

Федеральное агентство по образованию РФ
Государственное образовательное учреждение высшего профессионального образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ГОУВПО «АмГУ»)

УТВЕРЖДАЮ

Зав. кафедрой АППиЭ

_____ А.Н. Рыбалев

«___» _____ 200__г.

Энергетический факультет

кафедра «Автоматизация производственных процессов и электротехники»

Учебно-методический комплекс дисциплины
ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ,
СИСТЕМЫ И СЕТИ

для специальности

22.03.01 «Автоматизация технологических процессов и производств»

Составитель:

Д.А. Теличенко

Благовещенск 2008

*Печатается по решению
редакционно-издательского совета
энергетического факультета
Амурского государственного
университета*

Вычислительные машины, системы и сети для специальности 22.03.01 «Автоматизация технологических процессов и производств»: учебно-методический комплекс дисциплины. / Теличенко Д.А. – Благовещенск. Изд-во Амурского гос. ун-та, 2008. 185 с.

Учебно-методический комплекс дисциплины «Вычислительные машины, системы и сети» представляет собой совокупность учебно-методических документов, призванных обеспечить организацию и содержательную целостность системы методов и средств обучения. Основной целью данного комплекса является систематизация содержания дисциплины, улучшение ее методического обеспечения, правильное планирование и организация работы и контроля знаний студентов.

©Амурский государственный университет, 2008

©Кафедра автоматизации производственных процессов
и электротехники, 2008

©Теличенко Денис Алексеевич, 2008

ОГЛАВЛЕНИЕ

Предисловие.....	5
1. Программа дисциплины.....	6
1.1 Выписка из Государственного образовательного стандарта высшего профессионального образования.....	6
1.2. Рабочая программа.....	6
2. График самостоятельной работы студентов.....	23
3. Методические рекомендации по проведению практических занятий и самостоятельной работы студентов (<i>формы проведения</i>).....	24
3.1 Практические занятия.....	24
3.2 Самостоятельная работа.....	24
4. Методические рекомендации по проведению лабораторных занятий (<i>формы проведения</i>).....	25
5. Список рекомендуемой литературы.....	27
6. План конспект лекций по каждой теме.....	28
7. Методические указания по выполнению лабораторных работ (<i>темы и рассматриваемые вопросы</i>).....	40
8. Методические указания к практическим занятиям (<i>темы и рассматриваемые вопросы</i>).....	45
9. Методические указания по выполнению контрольных работ (<i>формы проведения, темы и рассматриваемые вопросы</i>)..	49
9.1 Формы проведения контрольных работ.....	49
9.2 Темы и рассматриваемые вопросы контрольных работ.....	50
10. Перечень программных продуктов и методические указания по применению современных ИТ для преподавания учебной дисциплины.....	51
10.1 Возможности «Electronics Workbench».....	51
10.2 Работа с программой «Electronics Workbench».....	51

11.	Методические указания по организации межсессионного и экзаменационного контроля знаний студентов.....	55
	11.1 Организация межсессионного контроля.....	55
	11.2 Организация экзаменационного контроля.....	56
12.	Комплекты заданий.....	57
	12.1 Лабораторные работы.....	57
	12.2 Контрольные работы.....	143
	12.3 Домашние задания (РГР).....	146
13.	Фонд тестовых и контрольных заданий для оценки качества знаний по дисциплине.....	154
	13.1 Тест по главе 1.....	154
	13.2 Тест по главе 2.....	162
	13.3 Тест по главе 3.....	166
14.	Комплект экзаменационных билетов.....	172
15.	Карта обеспеченности дисциплины кадрами профессорско-преподавательского состава.....	185

ПРЕДИСЛОВИЕ

Учебно-методический комплекс дисциплины Вычислительные машины системы и сети:

составлен на основании Государственного образовательного стандарта ВПО 220300 «Автоматизированные технологии и производства» и учебного плана специальности 22.03.01 «Автоматизация технологических производств»: блок специальных дисциплин, СД.03 «Вычислительные машины, системы и сети»

обсужден на заседании кафедры автоматизации производственных процессов и электротехники

«__» _____ 200__ г., протокол № ____

Заведующий кафедрой _____ А.Н. Рыбалев

одобрен на заседании УМС 22.03.01 Автоматизация технологических процессов и производств

«__» _____ 200__ г., протокол № ____

Председатель _____ А.Н. Рыбалев

СОГЛАСОВАНО

Начальник УМУ

Г.Н. Торопчина
(подпись, И.О.Ф.)

«__» _____ 200__ г.

СОГЛАСОВАНО

Председатель УМС факультета

Ю.В. Мясоедов
(подпись, И.О.Ф.)

«__» _____ 200__ г.

СОГЛАСОВАНО

Заведующий выпускающей кафедрой

А.Н. Рыбалев
(подпись, И.О.Ф.)

«__» _____ 200__ г.

1. ПРОГРАММА ДИСЦИПЛИНЫ

1.1 ВЫПИСКА ИЗ ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

Вычислительные машины, системы и сети:

принципы построения вычислительных машин (ВМ), модели вычислений, многоуровневая организация вычислительных процессов, аппаратные и программные средства, классификация, назначение; понятия о функциональной, структурной организации и архитектуре ВМ; основные характеристики ВМ, методы оценки; влияние технологии производства интегральных схем на архитектуру и характеристики, классификация ВМ, система памяти, средства реализации, иерархическая организация, характеристики, архитектурные методы повышения производительности, процессоры, устройства; организация управления, адресация, система команд, производительность процессора, методы оценки, архитектурные способы повышения производительности, современные микропроцессоры, тенденции развития; микроконтроллеры, тенденции развития; типы и основные принципы построения периферийных устройств, организация ввода-вывода, прерывания, персональные компьютеры; принцип открытой архитектуры, шины, влияние на производительность, системный контроллер и контроллер шин, организация внутримашинных обменов, особенности организации рабочих станций и серверов, многомашинные комплексы, стандартные интерфейсы для связи компьютеров, многопроцессорные системы, оценки производительности, телекоммуникации и компьютерные сети, влияние сетевых технологий на архитектуру компьютеров, промышленные системы, унификация, комплексирование информационных и управляющих систем.

1.2 РАБОЧАЯ ПРОГРАММА

по дисциплине «Вычислительные машины системы и сети»

для специальности 22.03.01 «Автоматизация технологических процессов и производств»

Курс 3

Семестр 6

Лекции 54 (час.)

Экзамен 6

Практические (семинарские) занятия 18 (час.)

Лабораторные занятия 18 (час.)

Самостоятельная работа 55 (час.)

Всего часов: 145.

ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ, ЕЕ МЕСТО В УЧЕБНОМ ПРОЦЕССЕ

Цель преподавания дисциплины – сформировать у студентов знания о методах и способах использования вычислительных машин и компьютерных систем для решения прикладных задач в области автоматизации производственных процессов.

Задача изучения дисциплины: привить навыки по оценке, выбору и использованию современной вычислительной и управляющей техники; ознакомиться с основами проектирования средств автоматики и систем управления на основе вычислительной техники и микроконтроллеров; развить умение применять приемы и технологии современных информационных и управляющих сетей.

Дисциплина базируется на курсах: «Математика», «Физика», «Информатика», «Общая электротехника и электроника».

Знания и умения, приобретенные студентами при изучении дисциплины, используется в специальных курсах, в частности «Микропроцессорные системы управления», при курсовом и дипломном проектировании, а также в практической деятельности выпускника.

ТРЕБОВАНИЯ К УРОВНЮ ОСВОЕНИЯ СОДЕРЖАНИЯ ДИСЦИПЛИНЫ

В результате изучения дисциплины студенты должны:

знать:

- принципы организации и архитектуру вычислительных машин, систем и сетей;
- принципы функционирования вычислительных машин, систем и сетей;
- принципы организации функциональных и интерфейсных связей вычислительных систем;
- основные современные информационные технологии передачи и обработки данных;
- основы построения управляющих локальных и глобальных сетей;

уметь:

- анализировать работу и проектировать отдельные цифровые узлы современных вычислительных машин и их комплексов;
- правильно выбирать средства для анализа и проектирования систем автоматизации и управления;
- анализировать работу вычислительных машин и сетей;
- знать способы решения задач управления с использованием локальных и глобальных вычислительных сетей, в том числе основы проектирования локальных и глобальных вычислительных сетей.

СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

1. ЛЕКЦИОННЫЙ КУРС (54 часа)

1.1. Введение. Основные понятия о процессе автоматизированной обработки данных – 2 ч.

Информация. Меры информации (определение, формулы, примеры): объем данных, количество переданной информации, априорная и апостериорная информация, энтропия, степень информативности, тезаурус, полезность информации, экономический эффект. Показатели качества информации (определение, формулы, примеры): репрезентативность, содержательность, достаточность, доступность, актуальность, своевременность, точность, достоверность, ценность.. Выводы и замечания.

1.2. Принцип действия ВМ. Логические основы, построение и работа простейших цифровых устройств – 2 ч.

Системы исчисления. Представление информации в ВМ. Арифметические основы двоичной системы исчисления. Логические основы построения ВМ. Принципы и формы описания. Алгебра логики – основные понятия и базовые операции. Формы представления цифровых устройств. Основные законы алгебры логики. Простейшие цифровые узлы. Комбинационные схемы, последовательностные схемы. Основные принципы проектирования цифровых устройств и этапы.

1.3. Архитектура ВМ. Классификация ВМ и история развития вычислительной техники – 2 ч.

Определение ВМ, ее структуры и архитектуры. Быстродействие и производительность. Характеристики ВМ. Классификация средств электронно-вычислительной техники. Замечания по способам классификации ВМ. Принцип академика В.М. Глушкова. ВМ с точки зрения использования человеком.

1.4. Архитектура ВМ. Функциональная и структурная организация ВМ – 2 ч.

Общие понятия о функциональной и структурной организации ВМ. Различные точки зрения на функционирование вычислительного процесса. Точка зрения пользователя и программиста. Определение функциональной и структурной организации ВМ. степени детальности структурных схем. Обобщенная структура ВМ. Схема обобщенной структуры. Характеристики каждой из подсистем обобщенной структуры и ее особенности. Структура и состав ВМ. Концепция Дж. Фон Неймана. Устройство управления, арифметическо-логическое устройство, память и устройство ввода-вывода. Потоки данных, команд и управляющих сигналов. Принципы построения ВМ. Основные принципы взаимодействия между элементами вычислительной системы и основы их функционирования.

1.5. Аппаратные особенности ВМ различных поколений. Принцип построения и функционирования ВМ пятого поколения – 2 ч.

Вычислительные машины первого и второго поколений: структура, основные элементы, принципы взаимодействия, особенности построения. Структура простейшего АЛУ. Вычислительные машины третьего поколения: структура, основные элементы, принципы взаимодействия, особенности построения. Вычислительные машины четвертого и пятого поколений: структура, основные элементы, принципы взаимодействия, особенности построения. Основ-

ные принципы построения ВМ пятого поколения. Общие принципы функционирования ВМ пятого поколения. Кризис структуры Дж. Фон Неймана. Вычислительные машины шестого поколения.

1.6. Особенности организации современных однопроцессорных ВМ – 2 ч.

Понятие однопроцессорных и многопроцессорных систем. Таксономия М. Флина. SISD (ОКОД)-компьютеры: определение, характеристика, основные элементы, структура, принципы функционирования. CISC архитектура, RISC архитектура. Суперскалярная обработка: аппаратная реализация, VLIW архитектура. SIMD (ОКМД)-компьютеры: определение, характеристика, основные элементы, структура, принципы функционирования. Матричная архитектура, векторно-конвейерная архитектура, MMX технология.

1.7. Организация микропроцессоров. Функциональная структура МП – 2 ч.

Процессор: предназначение, понятие микропроцессора, основные достоинства, направление применения. Структурно-функциональная схема микропроцессора: Операционный блок – состав, предназначение основных элементов, основные операции, специализированные блоки аппаратного умножения и деления, разрядность микропроцессора, понятие специализации регистров, примеры, особенности изменения содержимого регистра словосостояния. Блок управления – предназначение, выполняемые действия, состав, особенности функционирования, фазы выполнения команд, структура команд, микрокоманды, микрооперации, понятие микропрограммного автомата с мягкой и жесткой логикой, особенности современной реализации элементов блока управления. Интерфейсный блок – предназначение, понятие интерфейса ввода-вывода, функции интерфейсного блока, понятие системной шины и электрической спецификации сигналов на шине, цикл шины, принцип квитирования.

1.8. Особенности организации процессоров при использовании внутренних регистров – 2 ч.

Аккумуляторная архитектура – основы построения, принцип работы, характеристики, примеры. Много аккумуляторная архитектура – основы построения, принцип работы, характеристики, примеры. Стековые процессоры. Понятие стека, дисциплина FIFO, LIFO. Принцип работы стека и его предназначение. Адресация в стеке. Примеры работы со стеком. Характеристики стека. Особенности стековых процессоров, их структура. Пример стекового процессора на базе специализированного арифметического сопроцессора для вычислений с плавающей точкой.

1.9. Система команд – 2 ч.

Система команд как одна из важнейших характеристик микропроцессора. Понятие системы команд (форматы команд и обрабатываемых данных, список команд и их функциональное назначение, способы адресации данных). Группы команд по функциональным признакам (предназначение, принцип работы, примеры, особенности использования): команды пересылок данных и ввода-вывода; арифметические и поразрядные двоичные команды; коман-

ды передачи управления. Замечания по системе команд современных микропроцессоров. Структура команд: операционная и адресная часть, их предназначение, характеристика. Особенности адресной части команд. Естественный и принудительный способ адресации команд. Примеры и пояснения.

1.10. Способы адресации – 2 ч.

Общие сведения по способам адресации. Адресация данных. Прямая адресация, принцип работы и особенности. Прямая регистровая адресация. Примеры. Непосредственная адресация – принцип, особенности, примеры. Неявная адресация – принцип, особенности, примеры. Косвенная адресация – принцип, особенности, примеры. Особенности и преимущества косвенной адресации на примере организации цикла. Относительная адресация или базирование – принцип, особенности, примеры. Формирование исполнительного адреса. Страничная организация и сегментированная память. Базовая и индексная адресация (особенности, принцип, примеры). Адресация команд.

1.11. Особенности организация памяти ВМ – 4 ч.

Понятие памяти ВМ, характеристики отдельных устройств памяти. Быстродействие памяти, время доступа к памяти, длительность цикла памяти. Противоречивость требований к увеличению емкости и быстродействию памяти. Уровни памяти. Сверхоперативный уровень, оперативный уровень, внешний уровень. Замечания по производительности ВМ и особенностям организации памяти. Организация внутренней памяти процессора (сверхоперативный уровень). Организация оперативной памяти (оперативный уровень). Базовые типы оперативной памяти принцип работы, особенности, сравнительная характеристика, методы управления оперативной памятью: методы управления без использования виртуальной памяти (распределение памяти фиксированными разделами, распределение памяти динамическими разделами, распределение памяти с перемещаемыми разделами); методы управления с использованием виртуальной памяти (понятие виртуальной памяти, задачи виртуальной памяти, страничное распределение, сегментное распределение, странично-сегментное распределение, свопинг). Системы внешней памяти: жесткие диски, гибкие магнитные диски, CD, DVD, новые форматы записи, flash. Методы организации кэш памяти, ее структура и принцип работы. Способы размещения данных в кэш памяти. Методы обновления строк в основной памяти, связь с кэш-памятью. Методы повышения пропускной способности оперативной памяти.

1.12. Организация обмена данными в ВМ – 2 ч.

Общие сведения. Принципы организации обмена. Обмен данными между периферийными устройствами и вычислительным ядром системы. Особенности организации. Программно-управляемая передача. Передача информации с прерыванием программы. Понятие прерывания аппаратные и программные прерывания. Сигнал запроса прерывания. Работа системы при реакции на прерывания. Сравнения и выводы по программно-управляемой передаче и передаче с прерыванием. Передача информации в режиме прямого доступа к памяти (ПДП). Определение режима ПДП. Предназначение режима ПДП. Достоинства и недостатки. Способы организации, примеры.

1.13. Персональные компьютеры (ПК), особенности архитектуры и применения» – 6 ч.

Функциональная и структурная организация класса ПК. Микропроцессор, особенности, современные варианты выполнения. Разрядность, адресное пространство. Рабочая тактовая частота. Состав инструкций, конструктив, рабочее напряжение. Конвейерность, многозадачность работы, защищенный режим. Система виртуальных машин. Динамическое исполнение команд. Особенности системной шины, основной памяти, внешней памяти. Источник питания и таймер. Внешние устройства (их особенности). Дополнительные интегральные схемы. Функциональные характеристики ПК. Системная плата. Системный и периферийный интерфейс – определение, состав. Шины расширений – предназначение, характеристики, примеры использования. Локальные шины – предназначение, характеристики, примеры использования. Периферийные шины – предназначение, характеристики, примеры использования, протоколы взаимодействия. Интерфейс RS-232 и стандарт IEEE 1284, последовательные и параллельные порты. Универсальные последовательные периферийные шины. Универсальная шина USB. Технология Bluetooth. Стандарт IEEE 1394, цифровой последовательный интерфейс Fire Wire. Шина PCMCIA. Расширенный интерфейс ACPI.

Микропроцессорная память и кэш память в ПК – особенности. Основная память ПК. Общие замечания по организации: RAM и ROM, сравнение и особенности SRAM и DRAM. Сигналы управления RAS и CAS. Физическая структура основной памяти. Матричная организация памяти в ПК. Кодовые шины адреса, дешифраторы полуадресов, сигналы записи/считывания, регистр данных. Куб памяти. Типы модулей оперативной памяти – особенности, основные области применения, сравнительные характеристики, основные частоты и пропускная способность. Типы оперативной памяти – особенности, виды, характеристики, одноканальные и многоканальные модули, разрядность, тактовая частота, пиковая пропускная способность. Постоянные запоминающие устройства ПК: постоянные запоминающие устройства (ПЗУ), программируемые запоминающие устройства (ППЗУ); перезаписываемые запоминающие устройства (EEPROM) – принципы работы, предназначение, особенности использования и реализации. Внешние запоминающие устройства ПК. Понятие файловой системы. Режимы обмена между внешней памятью и оперативной памятью. Время доступа к информации в ПК. Технология SMART. Особенности Флэш-памяти ПК: принцип работы, характеристики, надежность, конструктивные варианты исполнения. Дисковые массивы RAID. Особенности использования в ПК. Уровни конфигурации RAID. Дисковые массивы различных поколений.

1.14. Проектирование микропроцессорных систем и средств автоматизации на их основе – 4 ч.

Проектирование. Уровни представления микропроцессорной системы – особенности, использование, характеристики. Разработка систем: ошибки, неисправности, дефекты. Основы тестового контроля. Субъективные неисправности, проектные неисправности. Ошибки – понятие, примеры. Дефекты

– понятие, примеры. Процедура отладки. Основные методы испытаний. Обнаружение ошибки и диагностика неисправности. Лабораторные испытания, уровни обнаружения. Свойство контролепригодности системы. Функции средств отладки. Этапы проектирования микропроцессорных систем и средств автоматики на их основе. Источники возникновения ошибок на соответствующих этапах проектирования. Проверка правильности проекта. Автономная отладка аппаратного обеспечения. Отладка программ. Комплексная отладка микропроцессорной системы. Встроенные средства тестирования и отладки при работе системы: состав, предназначение, примеры. JTAG-интерфейс. Разработка программного обеспечения. Понятие алгоритма. Машинно-ориентированные, процедурно-ориентированные и проблемно-ориентированные языки программирования. Трансляторы (компиляторы, интерпретаторы). Некоторые аспекты создания исполняемых программ на языке низкого уровня (Ассемблер). Процедуры формирования программы: составление, трансляция, листинг, компоновка объектного модуля, отладка программы, выполнение программы. Структура программы на языке ассемблера для получения исполняемого файла формата EXE. Основные сведения о листинге и его структуре. Краткие сведения об отладчике программ DEBUG.

1.15. Системное и прикладное программное обеспечение современных ВМ и МПС. Интерфейс пользователя – 4 ч.

Структура программного обеспечения – предназначение и основные характеристики. Общее (системное) программное обеспечение. Операционные системы – цели применения, набор программных модулей. Основные примеры операционных систем и их особенности. Структура DOS: программа начальной загрузки, базовая система ввода-вывода, базовый модуль DOS, командный процессор, утилиты. Системы автоматизации программирования. Комплекс технического обслуживания: проверочные тесты, наладочные тесты, диагностические тесты. Пакеты программ дополняющих возможности и системы документации. Специальное или прикладное обеспечение: пакеты прикладных программ: предназначение, и основные характеристики. Текстовые процессоры, редакторы широкого назначения, издательские системы, системы обработки электронных таблиц и табличные процессоры, графические редакторы, системы управления базами данных, графические редакторы, интегрированные системы.

1.16. Централизованные и распределенные системы обработки данных – 2 ч.

Понятие о централизованных и распределенных системах обработки данных и системах реального времени. Обобщенная структура типовой системы управления (микроконтроллера), ее состав. Объект управления исполнительные устройства, система датчиков, устройства сопряжения с объектом, пульт управления, микропроцессорный (цифровой регулятор). Реальное время протекания процесса, шаг квантования. Иерархическая организация системы управления сложными, распределенными в пространстве объектами – двух и трех уровневая модель. Организация микроконтроллерных систем. Встраиваемые системы управления. Централизованная и распределенная система.

Отличительные черты микропроцессорной организации цифровых регуляторов. Встраиваемые средства на базе микроконтроллеров – функции, способы организации. Типы микроконтроллерных систем: автономная, локальная, сетевая конфигурация. Типовая структура микроконтроллера – общие сведения. Основные типы и семейства микроконтроллеров. Базовые принципы организации, состав, основные модули.

1.17. Вычислительные системы параллельной обработки данных – 4 ч. Параллельная обработка данных как архитектурный способ повышения производительности. Методы увеличения производительности вычислительных систем. Основы параллельной обработки. Мультипроцессорные архитектуры, ее преимущества. Трудности реализации мультипроцессорных архитектур (новые типы ошибок, сложности понимания и анализа параллельных процессов, недостаточная разработанность теоретических моделей и методов параллельного программирования). Классификация систем параллельной обработки. Многопроцессорные вычислительные системы (особенности организации и функционирования, примеры, преимущества и недостатки). MISD компьютеры (МКОД), MIMD компьютеры (МКМД), многопроцессорные вычислительные системы (сильно связанные): с общей шиной, с использованием многоходовой памяти, многомашинные вычислительные системы (слабосвязанные): многомашинные комплексы, системы массового параллелизма.

1.18. Вычислительные системы – состояние, производительность, направления развития – 2 ч.

Состояние производства и использования. Направления развития архитектуры. Направления развития высокопроизводительных вычислительных систем. Тенденции развития архитектур с общей памятью. Тенденции развития архитектур систем с разделяемой памятью. Развитие архитектур микропроцессоров высокопроизводительных вычислительных систем (организация внутрикристалльной памяти, увеличение состава и числа функциональных устройств, интеграция функций). Направления развития мультипроцессорных систем с распределенной памятью. Производительность мультипроцессорных систем при увеличении числа процессоров. Вычислительные системы на кристалле. Переход к нанотехнологии производства интегральных схем.

1.19. Телекоммуникационные вычислительные сети – 6 ч.

Принципы построения вычислительных сетей. Основные понятия: телекоммуникационная сеть, абонентская станция, телекоммуникационная система. Обобщенная функциональная схема. Организация и работа простейшей сети. Формат сообщений при обмене. Каналы передачи сообщений. Помехоустойчивое кодирование. Последовательность действий при передаче/приеме сообщений. Реакция на подтверждение приема. Параметры производительности сети: задержка времени в передающем узле, время передачи данных, время продвижения сигналов, задержка в приемном узле, время транспортировки, время обмена. Классификация вычислительных сетей: глобальные, региональные, локальные, системные. Архитектурные принципы построения сетей. Физические блоки, логическая организация, топология сети, основы

обработки сообщений. Протокол сети. Семиуровневая эталонная модель взаимодействия открытых систем OSI/ISO. Уровни иерархии (прикладной, представительский, сеансовый, транспортный, сетевой, канальный, физический) – характеристика, предназначение, организация. Коммутация и маршрутизация при передаче данных в сети. Коммутация сообщений и пакетов. Дейтограмма. Виртуальный канал. Основы маршрутизации. Основные типы сетевого оборудования: коммутаторы, концентраторы, повторители, мосты, шлюзы, маршрутизаторы, мультиплексоры. Локальные вычислительные сети (ЛВС). Характеристики ЛВС. Типы каналов, способы организации. Асинхронный и синхронный формат сообщений. Цифровые коды. Топологии ЛВС. Одноранговые и многогранговые сети. Файл-сервер, клиент-сервер. Локальная вычислительная сеть Ethernet. Трех уровневая организация. Средства подключения ВМ и ЛВС Ethernet. Способы доступа к среде. Формат кадра. Основные скорости передачи. Сеть Интернет. Стек протоколов TCP/IP. Уровни протоколов сети Интернет. Понятия FTP, SMTP, HTTP, TELNET, WWW. Способы подключения абонента к сети Интернет. Корпоративные сети.

