6	RET
12	JMP 006	RD #76	WR 14	WRR2	PUSH R2	RET	CALL 001

Не следует рассматривать заданную последовательность команд как фрагмент программы. Некоторые конструкции, например, последовательность команд PUSH R6, RET в общем случае не возвращает программу в точку вызова подпрограммы. Такие группы команд введены в задание для того, чтобы обратить внимание студентов на особенности функционирования стека.

Порядок выполнения работы

1. Ввести в модель учебной ЭВМ текст своего варианта программы (см. табл. 2), ассемблировать его и сохранить на диске в виде txt-файла.
2. Установить параметры кэш-памяти размером 4 ячейки, выбрать режим записи и алгоритм замещения в соответствии с первой строкой своего варианта из табл. 13.
3. В шаговом режиме выполнить программу, фиксируя после каждого шага состояние кэш-памяти.
4. Для одной из команд записи (WR) перейти в режим **Такт** и отметить, в каких микрокомандах происходит изменение кэш-памяти.
5. Для кэш-памяти размером 8 ячеек установить параметры в соответствии со второй строкой своего варианта из табл. 13 и выполнить программу в шаговом режиме еще раз, фиксируя последовательность номеров замещаемых ячеек кэш-памяти.

Содержание отчета

1. Вариант задания — текст программы и режимы кэш-памяти.
2. Последовательность состояний кэш-памяти размером 4 ячейки при однократном выполнении программы (команды 1—7).
3. Последовательность микрокоманд при выполнении команды wr с отметкой тех микрокоманд, в которых возможна модификация кэш-памяти.
4. Для варианта кэш-памяти размером 8 ячеек — последовательность номеров замещаемых ячеек кэш-памяти для второго варианта параметров кэш памяти при двукратном выполнении программы (команды 1—7).

Контрольные вопросы

1. В чем смысл включения кэш-памяти в состав ЭВМ?
2. Как работает кэш-память в режиме обратной записи? Сквозной записи?
3. Как зависит эффективность работы ЭВМ от размера кэш-памяти?
4. В какую ячейку кэш-памяти будет помещаться очередное слово, если свободные ячейки отсутствуют?
5. Какие алгоритмы замещения ячеек кэш-памяти вам известны?