2. ПРАКТИЧЕСКИЕ ЗАНЯТИЯ (18 часов)

2.1. Представление информации в вычислительных машинах – 2 часа.

Системы исчисления. Способы перевода целых чисел. Способы перевода дробных чисел. Представление положительных и отрицательных чисел, чисел с плавающей точкой. Правила сложения двоичных чисел. Перевод в машинные коды. Арифметические операции над числами с фиксированной и плавающей точкой. Арифметические операции над двоично-десятичными кодами чисел.

2.2. Основы алгебры логики – 4 часа.

Упрощение логических функций. Составление таблиц истинности по логическим функциям. Получение логических функций по таблицам истинности (СДНФ, СКНФ, карты Карно). Построение схемы по таблице истинности. Составление логической функции по схеме. Реализация в различных базисах.

2.3. Основы представления информации и алгебры логики – 2 часа.

Контрольная работа №1 (вопросы, см. п. 5.1)

2.4. Построение и применение простейших комбинационных устройств – 2 часа.

Синтез пороговой ячейки. Построение различных схем шифраторов и дешифраторов на базе простейших логических элементов. Реализация логических функций на мультиплексоре.

2.5. Комбинационные устройства средней степени интеграции – 2 часа.

Построение схемы дешифратора 3×8 из двух дешифраторов 2×4 . Реализация логических схем на основе дешифратора и логических элементов указанного типа. Построение цифрового устройства формирующего заданные комбинации двоичных чисел на своих выходах.

2.6. Комбинационные устройства средней степени интеграции – 2 часа.

Контрольная работа №2 (вопросы, см. п. 5.2)

2.7. Применение комбинационных устройств – 2 часа.

Синтез сегментных дисплеев различной конфигурации. Построение схемы выборки микросхем памяти (ОЗУ и ПЗУ) на основе дешифраторов различной структуры для микроконтроллера КР-580.

2.8. Последовательностные схемы – 2 часа.

Синтез цифровых устройств для обнаружения спада или установки сигнала на входе микросхемы. Разработка схем счетчиков с измененным коэффициентом пересчета. Техническое устранение иголок в выходном сигнале счетчиков с измененным коэффициентом пересчета.

3. ЛАБОРАТОРНЫЕ ЗАНЯТИЯ (18 часов).

3.1. Изучение базовых возможностей среды Electronics Workbench – 2 часа.

3.2. Изучение логических схем и функций – 2 часа.

3.3. Изучение работы шифраторов, дешифраторов и мультиплексоров – 4 часа.

3.4. Изучение работы триггеров – 4 часа.

3.5. Изучение сумматоров, полусумматоров, регистров и счетчиков – 4 часа.

3.6. Изучение сетевых возможностей Windows NT – 2 часа.

4. САМОСТОЯТЕЛЬНАЯ РАБОТА (55 часов).

Самостоятельная работа состоит в проработке ряда вопросов лекционного и практического курса в соответствии с учебно-методической картой дисциплины.

5. ПЕРЕЧЕНЬ И ТЕМЫ ПРОМЕЖУТОЧНЫХ ФОРМ КОНТРОЛЯ ЗНАНИЙ.

Промежуточный контроль знаний студентов по дисциплине предусматривает две контрольные точки, оценки по которым выставляются на основе информации о работе на практических занятиях, а также на основе выполнения двух контрольных работ. Предусмотрены контрольные работы со следующими темами заданий:

5.1. Перевод из одной системы исчисления в другую; сложение и вычитание чисел в различной системе исчисления; составление логических функций по таблице истинности; упрощение полученных логических функций; построение схемы в произвольном базисе по логическим функциям; реализация логических функций в базисе «И-НЕ», построение соответствующих схем; реализация логических функций в базисе «ИЛИ-НЕ», построение соответствующих схем; построение по заданной схеме логической функции.

5.2. Реализовать заданные логические функции на мультиплексоре (логические функции задаются в виде СДНФ, в произвольном виде); реализовать заданные логические функции на основе дешифратора и логических элементов заданного типа (логические элементы «И-НЕ», «ИЛИ-

HE»); разработать схему на основе дешифратора, формирующую «1» («0») на указанных выходах, при определенных комбинациях входных чисел.

6. ВОПРОСЫ К ЭКЗАМЕНУ.

1. Основные понятия о процессе автоматизированной обработки данных. Определения. Меры информации.
2. Основные понятия о процессе автоматизированной обработки данных. Показатели качества информации. Показатели качества и функционирование системы управления.
3. Принцип действия ВМ. Основы двоичной системы исчисления, виды систем исчисления, перевод из одной формы записи в другую.
4. Принцип действия ВМ. Основы алгебры логики. Базовые логические операции и логические элементы.
5. Принцип действия ВМ. Основные законы алгебры логики, виды записи логических выражений, переход от одной формы представления работы цифрового устройства к другой.
6. Принцип действия ВМ. Представление работы цифрового устройства в различных базисах.
7. Принцип действия ВМ. Комбинационные устройства средней степени интеграции: шифраторы, дешифраторы, мультиплексоры, сумматоры, полусумматоры.
8. Принцип действия ВМ. Простейшие последовательностные схемы: триггеры и основные элементы на триггерных схемах (счетчики, регистры)
9. Архитектура и классификация ВМ. Определения, основные характеристики ВМ.
10. Архитектура и классификация ВМ. Классификация ВМ по принципу действия, элементной базе, по назначению, по мощности. Основные области применения ВМ.
11. Общие понятия о функциональной и структурной организации ВМ. Обобщенная структура ВМ, ее подсистемы.
12. Аппаратные особенности ВМ первого и второго поколений. Структура простейшего АЛУ.
13. Аппаратные особенности ВМ третьего, четвертого и пятого поколений.
14. Основные принципы построения и функционирования ВМ. Кризис структуры фон-Неймана, ВМ шестого поколения.
15. Особенности организации современных ВМ. SISD компьютеры. CISC и RISC архитектура.
16. Особенности организации современных ВМ. Основы организации суперскалярной обработки данных.
17. Особенности организации современных ВМ. SIMD компьютеры. Матричная, векторно-конвейерная архитектура, MMX технология.

18. Организация микропроцессоров. Понятие микропроцессора, его основные особенности, преимущества использования. Функциональная структура МП.
19. Организация микропроцессоров. Предназначение, характеристика, состав и принцип работы операционного блока.
20. Организация микропроцессоров. Предназначение, характеристика, состав и принцип работы блока управления и интерфейсного блока.
21. Особенности организации процессоров при использовании внутренних регистров.
22. Система команд микропроцессора.
23. Основные понятия о способах и методах адресации.
24. Организация памяти ВМ. Общая структура и характеристики каждого уровня.
25. Организация памяти ВМ. Организация оперативного, сверх оперативного уровней. Методы управления оперативной памятью.
26. Системы внешней памяти и особенности организации кэш-памяти. Методы повышения пропускной способности оперативной памяти.
27. Организация обмена данными в ВМ – общие сведения. Обмен данными между периферийными устройствами и вычислительным ядром системы.
28. Персональные компьютеры, особенности архитектуры и применения. Функциональная и структурная организация, характеристики ПК.
29. Персональные компьютеры, особенности архитектуры и применения. Системная плата, внутримашинный и шинный интерфейсы, основные используемые шины.
30. Персональные компьютеры, особенности архитектуры и применения. Запоминающие устройства.
31. Проектирование микропроцессорных систем и средств автоматики на их основе. Уровни проектирования. Классификация ошибок, неисправностей, дефектов.
32. Проектирование микропроцессорных систем и средств автоматики на их основе. Обнаружение неисправностей, функции средств отладки.
33. Проектирование микропроцессорных систем и средств автоматики на их основе. Основные этапы проектирования, комплексная отладка.
34. Встроенные средства тестирования и отладки при работе системы.
35. Процесс разработки программного обеспечения: определения основные инструменты, процесс создания.
36. Особенности создания исполняемых программ на языке ассемблер.
37. Структура программного обеспечения ВМ – предназначение и основные характеристики. Системное программное обеспечение – классификация.
38. Системное обеспечение современных ВМ. Операционные системы, системы автоматизации программирования.
39. Комплекс технического обслуживания, пакеты программ дополняющих возможности и системы документации ВМ.

40. Специальное или прикладное обеспечение современных ВМ. Пакеты прикладных программ, различные системы обработки.
41. Централизованные и распределенные системы обработки данных. Вводные понятия и типовая структура.
42. Централизованные и распределенные системы обработки данных. Организация микроконтроллерных систем.
43. Типовая структура микроконтроллера, общие сведения.
44. Параллельная обработка данных как архитектурный способ повышения производительности.
45. Классификация систем параллельной обработки данных. Основные архитектуры многопроцессорных вычислительных систем.
46. Компьютеры класса MISD MIMD.
47. Многопроцессорные вычислительные системы (с общей шиной, с многовходовой памятью).
48. Многомашинные вычислительные системы: многомашинные комплексы, системы массового параллелизма.
49. Вычислительные системы – состояния производства, направления развития высокопроизводительных вычислительных систем, тенденции развития архитектур с общей и разделяемой памятью.
50. Вычислительные системы – развитие архитектур микропроцессоров, направления развития мультипроцессорных систем с распределенной памятью.
51. Общие оценки производительности мультипроцессорных систем при увеличении числа процессоров. Вычислительные системы на кристалле и нанотехнологии.
52. Принципы построения телекоммуникационных вычислительных систем: понятие, организация, параметры, классификация, архитектура.
53. Семиуровневая эталонная модель взаимодействия открытых систем (OSI).
54. Коммуникация и маршрутизация при передаче данных.
55. Локальные вычислительные сети и сеть Ethernet.
56. Корпоративные сети и сеть Интернет.

УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

1. ПЕРЕЧЕНЬ ОБЯЗАТЕЛЬНОЙ (ОСНОВНОЙ) ЛИТЕРАТУРЫ

1. Хомоненко А.Д. (ред.) Основы современных компьютерных технологий. Учеб.пособие.Рек.Мин.обр.РФ. – С-Пб.: Корона принт, 1998, 448 с. – 14 экз. (эн. фак, ч. 3.)
2. А.П. Пятибратов, Л.П. Гудыно, А.А. Кириченко. Вычислительные системы, сети и телекоммуникации. Учебник: Рек. Мин. обр. РФ. – М.: Финансы и статистика, 2004, 512 с. – 6 экз. (аб., ч.3.)

3. Н.Л. Прохоров, Г.А. Егоров, В.Е. Красовский и др. Управляющие вычислительные комплексы. Учеб. пособие: Рек. Мин. обр. РФ. – М.: Финансы и статистика, 2003, 352 с. – 6 экз. (аб., ч. 3.)

4. Б.Я. Цилькер, С.А. Орлов. Организация ЭВМ и систем. Учеб.: рек. Мин. обр. РФ. – Спб.: Питер, 2006, 668 с. – 1 экз. (аб)

5. А. В. Богданов [и др.] Архитектуры и топологии многопроцессорных вычислительных систем. Курс лекций: учеб. пособие: Рек. УМО вузов. – М.: Мир, 2006, 480 с. – 20 экз. (аб., ч. 3.)

6. В.Л. Бройдо. Вычислительные системы, сети и телекоммуникации. Учеб. пособие: доп. Мин. обр. РФ. – Спб.: Питер, 2005, 703 с. – 15 экз. (аб, ч.з.)

7. В. Ф. Малехин, Е. Г. Павловский. Вычислительные машины, системы и сети. Учеб.: рек. УМО. – М.: Издательский центр «Академия», 2006, 560 с. – 10 экз. (аб., ч.з.)

8. Теличенко Д. А., Романова М. В. Цифровые узлы и элементы организации вычислительных систем. Лабораторный практикум. Благовещенск: Амурский гос. ун-т, 2004, 104 с. – 50 экз. (5 – эн. фак., 40 – аб., 5 – ч. з.).

2. ПЕРЕЧЕНЬ ДОПОЛНИТЕЛЬНОЙ ЛИТЕРАТУРЫ

1. Поворознюк А.И. Архитектура компьютеров. Архитектура микропроцессорного ядра и системных устройств: Учеб пособие. Ч.1. – Харьков: Торнадо, 2004, 355 с.

2. Поворознюк А.И. Архитектура компьютеров. Архитектура внешней памяти, видеосистемы и внешних интерфейсов: Учеб пособие. Ч.2. – Харьков: Торнадо, 2004, 296 с.

3. Электротехника и электроника в экспериментах и упражнениях. Практикум на Electronics Workbench / Под общ. Ред. Д. И. Панфилова. – Т. 2 – Электроника. М.: Додека, 2000.

4. Теличенко Д. А., Бушманов А. В. Схемотехника. Лабораторный практикум. Благовещенск: Амурский гос. ун-т, 2006, 93 с. (рек. ДВ РУМНЦ).

УЧЕБНО-МЕТОДИЧЕСКАЯ (ТЕХНОЛОГИЧЕСКАЯ) КАРТА ДИСЦИПЛИНЫ

Учебно-методическая карта дисциплины «Вычислительные машины, системы и сети», соответствующая рабочей программе отражена в виде таблицы 1.

Здесь в графе 6, звездочкой помечено два используемых методических пособия:

а) *Теличенко Д. А., Романова М. В.* Цифровые узлы и элементы организации вычислительных систем. Лабораторный практикум. Благовещенск: Амурский гос. ун-т, 2004, 104 с.

б) *Теличенко Д. А., Бушманов А. В.* Схемотехника. Лабораторный практикум. Благовещенск: Амурский гос. ун-т, 2006, 93 с., (рекомендовано ДВ РУМНЦ).

Таблица 1 – Учебно-методическая карта дисциплины.

Номер недели	Номер темы	Вопросы, изучаемые на лекции	Занятия (номера)		Используемые нагляд. и метод. пособия	Самостоятельная работа студентов		Формы контроля
			практич. (семин.)	лаборат.		содержание	час.	
1	2	3	4	5	6	7	8	9
1	1	Введение. Основные понятия о процессе автоматизированной обработки данных	1		*	Представление информации в вычислительных машинах.	4	1-ая контрольная работа, экзамен, РГР
	2	Принцип действия ВМ. Логические основы, построение и работа простейших цифровых устройств	2 – 9	1 – 4	*	Получение логических функций.		защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
2	3	Архитектура ВМ. Классификация ВМ и история развития вычислительной техники				Получение логических функций.	3	1-ая контрольная работа, экзамен, РГР
3	4	Архитектура ВМ. Функциональная и структурная организация ВМ				Получение логических функций.	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
	5	Аппаратные особенности ВМ различных поколений. Принцип построения и функционирования ВМ пятого поколения						
4	6	Особенности организации современных однопроцессорных ВМ				Построение структурных схем	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
5	7	Организация микропроцессоров. Функциональная структура МП				Построение структурных схем	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
	8	Особенности организации процессоров при использовании внутренних регистров						
6	9	Система команд				Построение структурных схем	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
7	10	Способы адресации				Составление логических функций	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
	11	Особенности организация памяти ВМ – часть 1						

Продолжение таблицы 1.

1	2	3	4	5	6	7	8	9
8	11	Особенности организация памяти ВМ – часть 2				Составление логических функций	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
9	12	Организация обмена данными в ВМ				Составление логических функций	3	защита лабораторной работы № 1, 2-ая контрольная работа, экзамен, РГР
	13	Персональные компьютеры (ПК), особенности архитектуры и применения – часть 1						
10	13	Персональные компьютеры (ПК), особенности архитектуры и применения – часть 2				Применение дешифраторов	3	защита лабораторной работы № 2, 2-ая контрольная работа, экзамен, РГР
11	13	Персональные компьютеры (ПК), особенности архитектуры и применения – часть 3				Применение дешифраторов	3	защита лабораторной работы № 2, 2-ая контрольная работа, экзамен, РГР
	14	Проектирование микропроцессорных систем и средств автоматики на их основе – часть 1						
12	14	Проектирование микропроцессорных систем и средств автоматики на их основе – часть 2				Применение дешифраторов	3	защита лабораторной работы № 2, 2-ая контрольная работа, экзамен, РГР
13	15	Системное и прикладное программное обеспечение современных ВМ и МПС. Интерфейс пользователя – часть 1		5	*	Применение мультиплексора	3	защита лабораторной работы № 2, 2-ая контрольная работа, экзамен, РГР
		Системное и прикладное программное обеспечение современных ВМ и МПС. Интерфейс пользователя – часть 2		5	*			
14	16	Централизованные и распределенные системы обработки данных.				Применение мультиплексора	3	защита лабораторной работы № 2, 2-ая контрольная работа, экзамен, РГР

Продолжение таблицы 1.

1	2	3	4	5	6	7	8	9
15	17	Вычислительные системы параллельной обработки данных – часть 1				Применение мультимплексора	3	защита лабораторной работы № 2, экзамен, РГР
		Вычислительные системы параллельной обработки данных – часть 2						
16	18	Вычислительные системы – состояние, производительность, направления развития				Построение временных диаграмм	3	защита лабораторной работы № 3, экзамен, РГР
17	19	Телекоммуникационные вычислительные сети – часть 1		5	*	Построение временных диаграмм	3	защита лабораторной работы № 3, экзамен, РГР
		Телекоммуникационные вычислительные сети – часть 2		5	*			
18	19	Телекоммуникационные вычислительные сети – часть 3		5	*	Построение временных диаграмм	3	защита лабораторной работы № 3, экзамен, РГР

2. ГРАФИК САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

График самостоятельной работы студентов (соответствующий учебной программе и учебно-методической карте, см. табл.1) по дисциплине «Вычислительные машины системы и сети», отражен в виде таблицы 2.

Таблица 2 – График самостоятельной работы студентов.

Самостоятельная работа студентов		Формы контроля
содержание	час.	
1	2	3
Представление информации в вычислительных машинах (системы исчисления, перевод, арифметические операции)	4	1-ая контрольная работа, экзамен, РГР
Получение логических функций (по таблице истинности, от 2-х переменных)		защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
Получение логических функций (по таблице истинности, от 3-х переменных)	3	1-ая контрольная работа, экзамен, РГР
Получение логических функций (по таблице истинности, от 4-х переменных)	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
Построение структурных схем (по логической функции от 2-х переменных)	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
Построение структурных схем (по логической функции от 3-х переменных)	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
Построение структурных схем (по логической функции от 4-х переменных)	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
Составление логических функций (по схеме из 4-х элементов)	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
Составление логических функций (по схеме из 5-и элементов)	3	защита лабораторной работы № 1, 1-ая контрольная работа, экзамен, РГР
Составление логических функций (по схеме из 6-и элементов)	3	защита лабораторной работы № 1, 2-ая контрольная работа, экзамен, РГР
Применение дешифраторов (для реализации логических функций в произвольном виде)	3	защита лабораторной работы № 2, 2-ая контрольная работа, экзамен, РГР
Применение дешифраторов (для реализации логических функций в базисе «И-НЕ», «ИЛИ-НЕ»)	3	защита лабораторной работы № 2, 2-ая контрольная работа, экзамен, РГР
Применение дешифраторов (для формирования сигнала выбора микросхемы)	3	защита лабораторной работы № 2, 2-ая контрольная работа, экзамен, РГР
Применение мультиплексора (для реализации логических функций в виде СДНФ)	3	защита лабораторной работы № 2, 2-ая контрольная работа, экзамен, РГР
Применение мультиплексора (для реализации логических функций в виде СКНФ)	3	защита лабораторной работы № 2, 2-ая контрольная работа, экзамен, РГР
Применение мультиплексора (для реализации логических функций в произвольном базисе)	3	защита лабораторной работы № 2, экзамен, РГР
Построение временных диаграмм (для простейших асинхронных элементов)	3	защита лабораторной работы № 3, экзамен, РГР
Построение временных диаграмм (для синхронных комбинационных схем)	3	защита лабораторной работы № 3, экзамен, РГР
Построение временных диаграмм (для последовательностных устройств)	3	защита лабораторной работы № 3, экзамен, РГР

3. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ПРОВЕДЕНИЮ ПРАКТИЧЕСКИХ ЗАНЯТИЙ И САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ (формы проведения)

3.1. ПРАКТИЧЕСКИЕ ЗАНЯТИЯ

Предварительно студенты знакомятся со списком всех изучаемых тем, рассматриваемых на практических занятиях. Характер вопросов, прорабатываемых здесь, связан с лекционным курсом и графиком самостоятельной работы студентов. Большая часть времени уделяется решению конкретных задач (аналогичные по тематике задачи использованы в качестве третьего экзаменационного вопроса).

Задачи решаются студентами самостоятельно. При этом один из студентов вызывается к доске, решает поставленную задачу. Преподавателем контролируется не только правильность решения, но и даются: практические рекомендации по решению подобных заданий, применимость рассматриваемых тем к практике, а так же предлагается другим студентам предложить аналогичные способы решения. Каждому вышедшему к доске, а так же студентам принявшим участие в обсуждении выставляется оценка.

Помимо этого, персонально каждый студент на практических занятиях выполняет две контрольные работы (см. п.12.2).

3.2. САМОСТОЯТЕЛЬНАЯ РАБОТА

Самостоятельная работа состоит в проработке ряда вопросов лекционного и практического курса в соответствии с учебно-методической картой дисциплины (см. п.2, табл. 1,2). При этом студентам предлагается законспектировать рассматриваемый вопрос, в случае необходимости задать возникшие вопросы на практическом занятии.

Основной формой контроля проработки материала является опрос, проводимый при допуске к лабораторной работе, а так же результаты двух контрольных работ. Вопросы, прорабатываемые студентами самостоятельно, включены в экзаменационные билеты (см. п.14).

4. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ПРОВЕДЕНИЮ ЛАБОРАТОРНЫХ РАБОТ (формы проведения)

Лабораторный практикум представляет собой пять лабораторных работ, выполняемых по указанным в п.5 методическим пособиям. Каждая работа содержит необходимые теоретические сведения по исследуемой теме, задания для выполнения и контрольные вопросы. Помимо этого в ряде работ приведены упражнения для более глубокого освоения студентами изучаемого материала (данный комплект упражнения является расчетно-графической работой каждого студента).

Для выполнения лабораторных работ студентам предварительно предлагается самостоятельно ознакомиться с краткой теорией к каждой выполняемой работе. Это даст необходимую теоретическую основу и облегчит выполнение работ, позволив на занятии уделить большее внимание вопросам, обычно вызывающим наибольшее затруднение. Сами задания в лабораторном практикуме расположены в возрастающем порядке сложности. По мере того как вырастает объем работ, которые студенты выполняют в аудитории, уменьшается количество заданий для самостоятельного решения (упражнений).

Все занятия делятся на два цикла: выполнение работы, защита работы. Циклы повторяются для каждой работы, в порядке следования, без нарушения очередности. Для улучшения качества усваиваемого материала не рекомендуется: совмещать в рамках проведения одного цикла разные темы исследования; проводить одновременное снятие и защиту работы. Допускается после выполнения очередного цикла всеми студентами группы в случае оставшегося времени уделить время на ликвидацию образовавшихся задолжностей, если таковые имеются.

На вводном занятии: студенты группы делятся на бригады по два – три человека, им присваиваются варианты, номера которых сохраняются за ними на протяжении всего курса. Каждый из студентов имеет два варианта: первый – личный (для заданий, требующих самостоятельного решения), второй – вариант на бригаду (для заданий, допускающих групповое выполнение).

На первом этапе цикла (снятия работы):

- преподавателем осуществляется допуск к работе, на котором проверяется: знание студентов краткой теории по выполняемой работе; наличие заготовки отчета;
- выясняются вопросы, вызвавшие у студентов затруднения, даются необходимые пояснения по ним;
- даются комментарии по методике проведения экспериментов;
- контролируется выполнение работы каждой бригады и всеми студентами в целом.

Работа считается снятой, если: студенты одной бригады, и каждый в отдельности, выполнили все задания работы, согласно вариантам; зафиксировали снятые данные в заготовку отчета.

На втором этапе цикла (защита работы):

- преподавателем, каждому из студентов, выдается произвольный вариант необходимый для выполнения упражнений (для лабораторных работ 1, 2, 3);
- каждый из студентов лично выполняет упражнения, согласно выданному на данной работе варианту (в случае если работа не защищается на одном занятии, варианты на упражнения изменяются);
- преподавателем проверяется личный отчет каждого из студентов, задаются вопросы по ходу выполнения работы; задаются контрольные вопросы (список вопросов приведен в лабораторном практикуме к каждой работе).

Работа считается защищенной, если: правильно выполнен отчет по работе; даны корректные ответы на вопросы преподавателя; правильно выполнены упражнения.

Представляемый отчет (после успешной защиты работы отчет сдается преподавателю и сохраняется до успешной сдачи студентом экзамена) должен удовлетворять следующим требованиям:

- отчет выполняется на одной стороне белого листа формата А4 в рукописной или печатной форме, в варианте возможном для прочтения (почерк, шрифт, размер, интервал);
- титульный лист должен содержать следующие сведения: название предмета; тему работы, с ее порядковым номером; фамилию студента выполнившего работу с указанием номера группы и вариантов (личного и на бригаду); фамилию преподавателя, осуществляющего прием работы; дату снятия и защиты (дата защиты заполняется преподавателем лично).
- основная часть работы должна содержать следующие сведения: краткую теорию; цель работы; элементы, приборы и инструменты, используемые в работе; ход работы с необходимыми рисунками, схемами, таблицами и формулами (необходимый перечень приведен в лабораторном практикуме по каждой работе).

В случае если студент не снял или не защитил работу, он может приступить к следующей работе. Ликвидировать возникшую задолженность можно на оставшемся времени после проведения очередной лабораторной работы или на дополнительных занятиях. Если ликвидировать задолженность по лабораторным работам в течение семестра не удастся, студент является на экзамен с отчетами по несданным работам, где ему до ответа на экзаменационные вопросы дается возможность защитить каждую работу.

5. СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

Полный список рекомендованной (основной и дополнительной литературы) с указанием доступного количества экземпляров в библиотеке университета приведен ниже

УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

1. ПЕРЕЧЕНЬ ОБЯЗАТЕЛЬНОЙ (ОСНОВНОЙ) ЛИТЕРАТУРЫ

1. Хомоненко А.Д. (ред.) Основы современных компьютерных технологий. Учеб.пособие.Рек.Мин.обр.РФ. – С-Пб.: Корона принт, 1998, 448 с. – 14 экз. (эн. фак, ч. 3.)

2. А.П. Пятибратов, Л.П. Гудыно, А.А. Кириченко. Вычислительные системы, сети и телекоммуникации. Учебник: Рек. Мин. обр. РФ. – М.: Финансы и статистика, 2004, 512 с. – 6 экз. (аб., ч.з.)

3. Н.Л. Прохоров, Г.А. Егоров, В.Е. Красовский и др. Управляющие вычислительные комплексы. Учеб. пособие: Рек. Мин. обр. РФ. – М.: Финансы и статистика, 2003, 352 с. – 6 экз. (аб., ч. 3.)

4. Б.Я. Цилькер, С.А. Орлов. Организация ЭВМ и систем. Учеб.: рек. Мин. обр. РФ. – Спб.: Питер, 2006, 668 с. – 1 экз. (аб)

5. А. В. Богданов [и др.] Архитектуры и топологии многопроцессорных вычислительных систем. Курс лекций: учеб. пособие: Рек. УМО вузов. – М.: Мир, 2006, 480 с. – 20 экз. (аб., ч. 3.)

6. В.Л. Бройдо. Вычислительные системы, сети и телекоммуникации. Учеб. пособие: доп. Мин. обр. РФ. – Спб.: Питер, 2005, 703 с. – 15 экз. (аб, ч.з.)

7. В. Ф. Малехин, Е. Г. Павловский. Вычислительные машины, системы и сети. Учеб.: рек. УМО. – М.: Издательский центр «Академия», 2006, 560 с. – 10 экз. (аб., ч.з.)

8. Теличенко Д. А., Романова М. В. Цифровые узлы и элементы организации вычислительных систем. Лабораторный практикум. Благовещенск: Амурский гос. ун-т, 2004, 104 с. – 50 экз. (5 – эн. фак., 40 – аб., 5 – ч. 3.).

2. ПЕРЕЧЕНЬ ДОПОЛНИТЕЛЬНОЙ ЛИТЕРАТУРЫ

1. Поворознюк А.И. Архитектура компьютеров. Архитектура микропроцессорного ядра и системных устройств: Учеб пособие. Ч.1. – Харьков: Торнадо, 2004, 355 с.

2. Поворознюк А.И. Архитектура компьютеров. Архитектура внешней памяти, видеосистемы и внешних интерфейсов: Учеб пособие. Ч.2. – Харьков: Торнадо, 2004, 296 с.

3. Электротехника и электроника в экспериментах и упражнениях. Практикум на Electronics Workbench / Под общ. Ред. Д. И. Панфилова. – Т. 2 – Электроника. М.: Додека, 2000.

4. Теличенко Д. А., Бушманов А. В. Схемотехника. Лабораторный практикум. Благовещенск: Амурский гос. ун-т, 2006, 93 с. (рек. ДВ РУМНЦ).

6. ПЛАН КОНСПЕКТ ЛЕКЦИЙ ПО КАЖДОЙ ТЕМЕ

ГЛАВА 1

Основные понятия, принципы построения, архитектура ВМ и ее особенности

Лекция №1 «Введение. Основные понятия о процессе автоматизированной обработки данных.»

Понятие информации, определение, особые свойства информации.

Адекватность информации: синтаксическая, семантическая, прагматическая.

Меры информации (определение, формулы, примеры): объем данных, количество переданной информации, априорная и апостериорная информация, энтропия, степень информативности, тезаурус, полезность информации, экономический эффект.

Показатели качества информации (определение, формулы, примеры): репрезентативность, содержательность, достаточность, доступность, актуальность, своевременность, точность, достоверность, ценность.

Показатели качества информации и функционирование системы управления.

Выводы: выделение данных как одной из основных форм представления информации изучаемых в курсе дисциплины; ключевой режим работы элементарных узлов цифровой вычислительной техники; основная задача обработки данных; важнейшая характеристика информации – адекватность; принятие принципа имеющейся прагматической ценности для дальнейшего рассмотрения информации обрабатываемой вычислительной системой.

Заключение: понятие автоматической и автоматизированной системы; основные тенденции развития современных информационных технологий; характеристика увеличения в геометрической прогрессии объема, скорости, достоверности обрабатываемых данных современными вычислительными системами; основные тенденции развития и недоступные для вычислительных систем области деятельности человека.

Лекция №2 «Принцип действия ВМ. Логические основы, построение и работа простейших цифровых устройств – обзорная лекция.»

Системы исчисления (позиционные и непозиционные). Перевод чисел из одной формы записи в другую.

Представление информации в ВМ (числа со знаком, с точкой, целые). Представление других видов информации.

Арифметические основы двоичной системы исчисления: правила сложения и вычитания двоичных чисел. Прямой и обратный коды двоичного числа. Умножение и деление.

Логические основы построения ВМ. Принципы и формы описания.

Алгебра логики – основные понятия и базовые операции.

Формы представления цифровых устройств (логическая функция, таблица истинности, схемное представление). Обозначения, принятые в нашей стране и за рубежом.

Основные законы алгебры логики. Минимизация логических функций и выражений. Понятие базиса, переход от одной формы представления к другой.

Простейшие цифровые узлы. Определение комбинационных схем. Комбинационные схемы средней степени интеграции (шифраторы, дешифраторы, мультиплексоры, сумматоры и вычитатели) – принцип работы, применение, способы представления.

Последовательностные схемы (схемы с памятью). Определение, предназначение. Триггеры и счетчики (виды, характеристики, основные достоинства и недостатки, различные схемы, временные диаграммы).

Осиновые принципы проектирования цифровых устройств. Этапы проектирования. Примеры применения цифровых устройств средней и малой степени интеграции. Современные тенденции развития.

Лекция №3 «Архитектура ВМ. Классификация ВМ и история развития вычислительной техники.»

Определение ВМ, ее структуры и архитектуры. Понятие архитектуры ВМ с точки зрения ее характеристик.

Быстродействие и производительность ВМ – понятие, примеры.

Характеристики ВМ: емкость запоминающих устройств, надежность, точность, достоверность – понятие, примеры.

Классификация средств электронно-вычислительной техники, принципы, определения, примеры:

- по принципу действия (аналоговые цифровые, гибридные)
- по этапам создания и элементной базе (первое, второе, третье, четвертое, пятое и шестое поколения)
- по назначению (универсальные, проблемно-ориентированные, специализированные)
- по размерам и вычислительной мощности (СуперЭВМ, Большие ЭВМ, Малые ЭВМ, МикроЭВМ – универсальные, специализированные, персональные).

Замечания по предложенным выше способам классификации ВМ: быстрое устаревание, необходимость регулярной корректировки. Выделение общих закономерностей развития.

Принцип академика В.М. Глушкова. ВМ с точки зрения использования человеком. Первое направление – применение ВМ для автоматизации вычислений. Второе направление – применение ВМ в системах управления. Третье направление – применением ВМ для решения задач, имеющих статистический и вероятностный подход, т.е. получение не точного результата, а чаще всего осредненного в статистическом, вероятностном смысле.

Лекция №4 «Архитектура ВМ. Функциональная и структурная организация ВМ»

1. Общие понятия о функциональной и структурной организации ВМ.

Различные точки зрения на функционирование вычислительного процесса. Точка зрения пользователя, определение функциональной организации ВМ. Точка зрения программиста, определение структурной организации ВМ, степени детальности структурных схем (элемент, узел, блок, устройство).

Обобщенная структура ВМ. Схема обобщенной структуры, включающая обрабатывающую подсистему; подсистему памяти; подсистему ввода-вывода; подсистему управления и обслуживания.

Характеристики каждой из подсистем обобщенной структуры и ее особенности:

- обрабатывающая подсистема (состав, предназначение, способы повышения производительности);
- подсистема памяти (иерархическая структура – сверх оперативный, оперативный и внешний уровень, предназначение, эффективные методы повышения производительности);
- подсистема ввода-вывода (состав, задачи выполняемые устройствами ввода-вывода, основные направления повышения производительности);
- подсистема управления и обслуживания (состав, предназначение, способы повышения производительности).

2. Структура и состав ВМ

Концепция Дж. Фон Неймана. Устройство управления, арифметическо-логическое устройство, память и устройство ввода-вывода. Потоки данных, команд и управляющих сигналов. Принципы построения ВМ: двоичное кодирование информации, линейно-адресная организация памяти, программа, команды, код операции, адреса операндов, единая память, вычислительный процесс, жесткость архитектуры, процессор.

Основные принципы взаимодействия между элементами вычислительной системы и основы их функционирования.

Лекция №5 «Аппаратные особенности ВМ различных поколений. Принцип построения и функционирования ВМ пятого поколения»

Вычислительные машины первого и второго поколений: структура, основные элементы (устройство ввода-вывода, оперативное запоминающее устройство, арифметическо-логическое устройство, устройство управления, внешние запоминающие устройства), принципы взаимодействия, особенности построения. Структура простейшего АЛУ.

Вычислительные машины третьего поколения: структура, основные элементы (процессор, устройство ввода-вывода, оперативное запоминающее устройство, внешние запоминающие устройства, каналы ввода-вывода), принципы взаимодействия, особенности построения.

Вычислительные машины четвертого и пятого поколений: структура, основные элементы (процессор, оперативная память, шина, контроллер пря-

мого доступа к памяти, контроллеры внешних устройств, таймер), принципы взаимодействия, особенности построения.

Основные принципы построения ВМ пятого поколения: модульность, иерархическая организация, децентрализация управления.

Общие принципы функционирования ВМ пятого поколения.

Кризис структуры Дж. Фон Неймана.

Вычислительные машины шестого поколения.

Лекция №6 «Особенности организации современных однопроцессорных ВМ»

Понятие однопроцессорных и многопроцессорных систем. Таксономия М. Флина.

1. SISD (ОКОД)-компьютеры: определение, характеристика, основные элементы, структура, принципы функционирования.

1.1 CISC архитектура

1.2 RISC архитектура

1.3 Суперскалярная обработка: аппаратная реализация, VLIW архитектура.

2. SIMD (ОКМД)-компьютеры: определение, характеристика, основные элементы, структура, принципы функционирования.

2.1 Матричная архитектура

2.2 Векторно-конвейерная архитектура

2.3 MMX технология

ГЛАВА 2

Основы построения и работы микропроцессорных систем (МПС).

Лекция №7 «Организация микропроцессоров. Функциональная структура МП»

Процессор: предназначение, понятие микропроцессора, основные достоинства, направление применения.

Структурно-функциональная схема микропроцессора:

Операционный блок – состав (АЛУ, буферные регистры, регистр флагов, регистры общего назначения, аккумулятор), предназначение основных элементов, основные операции, специализированные блоки аппаратного умножения и деления, разрядность микропроцессора, понятие специализации регистров (аккумуляторная и многоаккумуляторная архитектура), примеры, особенности изменения содержимого регистра словосостояния.

Блок управления – предназначение, выполняемые действия (считывание и запоминание текущей команды, формирование адреса следующей команды, реализация потактового выполнения поступившей команды, управление обменом информацией с внешними устройствами), состав (регистр команд, дешифратор команд, блок формирования управляющих сигналов, программный счетчик, указатель стека), особенности функционирования, фазы выполнения команд (выборка, дешифрирование, выполнение), структура команд, микрокоманды, микрооперации, понятие микропрограммного автомата

с мягкой и жесткой логикой, особенности современной реализации элементов блока управления.

Интерфейсный блок – предназначение, понятие интерфейса ввода-вывода, функции интерфейсного блока, понятие системной шины и электрической спецификации сигналов на шине, цикл шины, принцип квитирования.

Лекция №8 «Особенности организации процессоров при использовании внутренних регистров»

Аккумуляторная архитектура – основы построения, принцип работы, характеристики, примеры.

Много аккумуляторная архитектура – основы построения, принцип работы, характеристики, примеры.

Стековые процессоры. Понятие стека, дисциплина FIFO, LIFO. Принцип работы стека и его предназначение. Адресация в стеке. Примеры работы со стеком. Характеристики стека. Особенности стековых процессоров, их структура. Пример стекового процессора на базе специализированного арифметического сопроцессора для вычислений с плавающей точкой.

Лекция №9 «Система команд»

Система команд как одна из важнейших характеристик микропроцессора. Понятие системы команд (форматы команд и обрабатываемых данных, список команд и их функциональное назначение, способы адресации данных).

1. Группы команд по функциональным признакам (предназначение, принцип работы, примеры, особенности использования):

- команды пересылок данных и ввода-вывода;
- арифметические и поразрядные двоичные команды;
- команды передачи управления.

Замечания по системе команд современных микропроцессоров.

2. Структура команд: операционная и адресная часть, их предназначение, характеристика. Особенности адресной части команд. Естественный и принудительный способ адресации команд. Примеры и пояснения.

Лекция №10 «Способы адресации»

1. Общие сведения по способам адресации.

2. Адресация данных.

2.1 Прямая адресация, принцип работы и особенности. Прямая регистровая адресация. Примеры.

2.2 Непосредственная адресация – принцип, особенности, примеры.

2.3. Неявная адресация – принцип, особенности, примеры.

2.4. Косвенная адресация – принцип, особенности, примеры. Особенности и преимущества косвенной адресации на примере организации цикла.

2.5. Относительная адресация или базирование – принцип, особенности, примеры. Формирование исполнительного адреса. Страничная организация и

сегментированная память. Базовая и индексная адресация (особенности, принцип, примеры)

3. Адресация команд – заключительные замечания.

Лекция №11, 12 «Особенности организация памяти ВМ»

Понятие памяти ВМ, характеристики отдельных устройств памяти. Быстродействие памяти, время доступа к памяти, длительность цикла памяти. Противоречивость требований к увеличению емкости и быстродействию памяти.

1. Уровни памяти (наименование, общая характеристика, состав, предназначение)

Сверх оперативный уровень;

Оперативный уровень;

Внешний уровень.

Замечания по производительности ВМ и особенностям организации памяти

2. Организация внутренней памяти процессора (сверхоперативный уровень), особенности, ссылка на ранее рассмотренный в лекциях материал.

3. Организация оперативной памяти (оперативный уровень)

- базовые типы оперативной памяти (DRAM, SRAM) принцип работы, особенности, сравнительная характеристика

- методы управления оперативной памятью: методы управления без использования виртуальной памяти (распределение памяти фиксированными разделами, распределение памяти динамическими разделами, распределение памяти с перемещаемыми разделами); методы управления с использованием виртуальной памяти (понятие виртуальной памяти, задачи виртуальной памяти, страничное распределение, сегментное распределение, странично-сегментное распределение, свопинг)

4. Системы внешней памяти: жесткие диски, гибкие магнитные диски, CD, DVD, новые форматы записи, flash.

5. Заключение:

Методы организации кэш памяти, ее структура и принцип работы.

Способы размещения данных в кэш памяти (прямое распределение, полностью ассоциативное распределение, частично ассоциативное распределение, распределение секторов)

Методы обновления строк в основной памяти, связь с кэш-памятью

Методы повышения пропускной способности оперативной памяти (выборка широким словом, расслоение сообщений, и т.п.)

Лекция №13 «Организация обмена данными в ВМ»

1. Общие сведения. Принципы организации обмена (подчиненность, квитиование, унификация характеристик).

2. Обмен данными между периферийными устройствами и вычислительным ядром системы. Особенности организации.

Программно-управляемая передача (синхронный и асинхронный обмен, специальные средства обмена, контроллеры периферийных устройств, режим ввода информации, режим вывода информации, сигнал готовности и его предназначение, преимущества и недостатки такой передачи, понятие таймаута).

Передача информации с прерыванием программы. Понятие прерывания аппаратные и программные прерывания. Сигнал запроса прерывания. Работа системы при реакции на прерывания.

Сравнения и выводы по программно-управляемой передаче и передаче с прерыванием.

Передача информации в режиме прямого доступа к памяти (ПДП). Определение режима ПДП. Предназначение режима ПДП. Достоинства и недостатки. Способы организации, примеры.

Лекция №14,15 «Персональные компьютеры (ПК), особенности архитектуры и применения» (микропроцессор, системная плата, системный и периферийный интерфейс).»

1. Функциональная и структурная организация класса ПК:

Микропроцессор, особенности, современные варианты выполнения. Разрядность, адресное пространство. Рабочая тактовая частота. Состав инструкций, конструктив, рабочее напряжение. Конвейерность, многозадачность работы, защищенный режим. Система виртуальных машин. Динамическое исполнение команд.

Особенности системной шины ПК.

Особенности основной памяти

Особенности внешней памяти

Источник питания и таймер.

Внешние устройства (их особенности): диалоговые средства пользователя, устройства ввода-вывода, устройства связи и телекоммуникации.

Дополнительные интегральные схемы: математический сопроцессор, контроллер DMA, сопроцессор ввода-вывода, контроллер прерываний.

Функциональные характеристики ПК.

2. Системная плата:

Состав и предназначение основных модулей, основные микросхемы.

Различные типы системных плат, тактовая частота системной шины, процессорные разъемы, понятие чипсета.

Северный и южный мост, разъемы для подключения памяти и их особенности.

Понятие BIOS, CMOS памяти.

3. Системный и периферийный интерфейс

Определение интерфейса, его состав.

Шины расширений (ISA, PC/XT, PC/AT, EISA, MCA) – предназначение, характеристики, примеры использования;

Локальные шины (VLB, PCI, AGP) – предназначение, характеристики, примеры использования;

Периферийные шины (IDE, ATA, ATA-2, SATA, ATAPI, SCSI) – предназначение, характеристики, примеры использования, протоколы взаимодействия. Интерфейс RS-232 и стандарт IEEE 1284, последовательные и параллельные порты;

Универсальные последовательные периферийные шины. Универсальная шина USB. Технология Bluetooth. Стандарт IEEE 1394, цифровой последовательный интерфейс Fire Wire. Шина PCMCIA. Расширенный интерфейс ACPI.

Лекция №16 «Персональные компьютеры (ПК), особенности архитектуры и применения» (запоминающие устройства)»

1. Микропроцессорная память и кэш память в ПК – особенности (в сравнении с ранее рассмотренными вопросами).

2. Основная память ПК:

2.1 Общие замечания по организации: RAM и ROM, сравнение и особенности SRAM и DRAM. Сигналы управления RAS и CAS.

2.2 Физическая структура основной памяти. Матричная организация памяти в ПК. Кодовые шины адреса, дешифраторы полуадресов, сигналы записи/считывания, регистр данных. Куб памяти.

2.3 Типы модулей оперативной памяти: DIP, SIP, SIPP, SIMM, DIMM, RIM – особенности, основные области применения, сравнительные характеристики, основные частоты и пропускная способность.

2.4 Типы оперативной памяти: FPM DRAM, RAM EDO, BEDO DRAM, SDRAM, DDR SDRAM, DRDRAM – особенности, виды, характеристики, одноканальные и многоканальные модули, разрядность, тактовая частота, пиковая пропускная способность.

3. Постоянные запоминающие устройства ПК: постоянные запоминающие устройства (ПЗУ), программируемые запоминающие устройства (ППЗУ); перезаписываемые запоминающие устройства (EEPROM) – принципы работы, предназначение, особенности использования и реализации.

4. Внешние запоминающие устройства ПК – особенности. Понятие файловой системы. Режимы обмена между внешней памятью и оперативной памятью. Время доступа к информации в ПК. Технология SMART.

5. Особенности Флэш-памяти ПК: принцип работы, характеристики, надежность, конструктивные варианты исполнения: ATA Flash, (PC Card ATA), Compact Flash (CF), Smart Media (SM), xD-Picture, MultiMedia Card (MMC), Secure Digital Card (SD), Miniature Card (ViniCard), Memory Stick.

6. Дисковые массивы RAID. Особенности использования в ПК. Уровни конфигурации RAID. Дисковые массивы различных поколений.

Лекция №17, 18 «Проектирование микропроцессорных систем и средств автоматизации на их основе»

1. Проектирование.

Уровни представления микропроцессорной системы: «черный ящик», структурный, программный, логический, схемный – особенности, использование, характеристики.

Разработка систем: ошибки, неисправности, дефекты. Основы тестового контроля. Субъективные неисправности, проектные неисправности. Ошибки – понятие, примеры. Дефекты – понятие, примеры.

Процедура отладки. Основные методы испытаний.

Обнаружение ошибки и диагностика неисправности. Лабораторные испытания, уровни обнаружения.

Свойство контролепригодности системы.

Функции средств отладки.

Этапы проектирования микропроцессорных систем и средств автоматики на их основе: этап 1 (формализация требований к системе), этап 2 (разработка структуры и архитектуры системы), этап 3 (разработка и изготовление аппаратных средств и программного обеспечения), этап 4 (комплексная отладка и приемочные испытания).

Источники возникновения ошибок на соответствующих этапах проектирования.

Проверка правильности проекта.

Автономная отладка аппаратного обеспечения.

Отладка программ.

Комплексная отладка микропроцессорной системы.

2. Встроенные средства тестирования и отладки при работе системы: состав, предназначение, примеры. JTAG-интерфейс.

3. Разработка программного обеспечения:

Понятие алгоритма. Машинно-ориентированные, процедурно-ориентированные и проблемно-ориентированные языки программирования.

Трансляторы (компиляторы, интерпретаторы)

Некоторые аспекты создания исполняемых программ на языке низкого уровня (Ассемблер):

- создание исходного файла;
- объектный модуль;
- исполняемый файл.

Процедуры формирования программы: составление, трансляция, листинг, компоновка объектного модуля, отладка программы, выполнение программы.

Структура программы на языке ассемблер для получения исполняемого файла формата EXE

Основные сведения о листинге и его структуре.

Краткие сведения об отладчике программ DEBUG.

Лекция №19, 20 «Системное и прикладное программное обеспечение современных ВМ и МПС. Интерфейс пользователя»

1. Структура программного обеспечения: общее, или системное (general Software), и специальное, или прикладное (application or special Software) – предназначение и основные характеристики.

2. Общее (системное) программное обеспечение: операционные системы, система автоматизации программирования, комплекс технического обслуживания, пакеты программ дополняющих возможности операционной системы, системы документации.

2.1 Операционные системы – цели применения, набор программных модулей. Основные примеры операционных систем и их особенности (DOS, OS/2, UNIX, Windows). Структура DOS: программа начальной загрузки, базовая система ввода-вывода (постоянный модуль, модуль расширения), базовый модуль DOS, командный процессор, утилиты.

2.2. Системы автоматизации программирования (инструментальные программные средства): языки программирования, языковые трансляторы, редакторы, средства отладки и другие вспомогательные программы – обзор основные особенности.

2.3. Комплекс технического обслуживания: проверочные тесты, наладочные тесты, диагностические тесты.

2.4. Пакеты программ дополняющих возможности и системы документации.

3. Специальное или прикладное обеспечение: пакеты прикладных программ: предназначение, и основные характеристики. Текстовые процессоры, редакторы широкого назначения, издательские системы, системы обработки электронных таблиц и табличные процессоры, графические редакторы, системы управления базами данных, графические редакторы, интегрированные системы.

ГЛАВА 3

Организация вычислительных систем и сетей.

Лекция №21 «Централизованные и распределенные системы обработки данных»

1. Понятие о централизованных и распределенных системах обработки данных и системах реального времени.

Обобщенная структура типовой системы управления (микроконтроллера), ее состав. Объект управления исполнительные устройства, система датчиков, устройства сопряжения с объектом, пульт управления, микропроцессорный (цифровой регулятор). Реальное время протекания процесса, шаг квантования.

Иерархическая организация системы управления сложными, распределенными в пространстве объектами – двух и трех уровневая модель.

2. Организация микроконтроллерных систем. Встраиваемые системы управления. Централизованная и распределенная система. Отличительные

черты микропроцессорной организации цифровых регуляторов. Встраиваемые средства на базе микроконтроллеров – функции, способы организации. Типы микроконтроллерных систем: автономная, локальная, сетевая конфигурация.

3. Типовая структура микроконтроллера – общие сведения. Основные типы и семейства микроконтроллеров. Базовые принципы организации, состав, основные модули.

Лекция №22, 23 «Вычислительные системы параллельной обработки данных»

1. Параллельная обработка данных как архитектурный способ повышения производительности.

Методы увеличения производительности вычислительных систем. Основы параллельной обработки. Мультипроцессорные архитектуры, ее преимущества. Трудности реализации мультипроцессорных архитектур (новые типы ошибок, сложности понимания и анализа параллельных процессов, недостаточная разработанность теоретических моделей и методов параллельного программирования).

2. Классификация систем параллельной обработки. Многопроцессорные вычислительные системы (особенности организации и функционирования, примеры, преимущества и недостатки):

- MISD компьютеры (МКОД)

- MIMD компьютеры (МКМД)

- Многопроцессорные вычислительные системы (сильно связанные): с общей шиной, с использованием многовходовой памяти,

- Многомашинные вычислительные системы (слабосвязанные): многомашинные комплексы, системы массового параллелизма.

Понятие вычислительных сетей.

Лекция №24 «Вычислительные системы – состояние, производительность, направления развития»

1. Состояние производства и использования. Направления развития архитектуры.

2. Направления развития высокопроизводительных вычислительных систем.

2.1 Тенденции развития архитектур с общей памятью

2.2 Тенденции развития архитектур систем с разделяемой памятью

2.3 Развитие архитектур микропроцессоров высокопроизводительных вычислительных систем (организация внутрикристальной памяти, увеличение состава и числа функциональных устройств, интеграция функций)

2.4 Направления развития мультипроцессорных систем с распределенной памятью

3. Производительность мультипроцессорных систем при увеличении числа процессоров.

4. Вычислительные системы на кристалле.

Переход к нанотехнологии производства интегральных схем.

Лекция № 25, 26, 27 «Телекоммуникационные вычислительные сети»

1. Принципы построения вычислительных сетей

1.1 Основные понятия: телекоммуникационная сеть, абонентская станция, телекоммуникационная система. Обобщенная функциональная схема.

1.2 Организация и работа простейшей сети. Формат сообщений при обмене. Каналы передачи сообщений. Помехоустойчивое кодирование. Последовательность действий при передаче/приеме сообщений. Реакция на подтверждение приема.

1.3 Параметры производительности сети: задержка времени в передающем узле, время передачи данных, время продвижения сигналов, задержка в приемном узле, время транспортировки, время обмена.

1.4 Классификация вычислительных сетей: глобальные, региональные, локальные, системные

1.5 Архитектурные принципы построения сетей. Физические блоки, логическая организация, топология сети, основы обработки сообщений. Протокол сети.

2. Семиуровневая эталонная модель взаимодействия открытых систем OSI/ISO. Уровни иерархии (прикладной, представительский, сеансовый, транспортный, сетевой, канальный, физический) – характеристика, предназначение, организация.

3. Коммутация и маршрутизация при передаче данных в сети. Коммутация сообщений и пакетов. Дейтограмма. Виртуальный канал. Основы маршрутизации. Основные типы сетевого оборудования: коммутаторы, концентраторы, повторители, мосты, шлюзы, маршрутизаторы, мультиплексоры.

4. Локальные вычислительные сети (ЛВС). Характеристики ЛВС. Типы каналов, способы организации. Асинхронный и синхронный формат сообщений. Цифровые коды. Топологии ЛВС. Одноранговые и многогранговые сети. Файл-сервер, клиент-сервер.

5. Локальная вычислительная сеть Ethernet. Трех уровневая организация. Средства подключения ВМ и ЛВС Ethernet. Способы доступа к среде. Формат кадра. Основные скорости передачи.

6. Сеть Интернет. Стек протоколов TCP/IP. Уровни протоколов сети Интернет. Понятия FTP, SMTP, HTTP, TELNET, WWW. Способы подключения абонента к сети Интернет. Корпоративные сети.

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

(темы и рассматриваемые вопросы)

Лабораторные работы проводятся согласно рекомендациям и в форме, изложенной в п.4. Согласно рабочей программе (см. п.1) на лабораторные работы выделяется 18 часов (9 пар занятий). Тематика и рассматриваемые вопросы по каждому занятию представлены ниже. Полный комплект заданий для лабораторных работ представлен в разделе 12.1.

Первое занятие (2 час) – вводное.

Тема: «Изучение базовых возможностей Electronics Workbench»

Цель работы:

1. Ознакомится со средой имитационного исследования Electronics Workbench (EWB).
2. Освоить способы создания различных элементов в среде EWB.
3. Получить представление о методах исследования цифровых элементов в EWB.

Рассматриваемые вопросы:

1. Запуск EWB
2. Назначение панелей инструментов
3. Основные используемые элементы
4. Составление электрических схем различной структуры
5. Запуск имитационного моделирования
6. Снятие данных
7. Подготовка заготовки отчета для первой лабораторной работы

Второе занятие (1 час) – выполнение работы №1.

Тема: «Изучение логических схем и функций»

Цель работы:

1. Исследование базовых логических элементов.
2. Реализация логических функций при помощи логических элементов.
3. Синтез логических схем.

Рассматриваемые вопросы:

1. Исследование логической функции «И»
 - 1.1. Задание уровней логических сигналов
 - 1.2. Экспериментальное получение таблицы истинности элемента «И»
 - 1.3. Получение аналитического выражения для функции
2. Исследование логической функции «И-НЕ»
 - 2.1. Экспериментальное получение таблицы истинности логического элемента «И-НЕ», составленного из элементов «И» и «НЕ»
 - 2.2. Экспериментальное получение таблицы истинности логического элемента «И-НЕ»
3. Исследование логической функции «ИЛИ»

- 3.1. Экспериментальное получение таблицы истинности логического элемента «ИЛИ»
- 3.2. Получение аналитического выражения для функции
- 4. Исследование логической функции «ИЛИ-НЕ»
 - 4.1. Экспериментальное получение таблицы истинности логического элемента «ИЛИ-НЕ», составленного из элементов «ИЛИ» и «НЕ»
 - 4.2. Экспериментальное получение таблицы истинности логического элемента «ИЛИ-НЕ»
- 5. Исследование логических схем с помощью генератора слов
 - 5.1. Исследование микросхем различных типов
 - 5.2. Экспериментальное получение таблицы истинности микросхемы
- 6. Реализация логической функции 3-х и большего числа переменных
 - 6.1. Синтез схемы, реализующей заданную функцию при помощи логического преобразователя
 - 6.2. Синтез логической функции, реализующей заданную таблицу истинности при помощи логического преобразователя

Второе занятие (1 час) – защита работы №1.

Тема: «Изучение логических схем и функций»

Цель работы:

- 1. Выполнить отчет по лабораторной работе №1.
- 2. Ответить на вопросы по методике снятия работы.
- 3. Дать ответы на контрольные вопросы.
- 4. Выполнить упражнения 1.1, 1.2, 1.3.

Третье занятие (2 часа) – выполнение работы №2.

Тема: «Изучение работы шифраторов, дешифраторов и мультиплексоров»

Цель работы:

- 1. Изучение принципов работы шифраторов, дешифраторов и мультиплексоров.
- 2. Реализация логических функций с помощью мультиплексоров.
- 3. Изучение способов применения дешифраторов.

Рассматриваемые вопросы:

- 1. Исследование работы шифратора
- 2. Исследование работы дешифраторов
 - 2.1. Исследование развернутой схемы дешифратора
 - 2.2. Исследование работы схемы дешифратора 3*8 в основном режиме
 - 2.3. Исследование работы схемы дешифратора 3*8 в режиме 2*4
- 3. Применение дешифраторов
- 4. Исследование работы мультиплексора
 - 4.1. Исследование развернутой схемы мультиплексора
 - 4.2. Исследование работы схемы мультиплексора 3*8 в основном режиме
- 5. Реализация логической функции с помощью мультиплексора

Четвертое занятие (2 часа) – защита работы №2.

Тема: «Изучение работы шифраторов, дешифраторов и мультиплексоров»

Цель работы:

1. Выполнить отчет по лабораторной работе №2.
2. Ответить на вопросы по методике снятия работы.
3. Дать ответы на контрольные вопросы.
4. Выполнить упражнения 2.1, 2.2.

Пятое занятие (2 часа) – выполнение работы №3.

Тема: «Изучение работы триггеров»

Цель работы:

1. Изучение структуры и исследование работы асинхронных и синхронных триггеров.
2. Исследование функций переходов и возбуждения основных типов триггеров.
3. Изучение взаимозаменяемости триггеров различных типов.

Рассматриваемые вопросы:

1. Исследование работы схемы RS-триггера
 - 1.1. Исследование RS триггера, составленного из двух элементов «ИЛИ-НЕ»
 - 1.2. Исследование R'S'-триггера, составленного из двух элементов «И-НЕ»
2. Исследование работы JK-триггера
 - 2.1. Составление функции возбуждения JK-триггера
 - 2.2. Составление временных диаграмм JK-триггера
3. Исследование работы D-триггера
 - 3.1. Исследование D триггера составленного на базе JK-триггера
 - 3.2. Получение функций возбуждения D триггера
 - 3.3. Составление временной диаграммы D триггера
4. Исследование работы T триггера
 - 4.1. Исследование T-триггера на основе JK-триггера
 - 4.2. Исследование T-триггера на основе D-триггера

Шестое занятие (2 часа) – защита работы №3.

Тема: «Изучение работы триггеров»

Цель работы:

1. Выполнить отчет по лабораторной работе №3.
2. Ответить на вопросы по методике снятия работы.
3. Дать ответы на контрольные вопросы.
4. Выполнить упражнение 3.1.

Седьмое занятие (2 час) – выполнение работы №4.

Тема: «Изучение сумматоров, полусумматоров, регистров и счетчиков»

Цель работы:

1. Исследование сумматоров и полусумматоров.

2. Изучение структуры и исследование работы суммирующих и вычитающих счетчиков, счетчиков с измененным коэффициентом пересчета.
3. Изучение регистров.

Рассматриваемые вопросы:

1. Изучение полусумматоров и сумматоров
 - 1.1. Изучение работы полусумматора
 - 1.2. Изучение работы одноразрядного сумматора
 - 1.3. Изучение работы двоичного трехразрядного сумматора
2. Исследование работы суммирующего и вычитающего счетчиков
 - 2.1. Исследование первой реализации счетчика
 - 2.2. Исследование второй реализации счетчика
3. Исследование счетчика с измененным коэффициентом пересчета
 - 3.1. Первый вариант
 - 3.2. Второй вариант
4. Исследование регистров
 - 4.1. Исследование параллельного регистра
 - 4.2. Исследование последовательного регистра

Восьмое занятие (2 час) – защита работы №4.

Тема: «Изучение сумматоров, полусумматоров, регистров и счетчиков»

Цель работы:

1. Выполнить отчет по лабораторной работе №4.
2. Ответить на вопросы по методике снятия работы.
3. Дать ответы на контрольные вопросы.

Девятое занятие (2 час) – выполнение и защита работы №5.

Тема: «Изучение сетевых возможностей Windows NT»

Цель работы:

1. Ознакомиться с администрированием Windows NT.
2. Изучить свойства встроенных учетных записей и групп.
3. Ознакомиться с применением средств системной политики и аудита.
4. Рассмотреть сетевые свойства файловой системы NTFS.
5. Научиться управлять доступом к сетевым и локальным ресурсам.
6. Выполнить отчет по лабораторной работе №5.
7. Ответить на вопросы по методике снятия работы.
8. Дать ответы на контрольные вопросы.

Рассматриваемые вопросы:

1. Изучение встроенных учетных записей и групп
 - 1.1. Изучение встроенных групп при установке NT Server как контроллер домена
 - 1.2. Изучение встроенных групп при установке NT как обычного сервера
2. Исследование учетных записей Administrator и Guest
3. Изучение системных привилегий
4. Изучение способов внесения новых пользователей в систему

5. Изучение способов установки прав доступа к файлам и каталогам
6. Изучение разделяемых файловых ресурсов
7. Изучение способов создания персонального профиля
8. Изучение способов редактирования и создания политик

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРАКТИЧЕСКИМ ЗАНЯТИЯМ

(темы и рассматриваемые вопросы)

Практические занятия проводят согласно рекомендациям и в форме, изложенной в п.3.1. Согласно рабочей программе (см. п.1) на практическую работу студентов выделяется 18 часов (9 пар занятий). Тематика и рассматриваемые вопросы по каждому практическому занятию приведены ниже.

Первое занятие (2 часа).

Тема: «Представление информации в вычислительных машинах»

Рассматриваемые вопросы:

1. Системы исчисления
 - 1.1. Десятичная система исчисления
 - 1.2. Двоичная система исчисления
 - 1.3. Двоично-десятичная система
 - 1.4. Шестнадцатеричная система исчисления
 - 1.5. Восьмеричная система исчисления
2. Способы перевода целых чисел
3. Способы перевода дробных чисел
4. Представление положительных и отрицательных чисел, чисел с плавающей точкой
5. Правила сложения двоичных чисел
6. Перевод в машинные коды
7. Арифметические операции над числами с фиксированной и плавающей точкой
8. Арифметические операции над двоично-десятичными кодами чисел

Примечание: для каждого рассматриваемого вопроса варианты заданий формулируются в произвольном виде.

Второе занятие (2 часа).

Тема: «Основы алгебры логики I»

Рассматриваемые вопросы:

1. Упрощение логических функций
 - 1.1. Логические функции задаются в виде СДНФ
 - 1.2. Логические функции задаются в виде СКНФ
 - 1.3. Логические функции задаются в произвольном виде
2. Составление таблиц истинности по логическим функциям
 - 2.1. Логические функции задаются в виде СДНФ
 - 2.2. Логические функции задаются в виде СКНФ
 - 2.3. Логические функции задаются в произвольном виде

Примечание: для каждого рассматриваемого вопроса варианты заданий формулируются в произвольном виде (при этом количество булевых переменных, как правило, не должно превышать четырех).

Третье занятие (2 часа).

Тема: «Основы алгебры логики II»

Рассматриваемые вопросы:

1. Получение логических функций по таблицам истинности
 - 1.1. Таблица истинности от двух переменных
 - 1.2. Таблица истинности от трех переменных
2. Построение схемы по таблице истинности
 - 2.1. Таблица истинности задается от двух переменных
 - 2.2. Таблица истинности задается от трех переменных
3. Составление логической функции по схеме
 - 3.1. Схема на элементах «ИЛИ», «И»
 - 3.2. Схема на элементах «ИЛИ», «И», «НЕ»
 - 3.3. Схема на элементах «ИЛИ-НЕ», «И-НЕ»
4. Реализация в различных базисах
 - 4.1. Реализация в базисе «И-НЕ»
 - 4.2. Реализация в базисе «ИЛИ-НЕ»

Примечание: для каждого рассматриваемого вопроса варианты заданий формулируются в произвольном виде (при этом количество булевых переменных, как правило, не должно превышать четырех; а в схемах, как правило, не должно превышать десяти).

Четвертое занятие (2 часа).

Тема: «Основы представления информации и алгебры логики (промежуточная контрольная работа №1)»

Рассматриваемые вопросы:

Темы представлены в разделе 9.2, полный текст теста приведен в разделе 12.2.

Пятое занятие (2 часа).

Тема: «Построение и применение простейших комбинационных устройств»

Рассматриваемые вопросы:

1. Синтез пороговой ячейки (цифрового автомата формирующего на выходе «1», когда на двух любых его входах имеется «1»)
2. Синтез развернутой схемы шифратора (на базе простейших логических элементов)
3. Синтез развернутой схемы дешифратора (на базе простейших логических элементов)
4. Реализация логических функций на мультиплексоре
 - 4.1. Логические функции задаются в виде СДНФ
 - 4.2. Логические функции задаются в виде СКНФ
 - 4.3. Логические функции задаются в произвольном виде

Примечание: синтезируемые схемы шифраторов и дешифраторов приведены в методических указаниях к лабораторным работам.

Шестое занятие (2 часа).

Тема: «Комбинационные устройства средней степени интеграции»

Рассматриваемые вопросы:

1. Построение схемы дешифратора 3*8 из двух дешифраторов 2*4
2. Реализация логических схем на основе дешифратора и логических элементов указанного типа
 - 2.1. Логическая функция задается в виде СДНФ, логические элементы произвольные
 - 2.2. Логическая функция задается в виде СКНФ, логические элементы произвольные
 - 2.3. Логическая функция задается в виде СДНФ(СКНФ), логические элементы «И-НЕ»
 - 2.4. Логическая функция задается в виде СДНФ(СКНФ), логические элементы «ИЛИ-НЕ»
 - 2.5. Логическая функция задается в произвольном виде, логические элементы «И-НЕ» («ИЛИ-НЕ»)
3. Построение цифрового устройства формирующего заданные комбинации двоичных чисел на своих выходах
 - 3.1. Разработать схему на основе дешифратора, формирующую «0» на указанных выходах Y0, Y1, Y2, Y3, при определенных комбинациях входных чисел, заданных в шестнадцатеричной системе исчисления

Числа на входе, в hex	0, 1, A	2, 3, B	4, 5, C	6, 7, E
Выходы	Y0	Y1	Y2	Y3

Примечание:

- а) На вход схемы числа поступают в двоичной системе исчисления;
- б) На выходе формируется «0» тогда и только тогда когда на входе схемы имеется указанные числа (см. таблицу), во всех других случаях – на выходе «1».

- 3.2. Разработать схему на основе дешифратора, формирующую «1» на указанных выходах Y4, Y5, Y6, Y7, при определенных комбинациях входных чисел, заданных в шестнадцатеричной системе исчисления

Числа на входе, в hex	A, B, C	D, E, F	0, 2, 4	1, 3, 7
Выходы	Y4	Y5	Y6	Y7

Примечание:

- а) На вход схемы числа поступают в двоичной системе исчисления;
- б) На выходе формируется «1» тогда и только тогда когда на входе схемы имеется указанные числа (см. таблицу), во всех других случаях – на выходе «0».

Седьмое занятие (2 часа).

Тема: «Комбинационные устройства средней степени интеграции (промежуточная контрольная работа №2)»

Темы представлены в разделе 9.2, полный текст теста приведен в разделе 12.2.

Восьмое занятие (2 часа).

Тема: «Применение комбинационных устройств»

Рассматриваемые вопросы:

1. Синтез сегментных дисплеев различной конфигурации
2. Построение схемы выборки микросхем памяти (ОЗУ и ПЗУ) на основе дешифраторов различной структуры для микроконтроллера КР-580

Примечание: для каждого рассматриваемого вопроса варианты заданий формулируются в произвольном виде (при этом количество переменных входящих в логические функции, как правило, не должно превышать четырех).

Девятое занятие (2 часа).

Тема: «Последовательностные схемы»

Рассматриваемые вопросы:

1. Разработать схему (на основе RS триггера), которая обнаруживает появление хотя бы одного кратковременного спада (установки в «0») на своем входе импульса
2. Разработать схему (на основе RS триггера), которая обнаруживает хотя бы одно кратковременное появление (установки в «1») на своем входе импульса
3. Разработать схему счетчика с коэффициентом пересчета 2, 4
4. Разработайте схему счетчика с коэффициентом счета 3, 5
5. Разработать схему счетчика, с произвольным коэффициентом пересчета, которая устраняет явление «иголок»

Примечание:

а) Для заданий 1, 2 под обнаружением импульса подразумевается получение на выходе схемы «1», и сохранении ее в дальнейшем при произвольном изменении входного сигнала;

б) В рамках выполнения задания 4 подразумевается создание двух видов схем – в первой (простейшей) обнаруживается первая «нерабочая» комбинация и по ней происходит обнуление счетчика; во второй обнаруживаются все «нерабочие» комбинации и обнуление счетчика происходит по всем «нерабочим» комбинациям;

в) При выполнении задания 5 предполагается использование D триггера, в этом случае обнаруживается последняя рабочая комбинация пересчета, а триггер выполняет задержку сброса на один такт (считается, что время обработки сигнала в триггере меньше, чем в схеме сброса).

9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ КОНТРОЛЬНЫХ РАБОТ

(формы проведения, темы и рассматриваемые вопросы)

9.1. ФОРМЫ ПРОВЕДЕНИЯ КОНТРОЛЬНЫХ РАБОТ

На первом практическом занятии студенты знакомятся с формой проведения, списком прорабатываемых вопросов и датой выполнения контрольных работ. Всего предусматривается проведение двух промежуточных контрольных работ (см. п.9.2, п.12.2). Все контрольные работы выполняются студентами самостоятельно, в рамках практических занятий (см. п.8). При проведении контрольных работ рекомендуется разрешить студентам использовать собственные наработки (конспекты лекций и практических занятий) дабы стимулировать самостоятельную работу, но строго следить за отсутствием коллективного решения задач (в отличие от общего курса практических занятий).

Первая контрольная работа (выполняется на четвертом практическом занятии) предназначена для промежуточного контроля усвоения материала студентами. Вопросы, рассматриваемые в рамках данной работы (полный перечень вопросов представлен в п.12.2), предварительно прорабатываются при чтении лекционного курса, на практических и лабораторных занятиях, при выполнении студентами самостоятельной работы. Время проведения контрольной работы – 2 часа (1 пара занятий). Рекомендуемая форма оценки – пятибалльная, с проставлением десятых бала (например, 4.7). Оценка, полученная студентами, на контрольной работе ложится в основу оценки по первой контрольной точке. На следующем практическом занятии рекомендуется так же уделить время типичным ошибкам, встречающимся у большинства студентов (если таковые имеются).

Вторая контрольная работа (выполняется на седьмом практическом занятии) предназначена для промежуточного контроля усвоения и закреплении изученного материала студентами. Вопросы, рассматриваемые в рамках данной работы (полный перечень вопросов представлен в п.12.2), аналогично первой работе, предварительно прорабатываются при чтении лекционного курса, на практических и лабораторных занятиях, при выполнении студентами самостоятельной работы. Время проведения контрольной работы – 2 часа (1 пара занятий). Рекомендуемая форма оценки – пятибалльная, с проставлением десятых бала (например, 4.7). Оценка, полученная студентами, на контрольной работе ложится в основу оценки по второй контрольной точке. На следующем практическом занятии рекомендуется так же уделить время типичным ошибкам, встречающимся у большинства студентов (если таковые имеются).

9.2. ТЕМЫ И РАССМАТРИВАЕМЫЕ ВОПРОСЫ КОНТРОЛЬНЫХ РАБОТ

Контрольная работа №1: «Основы представления информации и алгебры логики»

Рассматриваемые вопросы:

1. Перевод из одной системы исчисления в другую
2. Сложение и вычитание чисел в различной системе исчисления
3. Составление логических функций по таблице истинности
4. Упрощение полученных логических функций, полученных в задании 1
5. Построение схемы в произвольном базисе по логическим функциям, полученным в задании 4
6. Реализация логических функций, полученных в задании 4 в базисе «И-НЕ»; построение соответствующих схем
7. Реализация логических функций, полученных в задании 4 в базисе «ИЛИ-НЕ»; построение соответствующих схем
8. По заданной схеме построить логическую функцию

Примечание: полный текст контрольной работы представлен в разделе 12.2.

Контрольная работа №2: «Комбинационные устройства средней степени интеграции»

Рассматриваемые вопросы:

1. Реализовать заданные логические функции на мультиплексоре (логические функции задаются в виде СДНФ, в произвольном виде)
2. Реализовать заданные логические функции на основе дешифратора и логических элементов заданного типа (логические элементы «И-НЕ», «ИЛИ-НЕ»)
3. Разработать схему на основе дешифратора, формирующую «1» («0») на указанных выходах, при определенных комбинациях входных чисел

Примечание: полный текст контрольной работы представлен в разделе 12.2.

10. ПЕРЕЧЕНЬ ПРОГРАММНЫХ ПРОДУКТОВ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРИМЕНЕНИЮ СОВРЕМЕННЫХ ИТ ДЛЯ ПРЕПОДАВАНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Для преподавания дисциплины «Вычислительные машины системы и сети» основным используемым программным продуктом является программа «Electronics Workbench» – удобная и гибкая среда компьютерного моделирования. Данная программа используется, в основном, при проведении лабораторных работ. Методические указания к лабораторным работам¹ содержат все необходимые сведения для проведения экспериментов. Необходимо отметить, что использование «Electronics Workbench» не является жестким ограничением: лабораторные работы можно проделать и в других системах компьютерного моделирования. Для этого в методическом пособии к лабораторным работам (см. п.12.1), помимо обозначения электронных элементов, принятых в нашей стране, описаны обозначения принятые на Западе. Для выполнения последней, пятой работы предполагается использование Windows NT – основные возможности и принцип работы, которой и являются целью лабораторной работы и здесь не приводятся.

10.1. ВОЗМОЖНОСТИ «ELECTRONICS WORKBENCH»²

Система схемотехнического моделирования «Electronics Workbench» предназначена для моделирования и анализа электрических схем.

Программа «Electronics Workbench» позволяет моделировать аналоговые, цифровые и цифро-аналоговые схемы большой степени сложности. Имеющиеся в программе библиотеки включают в себя большой набор широко распространенных электронных компонентов. Есть возможность подключения и создания новых библиотек компонентов. Параметры компонентов можно изменять в широком диапазоне значений. Простые компоненты описываются набором параметров, значения которых можно изменять непосредственно с клавиатуры, активные элементы - моделью, представляющей собой совокупность параметров и описывающей конкретный элемент или его идеальное представление. Модель выбирается из списка библиотек компонентов, параметры модели также могут быть изменены пользователем. Широкий набор приборов позволяет производить измерения различных величин, задавать входные воздействия, строить графики. Все приборы изображаются в виде, максимально приближенном к реальному, поэтому работать с ними просто и удобно. Результаты моделирования можно вывести на принтер или импортировать в текстовый или графический редактор для их дальнейшей обработки. Программа «Electronics Workbench» совместима с программой P-

¹ Теличенко Д. А., Романова М. В.. Цифровые узлы и элементы организации вычислительных систем. Лабораторный практикум. Благовещенск: Амурский гос. ун-т, 2004.

² Электротехника и электроника в экспериментах и упражнениях. Практикум на Electronics Workbench / Под общ. Ред. Д. И. Панфилова. – М.: Додека, 2000.

SPICE, то есть предоставляет возможность экспорта и импорта схем и результатов измерений в различные ее версии.

10.2. РАБОТА С ПРОГРАММОЙ «ELECTRONICS WORKBENCH»

В рамках проведения лабораторных занятий предусматривается использование программы «Electronics Workbench», далее EWB версии 5.12. Дальнейшее описание EWB касается необходимого минимума, достаточного для выполнения лабораторных заданий.

Работа с программой начинается с запуска файла «EWB32.EXE», находящегося в корневом каталоге «EWB512». При этом открывается главное окно, представленное на рис. 10.1

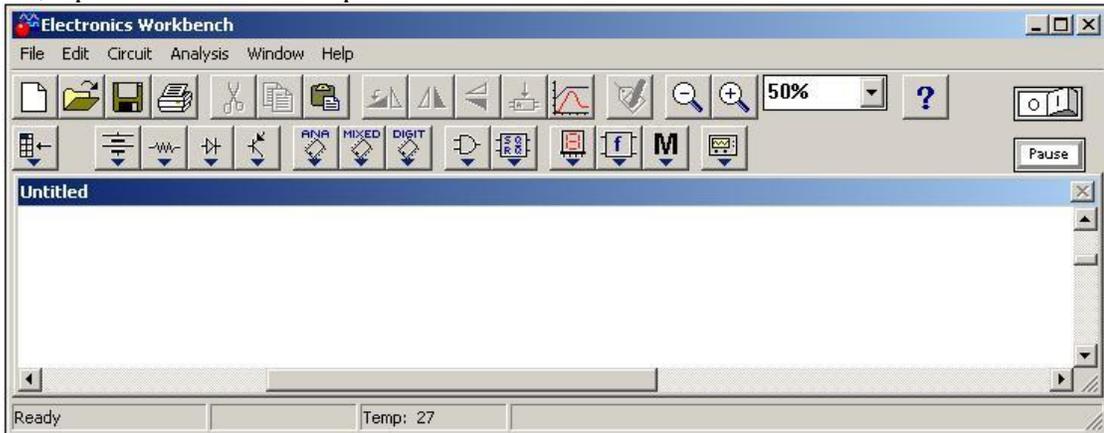


Рисунок 10.1 – Главное окно программы

Для создания исследуемой схемы необходимо выполнить следующее:

- открыть соответствующую панель инструментов (рисунок 10.2);
- «перетянуть» нужные элементы из панели инструментов на рабочий стол (рисунок 10.3);

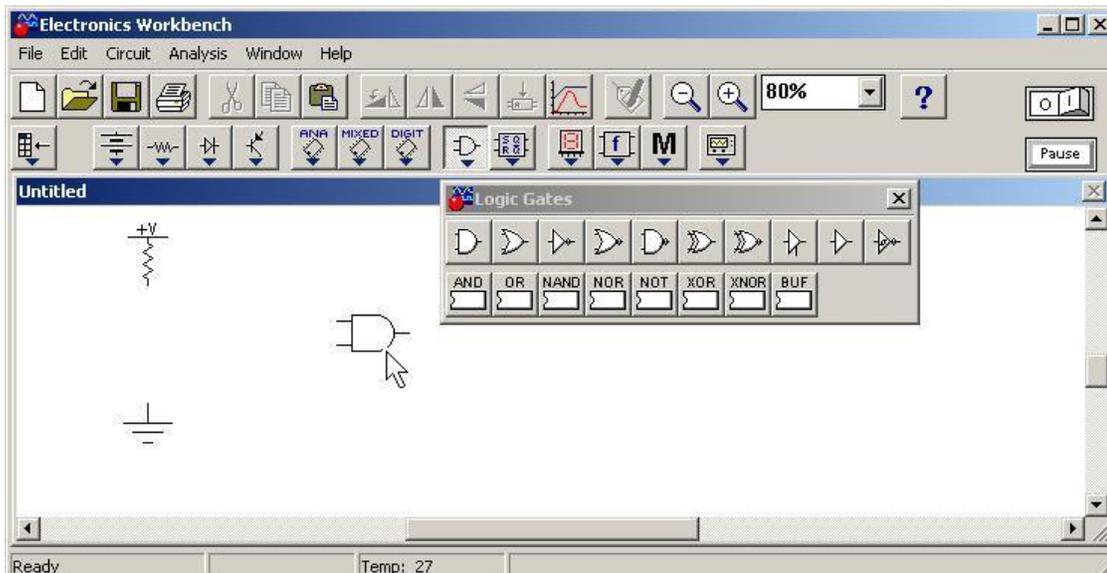


Рисунок 10.3 – Создание исследуемой схемы

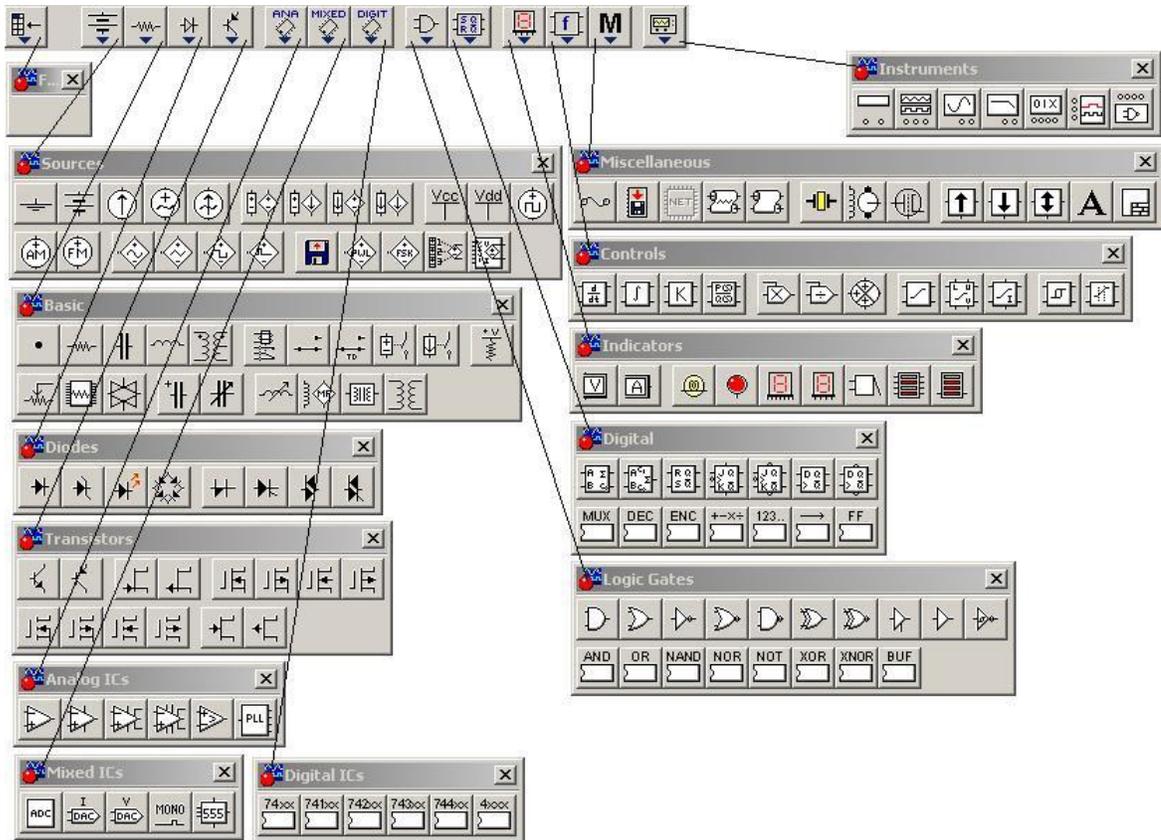


Рисунок 10.2 – Панели инструментов

Примечания:

1. выбрав какой либо элемент и нажав клавишу F1 можно вызвать справку об элементе, где указано предназначение, а так изменяемые свойства данного элемента (см. рисунок 10.4)
2. выбрав какой либо элемент и нажав на правую кнопку мыши можно вызвать ниспадающее меню и повернуть элемент в нужном направлении, а так же задать его особые свойства (см. рисунок 10.5)
3. окно свойств элемента можно так же вызвать, два раза щелкнув на элементе мышью, при этом выбираются ряд нужных параметров, таких как количество входов-выходов, кнопка переключения (для ключа) и т.п.

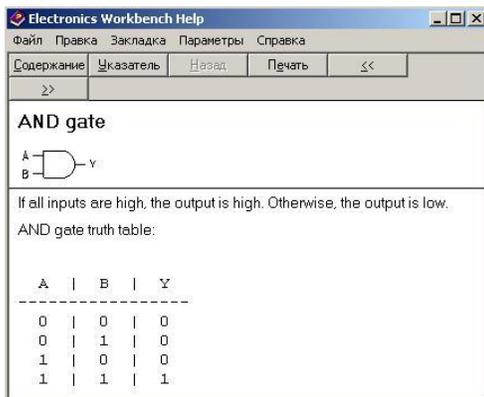


Рисунок 10.4

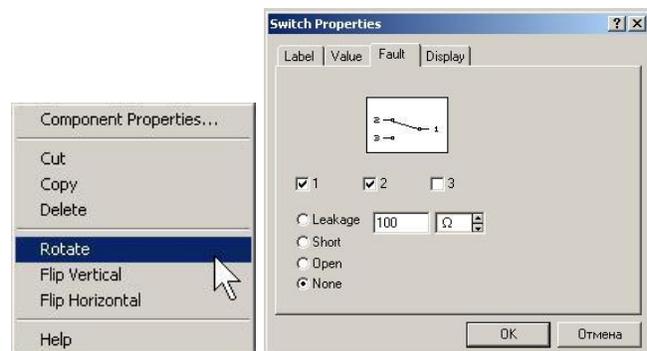


Рисунок 10.5

в) соединить соответствующие точки, создаваемой электрической схемы; для этого необходимо подвести курсор мыши к первому соединяемому элементу до появления черной точки, затем, нажав левую кнопку, подвести курсор ко второму соединяемому элементу до появления черной точки, и отпустить левую кнопку; (см. рисунок 10.6).

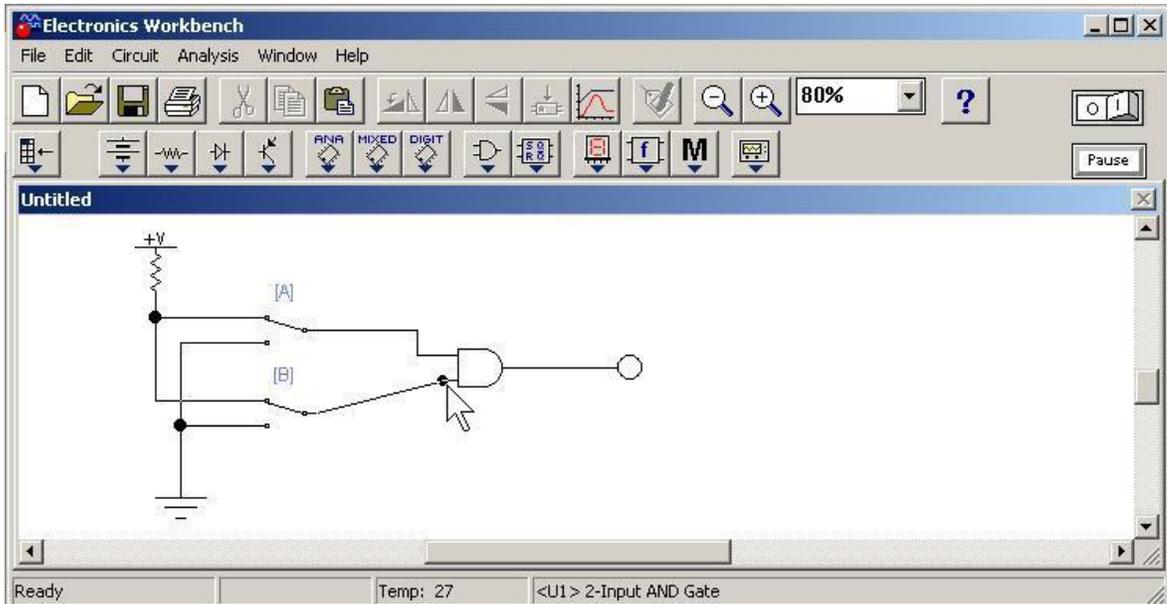


Рисунок 10.6 – Соединение элементов

После сбора схемы можно приступить к ее изучению, включив ее в работу тумблером, находящимся в правом верхнем углу экрана.

Общие замечания:

1. Пояснения по местонахождению основных элементов, способам задания специальных свойств приведены непосредственно в каждой лабораторной работе по мере использования соответствующих элементов (см. п. 12.1);
2. В ходе выполнения работы необходимо следить за тем, чтобы при исследовании схема находилась в работе, так как после изменения каких либо ее составных частей, а так же по истечению времени моделирования EWB принудительно прекращает режим моделирования.

11. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОРГАНИЗАЦИИ МЕЖСЕССИОННОГО И ЭКЗАМЕНАЦИОННОГО КОНТРОЛЯ ЗНАНИЙ СТУДЕНТОВ

Организация аттестации студентов, проводится в соответствии с положением АмГУ о курсовых экзаменах и зачетах (далее приведена выписка из положения):

2.1. Организация аттестации студентов в университете по специальностям и направлениям высшего профессионального образования регламентируется рабочим учебным планом, расписанием учебных занятий и программами учебных дисциплин, утверждаемыми в установленном в университете порядке.

Контроль за качеством освоения образовательных программ осуществляется путем текущей внутрисеместровой аттестации, ректорской контрольной аттестации, промежуточной аттестации студентов в форме курсовых экзаменов и зачетов и итоговой аттестации выпускников.

2.2. Курсовые экзамены и зачеты проводятся по дисциплинам утвержденного учебного плана по соответствующим специальностям и направлениям высшего профессионального образования. Знания, умения и навыки обучающегося определяются оценками "отлично", "хорошо", "удовлетворительно", "неудовлетворительно", "зачтено" и "незачтено".

2.3. Студенты, обучающиеся по основным программам высшего профессионального образования, сдают в течение учебного года не более 10 экзаменов и 12 зачетов. В это число не входит аттестация по физической культуре и факультативным дисциплинам.

Студенты, обучающиеся в сокращенные сроки (по индивидуальным планам), в течение учебного года сдают не более 20 экзаменов и 24 зачетов.

2.4. Сроки проведения курсовых зачетов и экзаменов (экзаменационная сессия) и начало очередного учебного семестра устанавливаются графиком учебного процесса, утвержденным проректором по учебной работе.

Расписание экзаменов составляется в соответствии с графиком учебного процесса, утверждается проректором по учебно-научной работе и доводится до сведения преподавателей и студентов не позднее, чем за две недели до начала сессии. Расписание составляется таким образом, чтобы на подготовку к экзаменам по каждой дисциплине было отведено не менее 3 дней, исключая день предыдущего экзамена. По согласованию с деканами и заведующими соответствующих кафедр отдельные экзамены (зачеты) могут проводиться в течение семестра по завершении преподавания дисциплины.

11.1. ОРГАНИЗАЦИЯ МЕЖСЕССИОННОГО КОНТРОЛЯ

В соответствии с вышеприведенным положением АмГУ рекомендуется следующий способ межсессионной аттестации студентов. Аттестация проводится дважды в семестр на 7 и 13 неделе 6 семестра. Аттестационная оценка складывается из следующих оценок:

- оценки полученной на соответствующей контрольной работе (п.9.1, п.12.2);
- оценки характеризующей выполнение и защиту лабораторных работ (п.4, п.7, 12.1);
- оценки характеризующей работу студентов на практических и семинарских занятиях (п.3.1, п.8).

При этом преимущественным весом обладают оценки, характеризующие персональное усвоение материала студентом (оценка по контрольной

работе и оценка, характеризующая выполнение лабораторных работ). Оценка, характеризующая работу студента на практических занятиях, в большинстве случаев может учитываться в роли повышающей, если таковая работа студента имеется.

11.2. ОРГАНИЗАЦИЯ ЭКЗАМЕНАЦИОННОГО КОНТРОЛЯ

В соответствии с вышеприведенным положением АмГУ итоговые знания и умения студента определяются оценками «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно». Учебным планом предусматривается устная сдача экзамена по дисциплине «Вычислительные машины системы и сети». Основные вопросы, на которые студенту предстоит ответить на экзамене, определяются экзаменационным билетом (п. 14). Экзаменационный билет состоит из двух теоретических вопросов и третьего – практического (в рамках которого студенту предлагается решить предложенные задачи).

Помимо ответа по экзаменационному билету в случае наличия неликвидированных задолженностей по лабораторным работам, студентом на экзамене так же защищаются и несданные работы. Оценка, полученная по результатам защиты лабораторных работ, учитывается при проставлении итоговой.

Студенты, проявившие особые успехи в освоении дисциплины (стоцентная посещаемость занятий, успешное выполнение плана по сдаче лабораторных работ, отличная работа на практических занятиях, получившие оценку отлично на контрольных работах) могут быть по результатам выполнения теста (см. п.13.2) освобождены от ответа на один или несколько экзаменационных вопросов.

12. КОМПЛЕКТЫ ЗАДАНИЙ

12.1. ЛАБОРАТОРНЫЕ РАБОТЫ

ЛАБОРАТОРНАЯ РАБОТА № 1

Изучение логических схем и функций

Цель работы:

1. Исследование базовых логических элементов.
2. Реализация логических функций при помощи логических элементов.
3. Синтез логических схем.

Приборы и элементы:

Логический преобразователь (панель «Instruments/Logic Converter»)
Генератор слов (панель «Instruments/Word Generator»)
Вольтметр (панель «Indicators/Voltmeter»)
Логические пробники (панель «Indicators/Red probe»)
Источник напряжения + 5 В (панель «Basic/Pull-Up Resistor»)
Земля (панель «Sources/Ground»)
Двухпозиционные переключатели (панель «Basic/Switch»)
Двухвходовые элементы И, И-НЕ, ИЛИ, ИЛИ-НЕ (панель «Logic Gates/2-Input AND, NAND, OR, NOR Gates»)
Микросхемы различных серий (панель «Logic Gates/»).

Краткие теоретические сведения

Любые цифровые микросхемы современных вычислительных машин строятся на основе простейших логических элементов «И», «ИЛИ», «НЕ» и их комбинаций. В настоящее время используется несколько технологий построения логических элементов:

- транзисторно-транзисторная логика (ТТЛ, TTL);
- логика на основе комплементарных МОП транзисторов (КМОП, CMOS);
- логика на основе сочетания комплементарных МОП и биполярных транзисторов (БиКМОП);
- эмиттерно-связанная логика (ЭСЛ), и т.д.

Прежде чем приступить к изучению базовых логических элементов введем основные понятия булевой алгебры или алгебры логики, на которой базируется все теоретическое обоснование работы цифровых устройств.

1. Основные определения и аксиомы алгебры логики

Переменные, рассматриваемые в алгебре логики, принимают только два значения – «0» или «1». Чаще всего сами переменные обозначаются латинскими буквами: либо малыми (x, y, z, \dots) или большими (A, B, C, \dots).

Примечание. Физически самым простым устройством, моделирующим поведение любой булевой переменной, является двух позиционный переключатель.

чател. У него за логический сигнал «1» принимается положение «включено», а за сигнал «0» положение «выключено». Поэтому самой простой схемой для изучения логических функций и выражений является электрическая схема, состоящая из исследуемого элемента или элементов, источников питания, заземления и устройств, моделирующих задание уровней логических сигналов (в простейшем случае ими являются переключатели). Причем в данном случае важно подчеркнуть неразрывную связь между физическим сигналом (ток или напряжение) и логическим сигналом: за логический сигнал «0» и «1» принимаются определенные уровни физических сигналов (чаще за логический «0» принимается сигнал по напряжению в 0 В, а за уровень логической «1» +5 В).

Перечислим основные операции и аксиомы алгебры логики.

Основные операции алгебры логики:

- отношение эквивалентности, обозначается знаком « \equiv »;
- операция логического сложения (дизъюнкция), обозначаемая знаком « \vee » или « $+$ »;
- операция логического умножения (конъюнкция), обозначаемая знаком « \wedge » или « \bullet »;
- операция отрицания (инверсии), обозначаемая надчеркиванием или апострофом «'».

Основные аксиомы булевой алгебры:

$$\left\{ \begin{array}{l} \overline{\overline{0}} = 1, \\ \overline{\overline{1}} = 0. \end{array} \right. \quad \left\{ \begin{array}{l} 0 \vee 0 = 0 + 0 = 0, \\ 1 \vee 0 = 1 + 0 = 1, \\ 1 \vee 1 = 1 + 1 = 1. \end{array} \right. \quad \left\{ \begin{array}{l} 0 \wedge 0 = 0 \cdot 0 = 0, \\ 0 \wedge 1 = 0 \cdot 1 = 0, \\ 1 \wedge 1 = 1 \cdot 1 = 1. \end{array} \right. \quad (1)$$

2. Логические выражения

Из логических переменных с помощью базовых логических операций можно составить логическое выражение. Логические выражения связывают значение логической функции со значениями логических переменных.

Логическое выражение является одним из способов описания цифрового устройства.

Запись логических выражений обычно осуществляют в конъюнктивной или дизъюнктивной нормальных формах. В дизъюнктивной форме логические выражения записываются как логическая сумма логических произведений, в конъюнктивной форме – как логическое произведение логических сумм.

3. Логические тождества

При преобразованиях логических выражений используются следующие логические тождества:

$$\begin{aligned} \bar{\bar{x}} &= x; & x \vee 1 &= 1; & x \vee 0 &= x; & x \cdot 1 &= x; & x \cdot 0 &= 0; \\ x \vee x &= x; & x \cdot x &= x; & x \vee x \cdot y &= x; \\ xy \vee x\bar{y} &= x; & (x \vee y)(x \vee \bar{y}) &= x; & x \vee \bar{x}y &= x \vee y; \\ \overline{xy} &= \bar{x} \vee \bar{y}; & \overline{\bar{x} \vee \bar{y}} &= \overline{\bar{x} \vee \bar{y}}. \end{aligned} \tag{2}$$

4. Логические функции

Любое логическое выражение, составленное из n переменных x_n, x_{n-1}, \dots, x_1 , с помощью конечного числа операций алгебры логики, можно рассматривать как некоторую функцию n переменных. Такую функцию называют логической. В соответствии с аксиомами алгебры логики такая функция может принимать (в зависимости от значения логических переменных входящих в нее) значение «0» или «1». Функция n логических переменных может быть определена для 2^n значений переменных, соответствующих всем возможным значениям n -разрядных двоичных чисел. Основным интересом представляют следующие базовые логические функции двух переменных x и y :

$$f_1(x, y) = x \cdot y \text{ – логическое умножение (конъюнкция),} \tag{3}$$

$$f_2(x, y) = x \vee y \text{ – логическое сложение (дизъюнкция),} \tag{4}$$

$$f_3(x, y) = \overline{x \cdot y} \text{ – логическое умножение с инверсией,} \tag{5}$$

$$f_4(x, y) = \overline{x \vee y} \text{ – логическое сложение с инверсией,} \tag{6}$$

$$f_5(x, y) = x \oplus y = x\bar{y} \vee \bar{x}y \text{ – суммирование по модулю 2,} \tag{7}$$

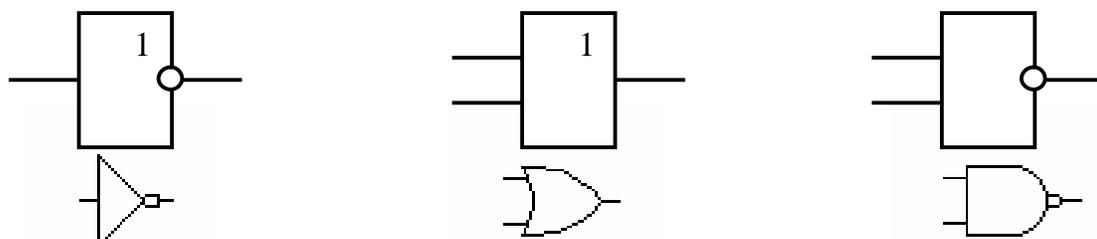
$$f_6(x, y) = x \oplus y = xy \vee \bar{x}\bar{y} \text{ – равнозначность.} \tag{8}$$

5. Логические схемы

Физическое устройство, реализующее одну из операций алгебры логики или простейшую логическую функцию, называется логическим элементом. Схема, составленная из конечного числа логических элементов по определенным правилам, называется логической схемой. Логическая схема наряду с логическим выражением является одним из способов задания цифровых элементов.

Условное обозначение основных логических элементов принятых в нашей стране и за рубежом приведено на рис. 1.

Условное обозначение основных логических элементов



Инвертор (НЕ)

ИЛИ

И-НЕ

Рис. 1 – Обозначение логических элементов

Примечание: количество входов в приведенных элементах может быть произвольным; на операцию инвертирования указывает кружок на выходном сигнале.

Здесь на рис. 1 в первом ряду приведено обозначение, принятое в нашей стране, во втором ряду, обозначение, принятое за рубежом.

6. Таблица истинности

Так как область определения любой логической функции n переменных конечна (2^n значений), такая функция может быть задана таблицей значений $f(v_i)$ которые она принимает в точках v_i , где $i = 0, 1, \dots, 2^n - 1$. Такие таблицы называют таблицами истинности. В таблице 1 представлены таблицы истинности, задающие указанные выше функции (3) – (8).

Таблица 1 – Произвольная таблица истинности

№	Значения переменных		Функции					
	x	y	f1	f2	f3	f4	f5	f6
1	0	0	0	0	1	1	0	1
2	0	1	0	1	1	0	1	0
3	1	0	0	1	1	0	1	0
4	1	1	1	1	0	0	0	1

В общем случае таблица истинности должна содержать все возможные комбинации логических переменных, входящих в логическое выражение (в случае таблицы 1 это комбинации логических переменных x и y), и значения логического выражения, соответствующие каждой комбинации логических переменных

Таблица истинности является третьим способом задания цифровых элементов. Необходимо отметить, что все три способа задания (с помощью логических выражений, логических схем и таблиц истинности) являются однозначными и в равной мере взаимно заменяемыми. Так по логическому выражению можно составить схему и записать таблицу истинности и наоборот.

7. Получение логических выражений

Целью преобразования сложных логических выражений (а так же получения по экспериментальной таблице истинности логического выражения, описывающего данную логическую зависимость) является получение компактной формы записи, которая в полной мере описывает данную логическую зависимость от любого числа переменных. Операции с логическими выражениями намного удобней, чем с таблицами истинности и со схемами.

Самый простой способ получения логических выражений это анализ таблицы истинности. Для наглядности рассмотрим пример: пусть требуется найти логическое выражение для функции f_m трех переменных X, Y, Z , описываемой таблицей истинности 2.

Таблица 2 – Таблица истинности

N	X	Y	Z	fm
1	0	0	0	0
2	0	0	1	0
3	0	1	0	0
4	0	1	1	1
5	1	0	0	0
6	1	0	1	1
7	1	1	0	1
8	1	1	1	1

Из правил записи логических выражений следует, что искомую зависимость можно получить в двух видах (конъюнктивной или дизъюнктивной нормальных формах – пункт 2). Чаще всего на практике применяется дизъюнктивно нормальная форма записи логической функции.

Дизъюнктивно нормальная форма записи представляет собой дизъюнкцию (логическое сложение) элементарных конъюнкций (произведение всех логических переменных, в нашем случае – это три переменные X, Y, Z).

Для записи логического выражения в дизъюнктивно нормальной форме выберем из таблицы истинности 2 строки, в которых функция fm принимает значение «1». Это строки 4, 6, 7, 8. Таким образом, элементов нашей дизъюнкции будет 4 (всего слагаемых). Каждое слагаемое это элементарная конъюнкция. В каждую же элементарную конъюнкцию (произведение), как было сказано выше, должны войти все логические переменные данной строки (в нашем случае X, Y, Z). При этом если соответствующий элемент (X, Y или Z) имеет для данной строки значение «1», то он входит в элементарную конъюнкцию (произведение) в обычном виде, если же он равен «0», то необходимо взять его отрицание. Так для каждой строки имеем элементарную конъюнкцию (произведение) в следующем виде:

$$\bar{X} \cdot Y \cdot Z \quad \text{для строки 4;}$$

$$X \cdot \bar{Y} \cdot Z \quad \text{для строки 6;}$$

$$X \cdot Y \cdot \bar{Z} \quad \text{для строки 7;}$$

$$X \cdot Y \cdot Z \quad \text{для строки 8;}$$

Искомой формой записи данного логического выражения в дизъюнктивно нормальной форме записи является дизъюнкция (сложение) полученных выше 4 конъюнкций:

$$fm = \bar{X}YZ \vee X\bar{Y}Z \vee XY\bar{Z} \vee XYZ \quad (9)$$

По данной логической функции можно записать таблицу истинности, идентичную таблице 2. Для этого необходимо в логическую функцию подставлять все возможные комбинации логических переменных X, Y, Z (всего таких значений 2^3) и на каждой подстановке, пользуясь аксиомами алгебры логики вычислять функцию fm .

Необходимо отметить, что данный способ получения логических выражений не является самым рациональным. Полученная логическая функция,

хоть и содержит исчерпывающую информацию о выбранной комбинации логических переменных, тем не менее, поддается еще упрощению с помощью выражений (2).

Другим способом составления упрощенного выражения по таблице истинности является составление карт Карно. Данный способ в данной работе не рассматривается по двум причинам:

во-первых, наглядность данного метода минимальна при достаточно большой сложности построения самих карт Карно, что вызывает большие трудности у студентов, особенно в начале изучения цифровой техники;

во-вторых, методика построения карт Карно хорошо описана в литературе по цифровой технике и электронике.

Порядок работы

Задание 1. Исследование логической функции «И»

а) Задание уровней логических сигналов

Создайте схему, изображенную на рис. 1.а

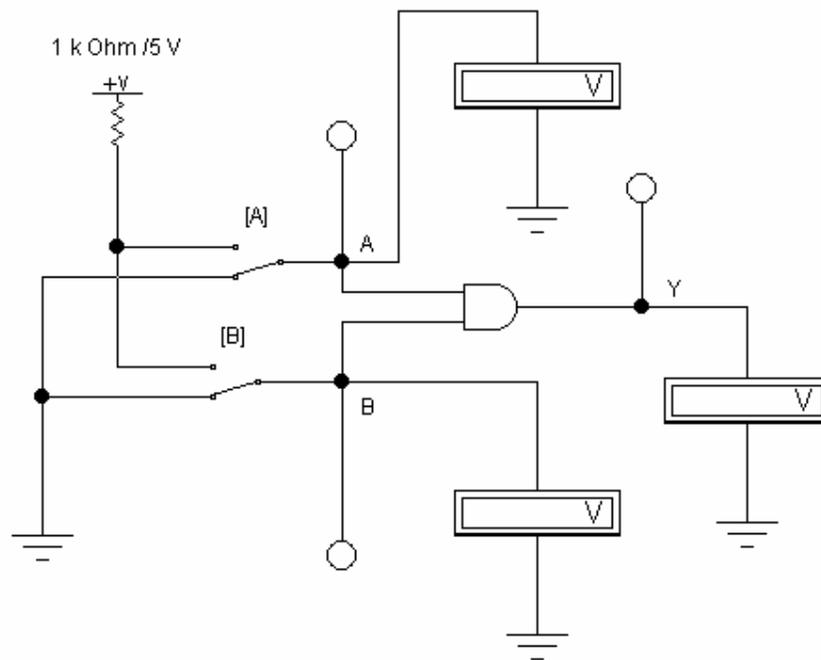


Рис. 1.а – Логическая функция «И»

В этой схеме два двухпозиционных переключателя «А» и «В» подают на входы логической схемы «И» уровни логических сигналов: «0» (контакт переключателя в нижнем положении) или «1» (контакт переключателя в верхнем положении). В физическом плане это соответствует подаче напряжения от источника (+ 5 В) на логический элемент. Уровень физического сигнала на входе или выходе логического элемента можно измерить с помощью вольтметра (в данном случае «0» В или «+5» В), а уровень логического сигнала с помощью логических пробников, которые информируют о наличии

на измеряемой линии сигнала вообще (если сигнал есть, то пробник загорается).

Включите схему (тумблером, находящимся в правом верхнем углу окна программы). Установите положения ключей в соответствии с табл. 1.а. Результаты замеров (логических и физических сигналов) занесите в табл. 1.а.

Таблица 1.а – Задание уровней логических сигналов

Положение переключателей		Сигналы на входах и выходе					
«А»	«В»	Логические (0 или 1)			Физические, В		
		А	В	У	А	В	У
Нижнее	Нижнее						
Нижнее	Верхнее						
Верхнее	Нижнее						
Верхнее	Верхнее						

Примечание: прежде чем начать работу с переключателями удобно каждому переключателю присвоить букву, при нажатии которой он включается/выключается. Это можно сделать до начала работы схемы, дважды щелкнув на переключатель и на закладке «Value» присвоив ему уникальную кнопку.

б) Экспериментальное получение таблицы истинности элемента «И»

Подайте на входы схемы (рис. 1.а) все возможные комбинации уровней сигналов «А» и «В» и для каждой комбинации зафиксируйте уровень выходного сигнала «У». Заполните таблицу истинности исследуемой логической схемы «И» (таблица 1.б)

Таблица 1.б – Таблица истинности логического элемента «И»

Входы		Выход
А	В	У

в) Получение аналитического выражения для функции

По таблице 1.б составьте аналитическое выражение функции элемента «И» и занесите его себе в отчет. Для этого можно воспользоваться одним из двух способов получения логических выражений по таблице истинности изложенных в краткой теории к данной лабораторной работе.

Задание 2. Исследование логической функции «И-НЕ»

а) Экспериментальное получение таблицы истинности логического элемента «И-НЕ», составленного из элементов «И» и «НЕ»

Соберите схему, изображенную на рис. 2.а.

Включите схему. Подайте на входы схемы все возможные комбинации уровней входных сигналов и, наблюдая уровни сигналов на входах и выходе

с помощью логических пробников, заполните таблицу истинности логической схемы «И-НЕ» (табл. 2.а).

Таблица 2.а – Таблица истинности элемента «И-НЕ» (составного)

Входы		Выход
A	B	Y
0	0	
0	1	
1	0	
1	1	

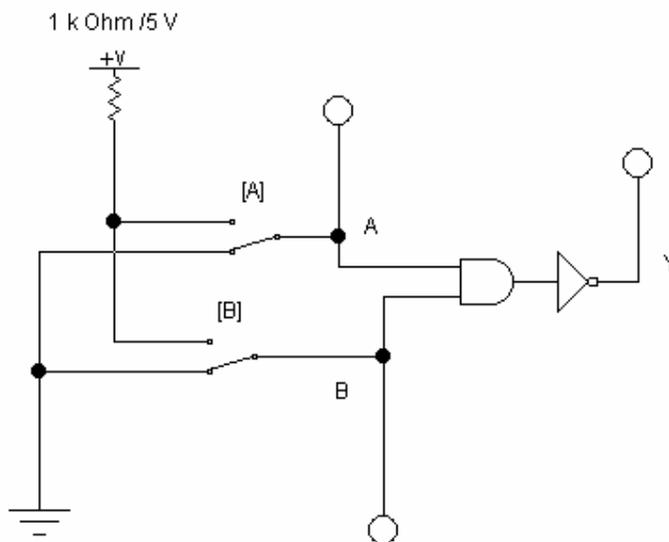


Рис. 2.а – Логическая функция «И-НЕ»

б) Экспериментальное получение таблицы истинности логического элемента «И-НЕ»

Соберите схему, изображенную на рис. 2.б.

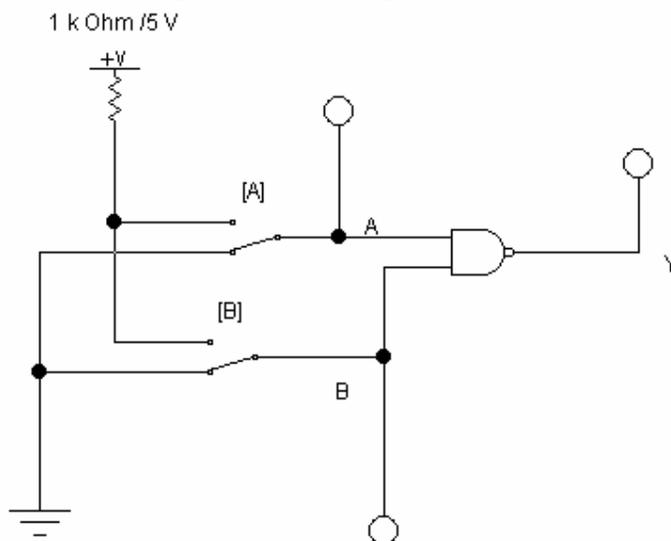


Рис. 2.б – Логическая функция «И-НЕ»

Включите схему. Подайте на входы схемы все возможные комбинации уровней входных сигналов и, наблюдая уровни сигналов на входах и выходе с помощью логических пробников, заполните таблицу истинности логической схемы «И-НЕ» (табл. 2.б).

Таблица 2.б – Таблица истинности элемента «И-НЕ»

Входы		Выход
A	B	Y
0	0	
0	1	
1	0	
1	1	

Сравните таблицы 2.а и 2.б между собой и сделайте соответствующие выводы.

Задание 3. Исследование логической функции «ИЛИ»

а) Экспериментальное получение таблицы истинности логического элемента «ИЛИ»

Соберите схему, изображенную на рис. 3.а.

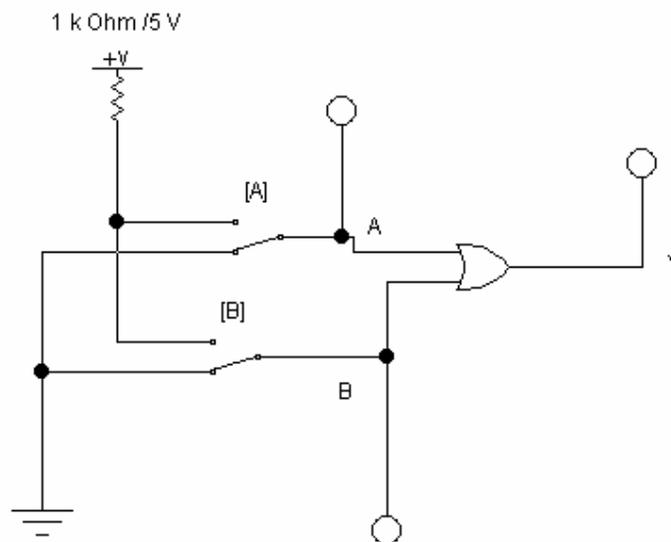


Рис. 3.а – Логическая функция «ИЛИ»

Включите схему. Подайте на входы схемы все возможные комбинации уровней входных сигналов и, наблюдая уровни сигналов на входах и выходе с помощью логических пробников, заполните таблицу истинности логической схемы «ИЛИ» (табл. 3.а).

Таблица 3.а – Таблица истинности элемента «ИЛИ»

Входы		Выход
А	В	У
0	0	
0	1	
1	0	
1	1	

б) *Получение аналитического выражения для функции*

По таблице 3.а составьте аналитическое выражение функции «ИЛИ» и занесите его в отчет. Для этого можно воспользоваться одним из двух способов получения логических выражений по таблице истинности изложенных в краткой теории к данной лабораторной работе.

Задание 4. Исследование логической функции «ИЛИ-НЕ»

а) *Экспериментальное получение таблицы истинности логического элемента «ИЛИ-НЕ», составленного из элементов «ИЛИ» и «НЕ»*

Соберите схему, изображенную на рис. 4.а.

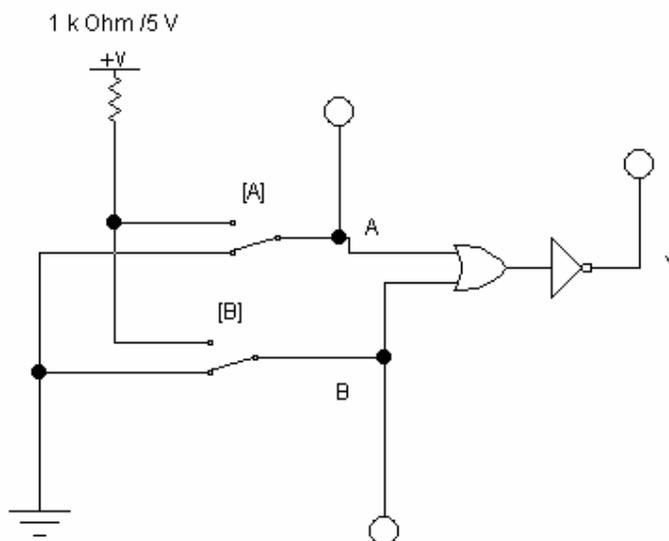


Рис. 4.а – Логическая функция «ИЛИ-НЕ»

Включите схему. Подайте на входы схемы все возможные комбинации уровней входных сигналов и, наблюдая уровни сигналов на входах и выходе с помощью логических пробников, заполните таблицу истинности логической схемы «ИЛИ-НЕ» (табл. 4.а).

Таблица 4.а – Таблица истинности элемента «ИЛИ-НЕ» (составного)

Входы		Выход
А	В	У
0	0	
0	1	
1	0	
1	1	

б) Экспериментальное получение таблицы истинности логического элемента «ИЛИ-НЕ»

Соберите схему, изображенную на рис. 4.б.

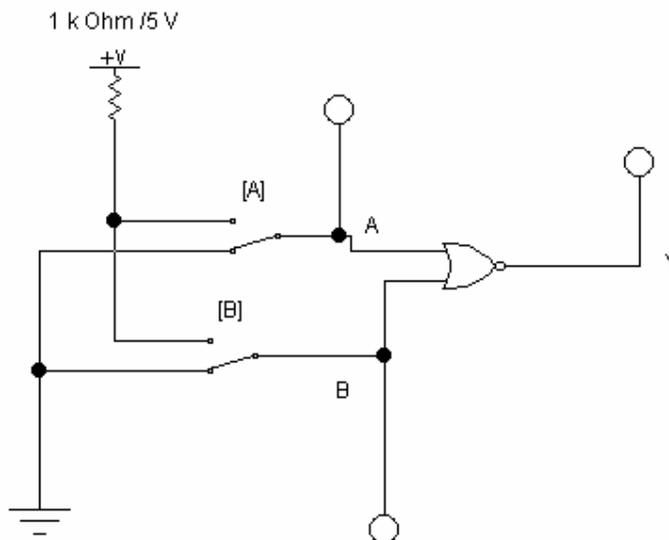


Рис. 4.б – Логическая функция «ИЛИ-НЕ»

Включите схему. Подайте на входы схемы все возможные комбинации уровней входных сигналов и, наблюдая уровни сигналов на входах и выходе с помощью логических пробников, заполните таблицу истинности логической схемы «ИЛИ-НЕ» (табл. 4.б).

Таблица 4.б – Таблица истинности элемента «ИЛИ-НЕ»

Входы		Выход
A	B	Y
0	0	
0	1	
1	0	
1	1	

Сравните таблицы 4.а и 4.б между собой и сделайте соответствующие выводы.

Задание 5. Исследование логических схем с помощью генератора слов

Анализ различных логических функций иногда удобно проводить с помощью микросхем реализующих различные логические элементы. Кроме того, использование микросхем (в отличие от расширенного представления логических функций, рассматриваемого в предыдущих заданиях) бывает необходимо на стадии проектирования сложных систем, когда в руках современного инженера зачастую имеется набор из различных, законченных модулей, реализующих вполне определенные функции.

Отдавая дань доступности и широкому распространению англоязычных микросхем, а так же их аналогов, изготовленных по западным стандар-

там (в том числе и современные российские разработки) в данной работе рассматриваются именно микросхемы западного стандарта.

Помимо этого для дальнейшего рассмотрения цифровой техники удобно вместо коммутирующих выключателей (задающих уровни логических сигналов) в некоторых случаях использовать «генератор слов» (прибор, предназначенный для выработки последовательности логических сигналов на своих выходах, причем данную последовательность пользователь имеет возможность задавать самостоятельно). Именно использованию законченных микросхем и «генератора слов» посвящено следующее задание.

а) Исследование микросхем различных типов

Выбор исследуемой микросхемы производится по номеру варианта, присвоенному студенту преподавателем. Типы микросхем в соответствии с вариантом задания приведены в таблице 5.а.

Примечание: Данные микросхемы в общем случае могут содержать несколько независимых друг от друга логических элементов (но одного типа, например, 3 логических элемента «И» – «AND»), причем у каждого логического элемента может быть и несколько входов (1, 2, 3 или 4).

Таблица 5.а – Варианты задания исследуемой микросхемы

Вариант задания	Тип микросхемы
1	AND/7408
2	AND/7421
3	AND/7411
4	NOR/7402
5	NOR/7427
6	NOR/7428
7	NOR/7433
8	OR/7432
9	XOR/7486
10	XNOR/74266

После выбора соответствующей микросхемы необходимо собрать схему для ее изучения. Данная схема в общем случае должна содержать: источник напряжения, заземление (земля), необходимое количество логических пробников и генератор слов. Пример законченной схемы для исследования микросхемы NOR/7400 приведен на рис. 5.а.

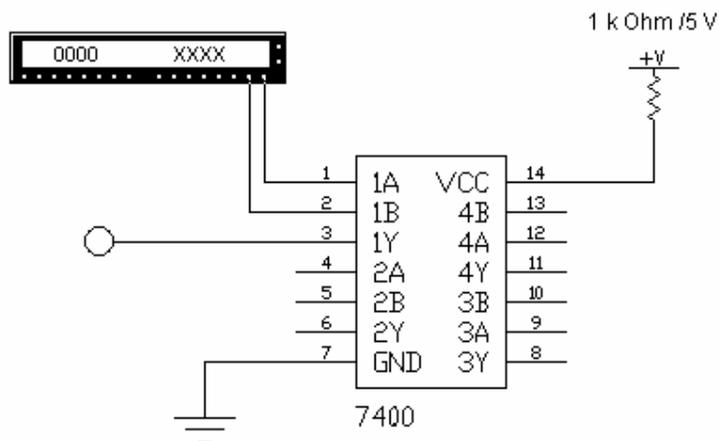


Рис. 5.а – Исследование микросхемы NOR/7400

Примечание: для дальнейшего исследования необходимо принимать во внимание только входы/выходы одного логического элемента на данной микросхеме (на рисунке 5.а используется первый логический элемент).

В отчете по данному пункту необходимо заполнить таблицу 5.а, дающую исчерпывающую информацию об используемой микросхеме.

Таблица 5.а – Информация о микросхеме

Тип микросхемы (полное обозначение по варианту задания)	
Тип базисных элементов (логических функций)	
Число базисных элементов в микросхеме (всего)	
Число исследуемых базисных элементов в микросхеме	
Обозначение выводов микросхемы используемых для подключения источника питания (номера и название)	
Обозначения выводов микросхемы используемых для подключения заземления (номера и название)	
Обозначение используемых входов (номера и название)	
Обозначение используемых выводов (номера и название)	

Примечание: исчерпывающую информацию об используемой микросхеме можно получить в справке (выделите интересующий вас элемент и щелкните на знак вопроса).

б) Экспериментальное получение таблицы истинности микросхемы

Для дальнейшего изучения микросхемы необходимо использование «генератора» слов (предполагается, что в пункте 5.а была собрана схема для изучения). Общий вид открытого окна свойств «генератора» показан на рис. 5.б. Для открытия данного окна необходимо дважды щелкнуть мышью на изображении генератора слов.

Окно свойств может быть условно разбито на ряд частей.

В левой части, в прокручивающемся окне, отображаются слова (набор логических последовательностей) формируемые «генератором слов» в 16-

ричной системе исчисления (данные последовательности могут быть изменены пользователем).

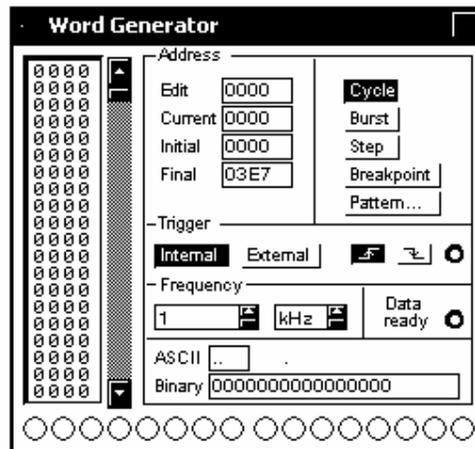


Рис. 5.б – Генератор слов

В правой части окна свойств находится ряд областей («Address», «Trigger», «Data ready»), несущих техническую информацию. Из этих областей мы будем пользоваться только кнопками «Cycle» и «Step», переводящий генератор слов в цикловой (заданные пользователем слова прокручиваются автоматически) и пошаговый режим работы (каждая следующая последовательность, вырабатываемого «генератором слов», вызывается нажатием кнопки «Step»).

В правой нижней части окна свойств находятся поля «ASCII», «Binary» и ряд, состоящий из логических пробников. Поле «Binary» (изменяемое) и ряд пробников (функционирующих только в режиме «Работа») несут одинаковую информационную сущность. Они отображают генерируемые последовательности слов в двоичном коде.

Отметим так же, что часто бывает удобнее задавать последовательность слов именно в поле «Binary», которое связано (изменение его вызывает изменение другого поля) и с прокручиваемым списком генерируемых слов, находящимся в левой части и рассмотренным ранее.

Для выполнения работы запрограммируйте «генератор слов» так, чтобы на его выходе получил последовательно следующие комбинаций: 00, 01, 10, 11 (необходимо помнить что, несмотря на то, что в анализируемых микросхемах количество базисных логических элементов и входов на каждом элементе множество, мы исследуем только два входа одного базисного логического элемента). Затем переведите генератор в режим пошаговой работы нажатием кнопки «Step» в окне свойств генератора. Каждое нажатие кнопки «Step» вызывает переход к очередному слову заданной последовательности, которое подается на выход генератора. Последовательно подавая на микросхему слова из заданной последовательности, заполните таблицу истинности элемента вашего базисного элемента (табл. 5.б).

Таблица 5.6 – Таблица истинности исследуемой микросхемы

Входы		Выход
1	2	Y
0	0	
0	1	
1	0	
1	1	

Указание: Для повторного прогона запрограммированной комбинации слов можно выключить и включить схему заново, тогда последовательности, заданные в «генераторе слов» будут изменяться, начиная с первой.

Задание 6. Реализация логической функции 3-х и большего числа переменных

В заключение изучения базовых логических схем и функций приведем задание, иллюстрирующее возможности «Electronic Workbench» в такой непростой и трудно формализуемой задаче как синтез логических схем и функций. Все предыдущие задания опирались в основном на моделирование уже существующих логических функций и изучение их свойств. Выполнив же это задание, студенты сами смогут автоматически реализовывать любую логическую функцию, заданную булевым выражением в различных базисах, получать таблицы истинности логических выражений и т. п. Все выше перечисленные процедуры можно реализовать с помощью инструмента «Logic Converter». Общий вид окна свойств «Logic Converter» показан на рис. 6. Напомним, что окно свойств вызывается двойным нажатием на изображении инструмента. Рассмотрим окно свойств подробнее.

Здесь в верхней части под буквами А, В, ..., Н находятся индикаторы, активизируемые путем нажатия на них.

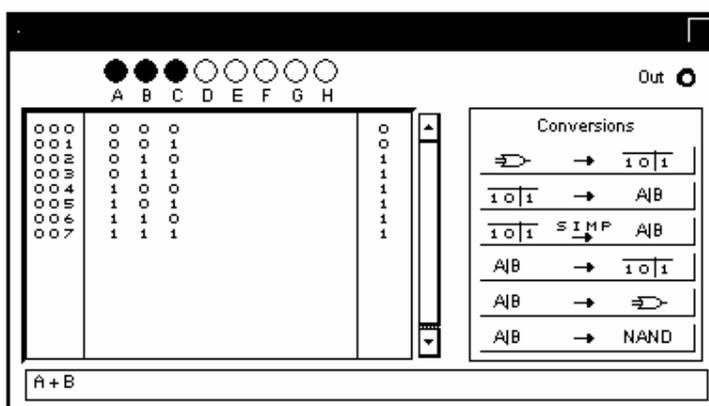


Рис. 6 – «Logic Converter»

На рис. 6 активизированы три индикатора А, В, С. Это подразумевает то, что в дальнейшем мы будем оперировать логической функцией, состоящей из трех переменных: А, В, С. Данный выбор вызывает появление всех

возможных комбинаций этих переменных (строки таблицы, изображенной на рис. 6, начиная с нулевой и заканчивая 7). В правой части полученной таблицы истинности, в отдельном столбике по умолчанию присвоены «0» выходной переменной. Пользователь сам, изменяя значения выхода анализируемой логической функции, тем самым полностью задает таблицу истинности.

В самом нижнем окне находится поле ввода/вывода логической функции (поле является выводящим, если пользователь сам задал таблицу истинности, а затем путем нажатия на соответствующую функциональную кнопку потребовал формирование логической функции; и поле является вводящим, если пользователь предварительно сам задал логическую функцию, а затем путем нажатия на соответствующую функциональную кнопку потребовал формирование таблицы истинности).

В правой части окна свойств находятся соответствующие функциональные кнопки (панель «Conversion»).

1)  →

1	0	1
---	---	---

 - Получение таблицы истинности для произвольной логической схемы, подсоединенной к входам и выходам «Logic Converter»;

2)

1	0	1
---	---	---

 → $A|B$ - Получение по таблице истинности логической функции (не упрощенной);

3)

1	0	1
---	---	---

 $\xrightarrow{\text{SIMP}}$ $A|B$ - Получение по таблице истинности упрощенной логической функции (аналогичная функция получается путем использования для анализа таблицы истинности карт Карно);

4) $A|B$ →

1	0	1
---	---	---

 - Получение по логической функции таблицы истинности;

5) $A|B$ →  - Получение по логической функции схемы, реализованной в произвольном базисе

6) $A|B$ → NAND - Получение по таблице истинности логической функции реализованной в базисе «И-НЕ».

a) Синтез схемы, реализующей заданную функцию при помощи логического преобразователя

Для получения схемы, реализующей функцию, описываемую логическим выражением f (задается преподавателем), можно воспользоваться логическим преобразователем. Варианты заданий логических выражений приведены в таблице 6.а (здесь апостроф указывает на операцию инвертирования).

Таблица 6.а – Варианты задания логического выражения f

Вариант задания	Логическое выражение f
1	$A+B*C$
2	$C+A*B$
3	$A*B*C$
4	$A'+B+C$
5	$A+B'C$
6	$C*B'+A$
7	$A'+B'+C'$
8	$C'+A'*B$
9	$A'+B'+C$
10	$B+C*A'$

Проделайте с анализируемым логическим выражением следующие действия:

- вызовите логический преобразователь;
- введите в нижнее окно панели преобразователя логическое выражение с клавиатуры (операции ИЛИ соответствует знак «+», инверсия обозначается апострофом, логическая операция умножения не вводится, например $AB=A*B$);
- для реализации схемы на элементах «И-НЕ» нажмите соответствующую функциональную кнопку на панели логического преобразователя.

Занесите полученную схему в отчет.

Удалите полученную схему.

- для простой реализации схемы на произвольных элементах нажмите соответствующую функциональную кнопку на панели логического преобразователя.

Занесите полученную схему в отчет.

Удалите полученную схему.

- для получения таблицы истинности заданной логической функции нажмите соответствующую функциональную кнопку на панели логического преобразователя.

Занесите полученную таблицу истинности в отчет.

б) Синтез логической функции, реализующей заданную таблицу истинности при помощи логического преобразователя

Для получения функции, реализующей таблицу истинности, необходимо воспользоваться таблицей истинности полученной, в предыдущем пункте. Для этого сделайте следующее:

- удалите с нижней строки логического преобразователя логическую функцию;
- очистите таблицу истинности преобразователя, нажав на его верхнюю функциональную кнопку;

- выберите в поле таблицы истинности количество используемых логических переменных (в нашем случае их 3), путем нажатия на соответствующие индикаторы;
- в правой колонке полученной таблицы истинности задайте значение искомой логической функции в соответствии с каждой строкой;
- для получения не упрощенной логической функции нажмите соответствующую функциональную кнопку на панели логического преобразователя, занесите в отчет полученную функцию;
- для получения упрощенной логической функции нажмите клавишу соответствующую функциональную кнопку на панели логического преобразователя, запишите упрощенную логическую функцию.

Удостоверьтесь, что упрощенные модели совпадают с заданными в начале моделями;

Удостоверьтесь, что упрощенная функция может быть получена из не упрощенной (проделайте операции упрощения).

Контрольные вопросы

1. Дайте определение логического сигнала.
2. Дайте определение логической переменной.
3. Дайте определение логической функции.
4. Какие значения могут принимать булевы переменные?
5. Приведите основные логические тождества.
6. Что может быть принято за уровни логических сигналов?
7. Подумайте, почему часто в технике за уровень логического нуля принимают физический сигнал (например, по току) отличный от нуля.
8. Как может быть получена логическая функция?
9. Чем в физическом смысле отличается работа схемы составленной по упрощенной логической функции от неупрощенной?
10. Сколько различных комбинаций сигналов надо подать на схему, имеющую 4 входа, для составления таблицы истинности?
11. Какой сигнал должен быть подан на неиспользуемые входы элемента «И-НЕ» имеющего 5 входов, если требуется реализовать ту же логическую функцию, но на 3 входы?
12. Какой сигнал должен быть подан на неиспользуемые входы элемента «ИЛИ» имеющего 5 входов, если требуется реализовать ту же логическую функцию, но на 4 входы?
13. Какой сигнал нужно подать на неиспользуемые входы элемента «И» имеющего 2 входы для реализации на его базе инвертора на один входной сигнал.
14. Какой логической функцией можно описать систему пуска трехфазного двигателя, если двигатель может быть запущен, когда три датчика подтверждают наличие фазных напряжений?

ЛАБОРАТОРНАЯ РАБОТА № 2

Изучение работы шифраторов, дешифраторов и мультиплексоров

Цель работы:

1. Изучение принципов работы шифраторов, дешифраторов и мультиплексоров.
2. Реализация логических функций с помощью мультиплексоров.
3. Изучение способов применения дешифраторов.

Приборы и элементы:

Генератор слов (панель «Instruments/Word Generator»)

Логический анализатор (панель «Instruments/Logic Analyzer»)

Логические пробники (панель «Indicators/Red probe»)

Источник напряжения + 5 В (панель «Basic/Pull-Up Resistor»)

Земля (панель «Sources/Ground»)

Двухвходовые элементы И, И-НЕ, ИЛИ, ИЛИ-НЕ (панель «Logic Gates/2-Input AND, NAND, OR, NOR Gates»)

Двухпозиционные переключатели (панель «Basic/Switch»)

Дешифратор (панель «Digital/DEC/Generic 8-to-1 DEMUX»)

Мультиплексор (панель «Digital/MUX/ Generic 1-of 8 MUX»)

Краткие теоретические сведения

1. Комбинационные схемы

Комбинационной схемой называется логическая схема, реализующая однозначное соответствие между значениями входных и выходных сигналов. Для реализации комбинационных схем используются логические элементы, выпускаемые в виде интегральных схем. В этот класс входят интегральные схемы дешифраторов, шифраторов, мультиплексоров, демультиплексоров и сумматоров.

2. Шифраторы

Шифратор – логическая комбинационная схема, которая имеет 2^n входов (где n – число информационных выходов). Часто $n=3$, тогда $2^n=8$. Подаче на один из входов активного сигнала будет соответствовать двоичное число, которое можно сформировать из его n выходов, эквивалентное номеру входа, на котором появился активный уровень.

Примечание: формирование двоичного числа на выходе шифратора означает следующее: каждый из выходов шифратора считается определенным разрядом искомого двоичного числа. Принцип работы шифратора противоположен принципу работы дешифратора, который подробно рассмотрен ниже.

Простейшим случаем применения шифратора может служить, например, схема отслеживания нажатия одной кнопки. Каждая кнопка представляет собой элементарный переключатель. Пусть таких кнопок всего 8, и они пронумерованы начиная с «0» и заканчивая «7». При нажатии определенной кнопки (например «3») формируется сигнал на входе шифратора (каждый вход – это вполне определенная кнопка), в итоге на выходе шифратора мож-

но получить двоичный сигнал равный номеру нажатой кнопки. В нашем случае число выходов равно 3 ($2^3=8$), и на каждом из выходов получится следующая комбинация на 3-м = 0, на 2-м = 1, на 1-м = 1. Полученное двоичное число «011», которое в десятичном коде равно 3.

3. Дешифраторы

Дешифратор – логическая комбинационная схема, которая имеет n информационных входов и 2^n выходов. Каждой комбинации логических уровней на входах будет соответствовать активный уровень на одном из 2^n выходов. Обычно n равно 2, 3 или 4. В отечественной литературе входы дешифратора принято обозначать (1, 2, 4, 8, ...), в англоязычной (A, B, C, \dots). Работа дешифратора может быть проиллюстрирована в соответствии с таблицей истинности 1.

Таблица 1 – Таблица истинности дешифратора

Входы				Выходы									
8	4	2	1	0	1	2	3	4	5	6	7	8	9
0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0	0	0
0	0	1	1	0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0	1	0	0	0	0	0
0	1	0	1	0	0	0	0	0	1	0	0	0	0
0	1	1	0	0	0	0	0	0	0	1	0	0	0
0	1	1	1	0	0	0	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	0	0	0	0	1

На рис. 1 представлен дешифратор, имеющий таблицу истинности 1. Условное обозначения базовых элементов соответствует обозначению, принятому в отечественной литературе.

В дальнейшем для детального рассмотрения дешифратора, мы будем пользоваться обозначением (микросхем и базовых элементов) принятых в ЕWB. На рис. 2 изображен дешифратор (блок микросхемы, использующийся для моделирования дешифратора в ЕWB) с $n = 3$.

Активным уровнем сигнала данной микросхемы является уровень логического нуля. То есть в отличие от таблицы истинности 1, у данного дешифратора на выходах, по диагонали расположены нули, а остальные элементы равны единице. На входы C, B, A можно подать следующие комбинации логических уровней: «000», «001», «010», ..., «111», всего 8 комбинаций. Схема имеет 8 выходов, на одном из которых формируется низкий потенциал, на остальных - высокий. Номер этого единственного выхода, на котором формируется активный (нулевой) уровень, соответствует числу N , определяемому состоянием входов C, B, A следующим образом:

$$N = C \cdot 2^2 + B \cdot 2^1 + A \cdot 2^0 \quad (1)$$

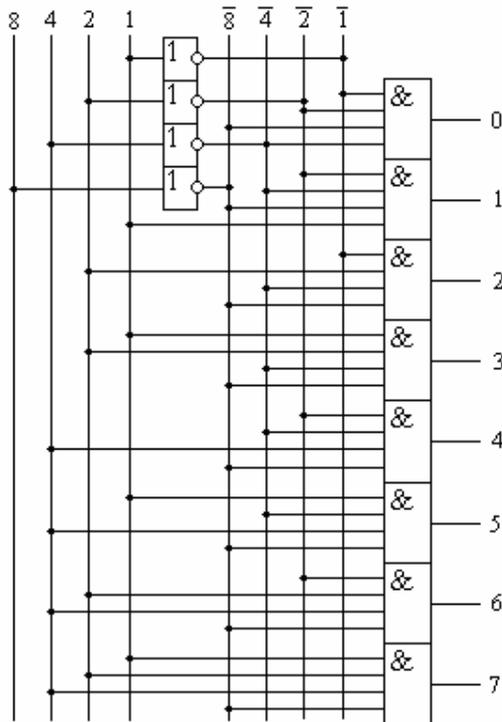


Рис. 1 – Дешифратор

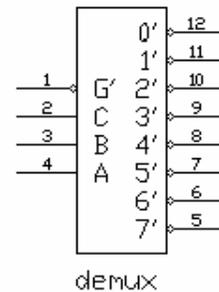


Рис. 2 – Условное обозначение

Например, если на входы подана комбинация логических уровней «011», то из восьми выходов микросхемы (рис. 2) на выходе с номером $N=3$ (в двоичном исчислении $3 = 011$) установится нулевой уровень сигнала ($Y_3=0$), а все остальные выходы будут иметь уровень логической единицы. Этот принцип формирования выходного сигнала можно описать следующим образом:

$$Y_i = \begin{cases} 0, & \text{если } i = k, \\ 1, & \text{если } i \neq k, \\ k = 2^2 \cdot C + 2^1 \cdot B + 2^0 \cdot A. \end{cases} \quad (2)$$

Помимо информационных входов A, B, C дешифраторы обычно имеют дополнительные входы управления G . Сигналы на этих входах, например, разрешают функционирование дешифратора или переводят его в пассивное состояние, при котором, независимо от сигналов на информационных входах, на всех выходах установится уровень логической единицы. Можно сказать, что существует некоторая функция разрешения, значение которой определяется состояниями управляющих входов.

Разрешающий вход дешифратора может быть прямым или инверсным. У дешифраторов с прямым разрешающим входом активным уровнем является уровень логической единицы, у дешифраторов с инверсным входом – уровень логического нуля. На рис. 2 представлен дешифратор с одним инверсным входом управления. Принцип формирования выходного сигнала в этом дешифраторе с учетом сигнала управления описывается следующим образом:

$$Y_i = \begin{cases} \overline{1 \cdot \overline{G}}, & \text{если } i = k, \\ 1, & \text{если } i \neq k, \\ k = 2^2 \cdot C + 2^1 \cdot B + 2^0 \cdot A. \end{cases} \quad (3)$$

У дешифратора с несколькими входами управления функция разрешения, как правило, представляет собой логическое произведение всех разрешающих сигналов управления. Например, для дешифратора серии 74138 с одним прямым входом управления $G1$ и двумя инверсными $G2A$ и $G2B$ (Рис. 3) функции выхода Y_i и разрешения G имеют вид:

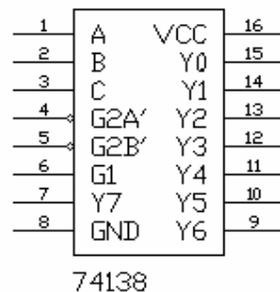


Рис. 3

$$Y_i = \begin{cases} \overline{1 \cdot \overline{G}}, & \text{если } i = k, \\ 1, & \text{если } i \neq k, \\ k = 2^2 \cdot C + 2^1 \cdot B + 2^0 \cdot A. \end{cases} \quad (4)$$

$$G = G1 \cdot \overline{G2A} \cdot \overline{G2B} \quad (5)$$

Обычно входы управления используются для каскадирования (увеличения разрядности) дешифраторов или при параллельной работе нескольких схем на общие выходные линии.

Дешифратор может быть использован как *демультиплексор* – логический коммутатор, подключающий входной сигнал к одному из выходов. В этом случае функцию информационного входа выполняет один из входов разрешения, а состояние входов C , B и A задает номер выхода, на который передается сигнал с входа разрешения.

4. Мультиплексоры

Мультиплексор – комбинационная логическая схема, представляющая собой управляемый переключатель, который подключает к выходу один из входов данных. Номер подключаемого входа равен числу (адресу), определяемому комбинацией логических уровней на адресных входах.

Демультиплексорами – называются устройства, которые позволяют подключать один вход к нескольким выходам.

В простейшем случае переключения (коммутацию) можно осуществить при помощи ключей, как это показано на рис. 4.

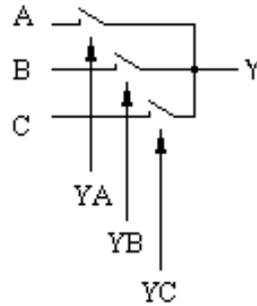


Рис. 4 – Мультиплексор на ключах.

В цифровых схемах управление ключами осуществляется при помощи логических сигналов. Сами ключи при этом, заменяются соответствующими логическим элементами.

Рассмотрим пример простейшей схемы мультиплексора. Для этого воспользуемся базовым логическим элементом «И» с таблицей истинности 2.

Таблица 2 – Таблица истинности элемента «И»

Входы		Выход
X	Y	Out
0	0	0
0	1	0
1	0	0
1	1	1

Теперь один из входов элемента будем рассматривать как информационный вход электронного ключа, а другой вход – как адресный. По таблице истинности отчетливо видно, что пока на адресный вход Y подан логический уровень «0» сигнал с входа X на выход Out не проходит. При подаче на адресный вход Y логической «1», сигнал, поступающий на вход X , поступает на выход Out . То есть логический элемент «И» можно использовать в качестве электронного ключа. При этом не важно, какой из входов элемента будет использоваться в качестве адресного входа, а какой – в качестве информационного. Остаётся только объединить выходы элементов «И» на один выход. Это делается при помощи элемента «ИЛИ». Условное обозначение такой схемы приведено на рис. 5.

Чаше всего для управления требуется много входов, поэтому в схему мультиплексора включают дешифратор. Это позволяет управлять переключением входов микросхемы на выход при помощи двоичных кодов. Пример такой схемы приведен на рис. 6.

Для дальнейшего рассмотрения мультиплексоров мы будем, как и прежде пользоваться условными обозначениями логических элементов и микросхем, принятых в EWB.

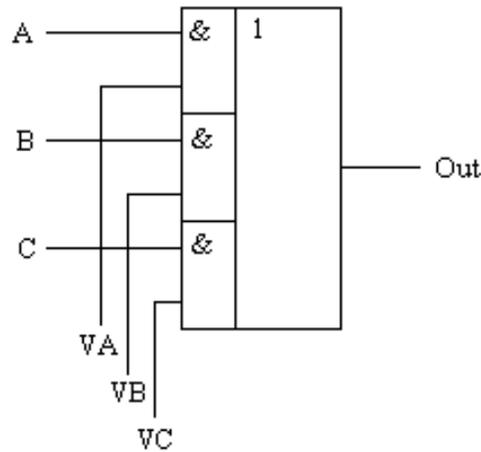


Рис. 5 – Принципиальная схема мультиплексора.

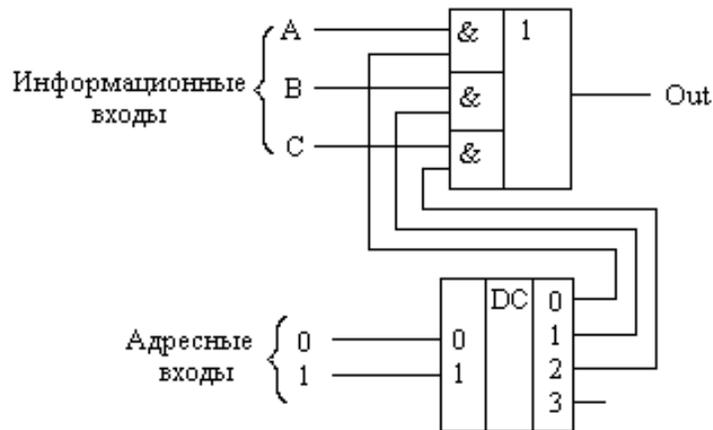


Рис. 6 – Мультиплексор, управляемый двоичным кодом.

Кроме информационных и адресных входов, схемы мультиплексоров содержат вход разрешения, при подаче на который активного уровня мультиплексор переходит в рабочее состояние. При подаче на вход разрешения пассивного уровня мультиплексор перейдет в нерабочее состояние, в котором сигнал на выходе сохраняет постоянное значение независимо от значений информационных и управляющих сигналов. Число информационных входов у мультиплексоров обычно равно 2, 4, 8 или 16.

На рис. 7 представлен мультиплексор (блок микросхемы, использующийся для моделирования дешифратора в EWB) 8x1 с инверсным входом разрешения G , прямым Y и инверсным W -выходами ($W = \bar{Y}$)

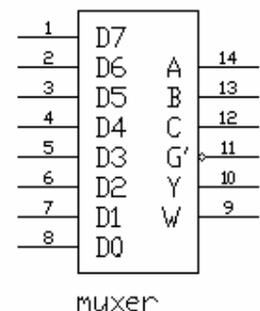


Рис. 7

5. Реализация логических функций

Функционирование мультиплексора, представленного на рис. 7, описывается характеристическим уравнением, связывающим сигнал на выходе (Y) с разрешающим (G), входными информационными ($D0...D7$) и адресными (A, B, C) сигналами:

$$Y = \left(\begin{array}{l} \bar{C} \cdot \bar{B} \cdot \bar{A} \cdot D0 + \bar{C} \cdot \bar{B} \cdot A \cdot D1 + \bar{C} \cdot B \cdot \bar{A} \cdot D2 + \bar{C} \cdot B \cdot A \cdot D3 + \\ + C \cdot \bar{B} \cdot \bar{A} \cdot D4 + C \cdot \bar{B} \cdot A \cdot D5 + C \cdot B \cdot \bar{A} \cdot D6 + C \cdot B \cdot A \cdot D7 \end{array} \right) \cdot \bar{G} \quad (6)$$

Как видно из уравнения, на мультиплексоре можно реализовать логические функции, для чего нужно определить, какие сигналы и логические константы следует подавать на входы мультиплексора.

Логическая функция n переменных определена для 2^n комбинаций значений переменных. Это позволяет реализовать функцию n переменных на мультиплексоре, имеющем n адресных и 2^n информационных входов. В этом случае каждой комбинации значений аргументов соответствует единственный информационный вход мультиплексора, на который подается значение функции.

Например, требуется реализовать функцию:

$$F_1 = \bar{c} \cdot \bar{b} \cdot \bar{a} \vee c \cdot b \cdot a \vee c \cdot b \cdot \bar{a} \vee \bar{c} \cdot b \cdot a \quad (7)$$

Эта функция определена только для 8 комбинаций значений переменных, поэтому для её реализации можно использовать мультиплексор 8×1 с тремя адресными входами. Составим таблицу истинности функции:

Таблица 3 – Таблица истинности логической функции F_1

Вход мультиплексора	Логические переменные			Выход функции
	с	b	а	
N				F_1
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

Из таблицы 3 видно, что для реализации функции на мультиплексоре необходимо подать на информационный вход мультиплексора с номером N сигнал, значение которого равно соответствующему значению функции F_1 , т. е. на входы с номерами 1, 2, 4, 5 следует подать уровень логического нуля, а на остальные - уровень логической единицы. Таким образом, при подаче комбинации логических уровней (a, b, c) на адресные входы мультиплексора,

к его выходу подключится вход, значение сигнала на котором равно соответствующему значению функции. Схемная реализация приведена на рис. 8.

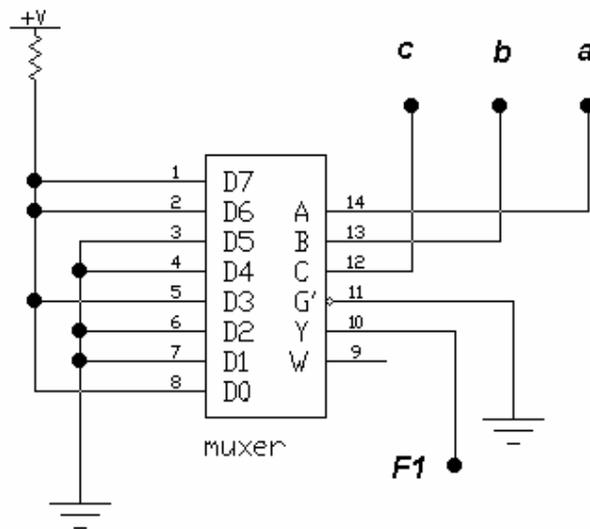


Рис. 8 – Реализация логической функции

При реализации логических функций на информационные входы можно подавать не только константы, но и изменяющиеся входные сигналы. Так, например, рассмотрим другой способ реализации функции F_1 , определенной выражением (7). Для этого минимизируем выражение функции с помощью известных логических тождеств (см. лабораторную работу № 1) до вида:

$$F_1 = \bar{c} \cdot \bar{b} \cdot \bar{a} \vee b \cdot c \vee b \cdot a \quad (8)$$

Составим таблицу истинности функции (8) в зависимости от значений переменных a и b (см. таблицу 4).

Для составления таблицы в выражение (8) подставлялись комбинации a и b и, пользуясь логическими тождествами (см. лабораторную работу № 1) получалось значение функции F_1 . Заданную такой таблицей функцию реализуют, как и в предыдущем случае, подав на вход с номером N сигнал, значение которого соответствует значению функции F_1 .

Таблица 4 – Таблица истинности упрощенной логической функции F_1

Вход мультимплексора	Логические переменные		Выход функции
	b	a	
N			F_1
0	0	0	\bar{c}
1	0	1	0
2	1	0	c
3	1	1	1

В данном случае сигналы c и \bar{c} соответствующие переменной c , подаются на информационные входы, как указано в таблице истинности. При этом сокращается число управляющих входов.

Схемная реализация такого способа задания функции приведена на рис. 9.

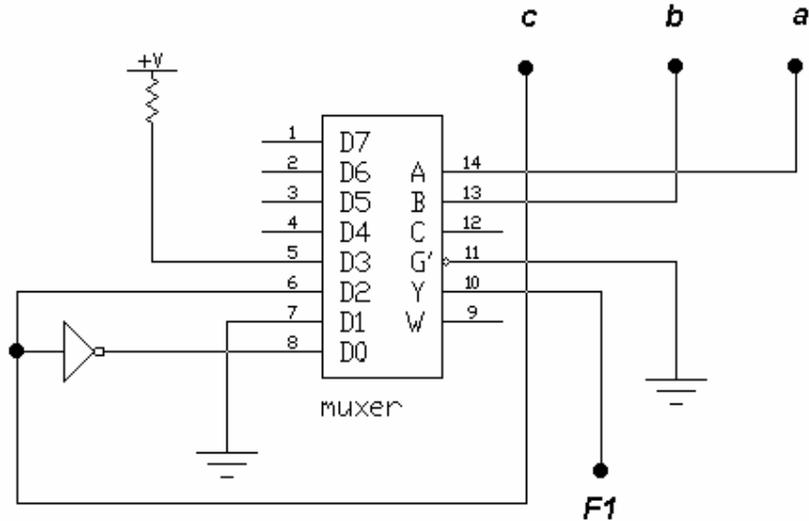


Рис. 9 – Реализация упрощенной логической функции

Так как используются только два адресных входа, вход C остается не подключенным. При этом состояние информационных входов $D4...D7$ безразлично. Схема рис. 9 по существу представляет собой мультиплексор 4×1 с двумя адресными и четырьмя информационными входами.

Если функцию можно представить в виде произведения одночлена на многочлен, то её также можно реализовать при помощи мультиплексора. Как следует из уравнения мультиплексора, сигнал, соответствующий одночлену, нужно подать на вход разрешения. Например, требуется реализовать функцию F_2 , описываемую следующим выражением:

$$F_2 = \bar{x} \cdot (d \cdot c \cdot \bar{b} \vee d \cdot \bar{b} \cdot a \vee e \cdot \bar{c} \cdot b \cdot a \vee c \cdot b \cdot a) \quad (9)$$

При реализации данной функции на мультиплексоре сигнал, соответствующий переменной x , следует подать на его разрешающий вход. Рассмотрим, какие сигналы необходимо подать на управляющие входы мультиплексора. Выражение в скобках можно рассматривать как некоторую функцию f пяти переменных: a, b, c, d, e , из которых наиболее часто используются переменные a, b и c . Поэтому сигналы, соответствующие этим переменным, нужно подать на адресные входы мультиплексора.

Определим, какие сигналы следует подать на информационные входы, чтобы реализовать функцию f . Для этого составим таблицу истинности функции в зависимости от значений переменных a, b и c (таблица 5).

Таблица 5 – Таблица истинности логической функции F_2

Вход мультиплексора	Логические переменные			Выход функции
	c	b	a	
N	c	b	a	f
0	0	0	0	0
1	0	0	1	d
2	0	1	0	0
3	0	1	1	e
4	1	0	0	d
5	1	0	1	d
6	1	1	0	0
7	1	1	1	1

Из таблицы видно, что на информационные входы с номерами $N = 0, 2, 6$ нужно подать уровень логического нуля. Сигнал, соответствующий переменной d , нужно подать на входы с номерами $N = 1, 4, 5$, сигнал, соответствующий переменной e , – на вход с номером 3. Схемная реализация такого способа задания функции приведена на рис. 10.

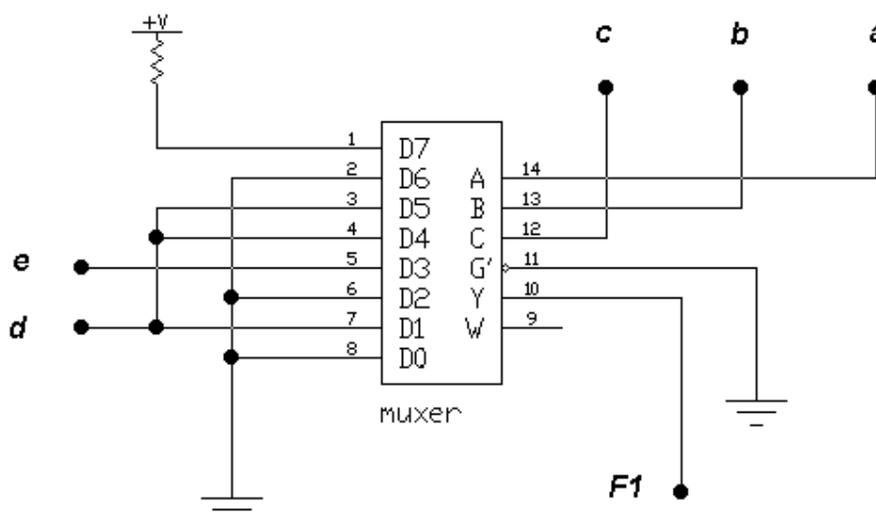


Рис. 10 – Реализация логической функции F_2

Порядок работы

Задание 1. Исследование работы шифратора
Создайте схему изображенную на рис. 1. а.

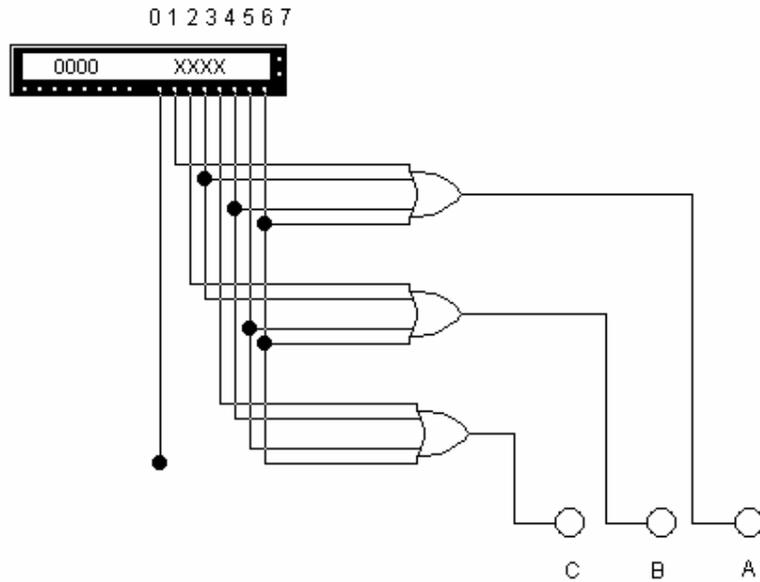


Рис. 1.а – Схема шифратора

Над генератором слов написаны цифры от 0 до 7 – они обозначают номера входов шифратора, на которые соответственно подаются сигналы управления. Сам шифратор составлен из трех элементов «ИЛИ». Выходы шифратора обозначаются как *A*, *B*, *C*. Где *A* – старший бит, *B* – средний бит, а *C* – младший бит двоичного числа, получаемого на выходе. (Это число показывает, на какой из входов подан логический сигнал).

Запрограммируйте генератор слов, так чтобы на его выходах сформировалась двоичная последовательность, эмулирующая поочередную подачу на вход шифратора сигнала логической единицы. Пошагово изменяя значения входов дешифратора (кнопкой «Step» в генераторе слов), заполните таблицу истинности шифратора (таблица 1.а).

Таблица 1.а – Таблица истинности шифратора

Входы шифратора								Выходы шифратора			Десятичное число
								Двоичное число			
0	1	2	3	4	5	6	7	C	B	A	
0	0	0	0	0	0	0	0				
0	1	0	0	0	0	0	0				
0	0	1	0	0	0	0	0				
0	0	0	1	0	0	0	0				
0	0	0	0	1	0	0	0				
0	0	0	0	0	1	0	0				
0	0	0	0	0	0	1	0				
0	0	0	0	0	0	0	1				

Переведите полученное двоичное число, составленное из разрядов *C B A* в десятичное. Сделайте вывод о работе шифратора.

Задание 2. Исследование работы дешифраторов

а) Исследование развернутой схемы дешифратора

Создайте схему изображенную на рис. 2. а.

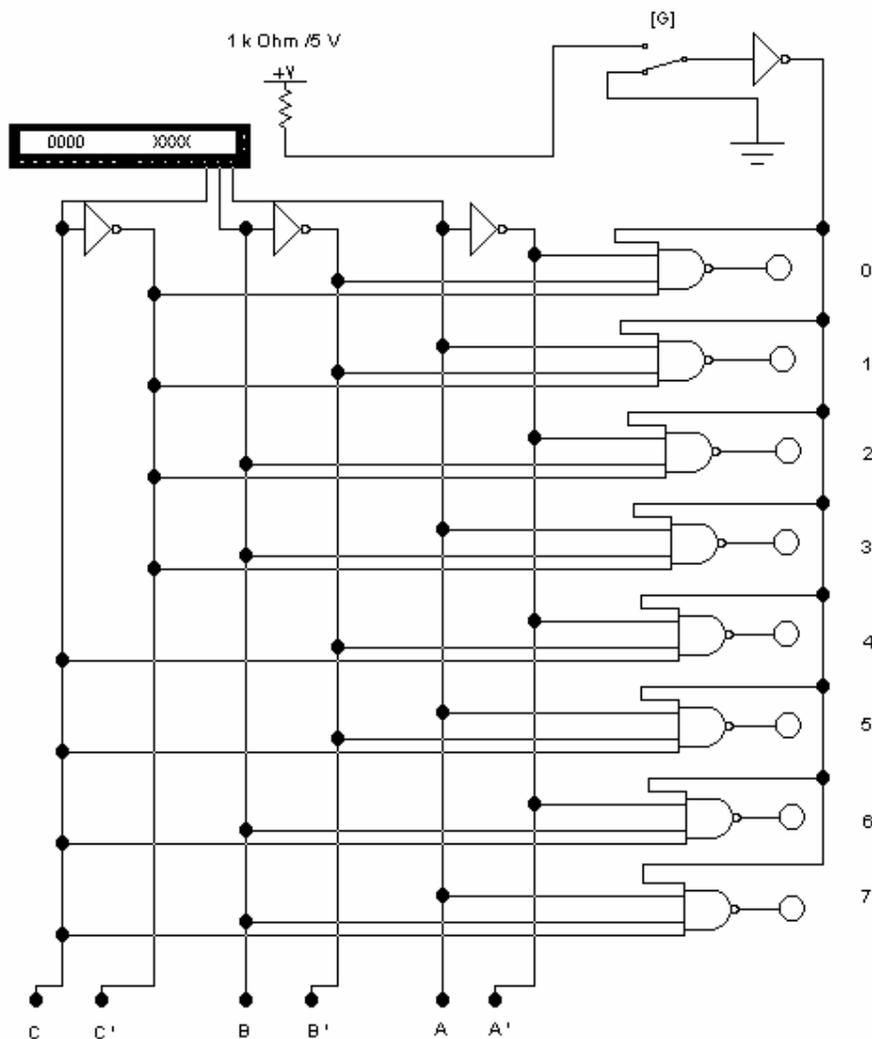


Рис. 2.а – Развернутая схема дешифратора

Здесь представлен дешифратор 3*8 (3 входа, 8 выходов). Дешифратор составлен из элементов «И-НЕ». C , B , A – входы дешифратора, 0, 1, ..., 7 – выходы дешифратора, G – вход разрешения.

Запрограммируйте генератор слов так, что бы на его выходе сформировать все возможные комбинации трехразрядного двоичного числа. Подавая на вход дешифратора различные комбинации двоичного числа C , B , A (кнопкой «Step» в генераторе слов) и разрешения G (ключом G), заполните таблицу истинности дешифратора (таблица 2.а).

Сделайте вывод о работе дешифратора.

Таблица 2.а – Таблица истинности развернутой схемы дешифратора

Входы дешифратора					Выходы дешифратора							
Число	С	В	А	Г	0	1	2	3	4	5	6	7
0	0	0	0	0								
1	0	0	1	0								
2	0	1	0	0								
3	0	1	1	0								
4	1	0	0	0								
5	1	0	1	0								
6	1	1	0	0								
7	1	1	1	0								
0	0	0	0	1								
1	0	0	1	1								
2	0	1	0	1								
3	0	1	1	1								
4	1	0	0	1								
5	1	0	1	1								
6	1	1	0	1								
7	1	1	1	1								

б) Исследование работы схемы дешифратора 3*8 в основном режиме
Создайте схему изображенную на рис. 2.б

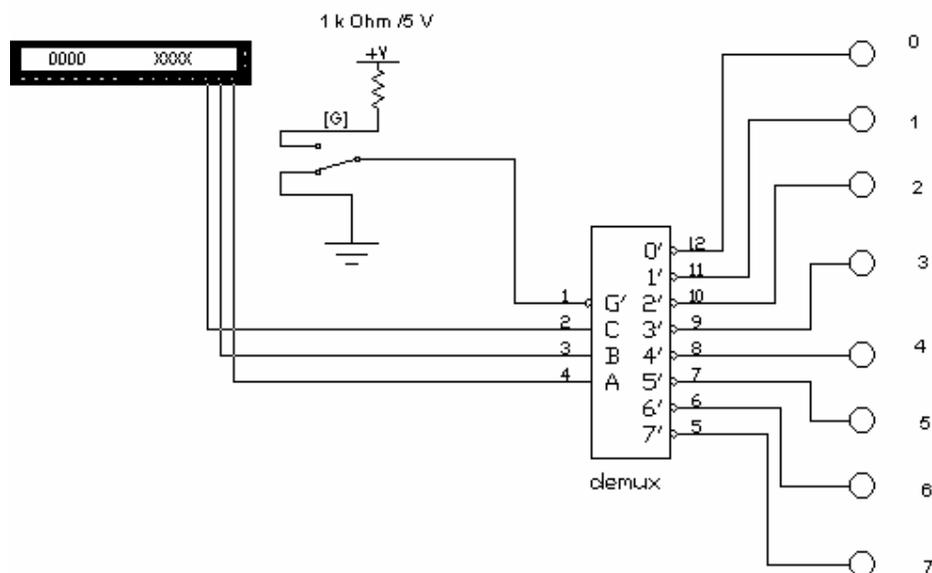


Рис. 2.б – Схема дешифратора

Запрограммируйте генератор слов аналогично пункту а. Подавая на вход дешифратора различные комбинации двоичного числа C, B, A (кнопкой «Step» в генераторе слов) и разрешения G (ключом G), заполните таблицу истинности дешифратора (таблица 2.б, аналогично таблице 2.а).

Сделайте вывод о работе дешифратора.

Сравните таблицы 2.а и 2.б.

в) Исследование работы схемы дешифратора 3*8 в режиме 2*4

Создайте схему изображенную на рис. 2.в.

В схеме рис. 2.в подключите вход C к общему проводу (земле), задав $C=0$. Изменяя сигналы на входах B и A , наблюдайте уровни сигналов на выходах схемы с помощью пробников. Укажите выходы, на которых уровень сигнала не меняется.

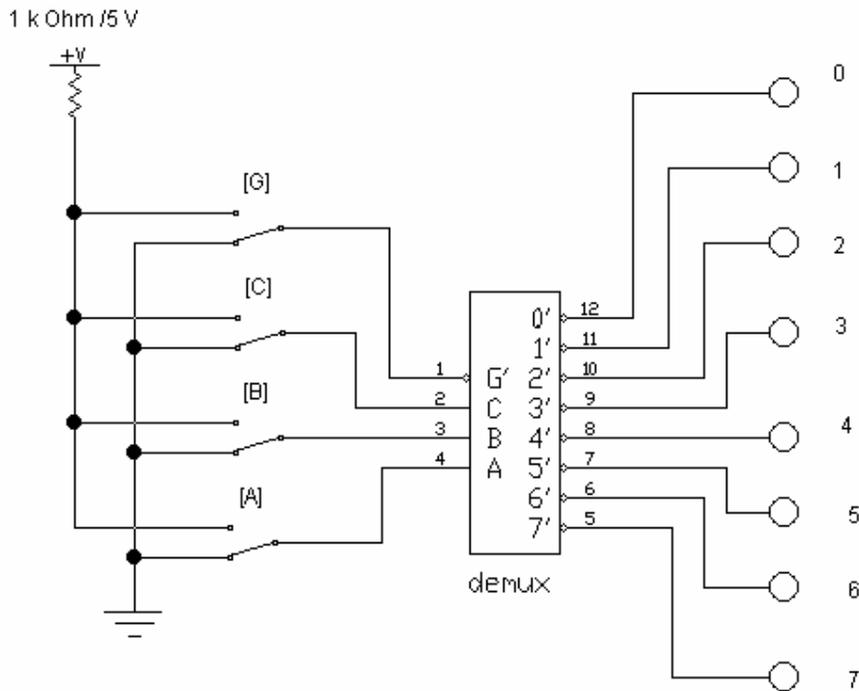


Рис. 2.в – Дешифратор в режиме 2*4

Подключите вход C к источнику питания (логической единицы) задав $C=1$. Аналогично изменяя сигналы на входах B и A , наблюдайте уровни сигналов на выходах схемы, с помощью пробников. Укажите выходы, на которых уровень сигнала не меняется.

Заземлите вход B ($B=0$), подавая на входы A и C все возможные комбинации логических уровней, сделайте вывод о работе схемы в этом случае.

Сформулируйте принцип, по которому можно использовать дешифраторы на меньшую разрешающую способность, чем на ту которую он рассчитан.

Задание 3. Применение дешифраторов

Применение дешифраторов в цифровой технике весьма различно. Наиболее часто они используются как формирующие элементы, например, в схемах различных микроконтроллеров для формирования сигнала выбора определенной микросхемы. В этом случае на входы дешифратора подаются сигналы с шины адреса микропроцессора. При этом каждому участку адресного пространства ставится в соответствие определенное назначение (например, для

ОЗУ выделяется первых 1024 байт, для ПЗУ - следующие 2048 байт и так далее). Дешифратор, в таком случае, помогает сформировать сигнал управления (выбора микросхемы), так как согласно его таблице истинности на каждом выходе активный уровень формируется лишь однажды, при вполне определенной комбинации входных сигналов. Таким образом, не составляет особого труда сначала составить карту памяти (это подробная запись всего содержимого адресного пространства, с записью начального и конечного адреса каждого блока), а затем по ней выделить те разряды адреса, которые однозначно определяют обращение к тому или иному блоку адресного пространства. Именно эти разряды и будут являться входами для дешифраторов. Выход каждого дешифратора будет соединен с входом разрешения работы той микросхемы, которая в данном случае необходима.

В данном задании внимание в основном будет акцентировано на другом способе применения дешифраторов, который, в общем, является базовым и для всех других. Это использование дешифратора в совокупности с логическими элементами.

Помимо этого в основном данном задании будет введено понятие временной диаграммы, которая также очень часто используется в технике для иллюстрации работы цифровых устройств.

Временная диаграмма в общем случае представляет собой график, по оси абсцисс которой откладывается время в тактах, а по оси ординат необходимая цифровая величина (вход или выход цифровой схемы), принимающее значения «0» или «1». Необходимо отметить, что чаще всего осей ординат несколько, и они располагаются друг под другом. Собственно говоря, именно в этом и состоит основное преимущество применения временных диаграмм, работа цифрового устройства наглядно представляется во времени. Простейший пример временной диаграммы приведен на рис. 3.а.

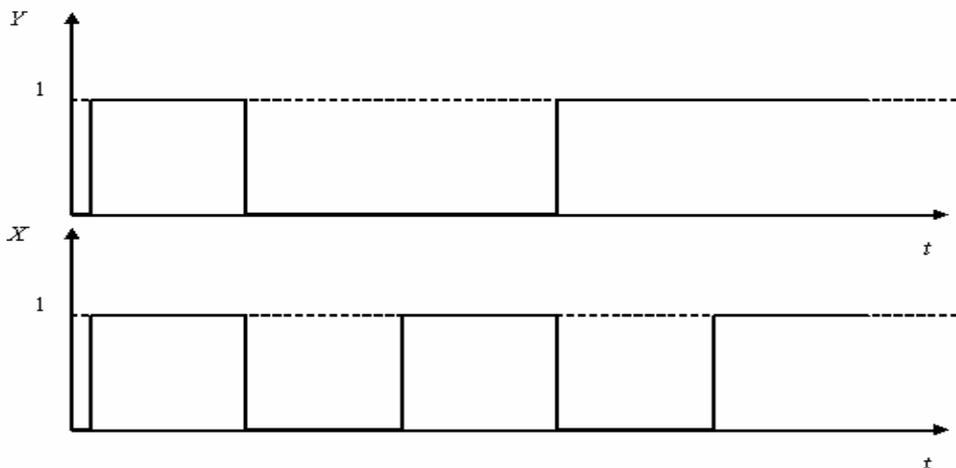


Рис. 3.а – Временная диаграмма

По таблице истинности логической функции можно без труда построить временную диаграмму. В *EWB* кроме инструментов автоматического построения таблицы истинности присутствуют инструменты и для построения

временных диаграмм. Это построение можно осуществить с помощью «Logic Analyzer». Общий вид окна свойств данного инструмента приведено на рис. 3.б (окно свойств вызывается двойным щелчком по изображению).

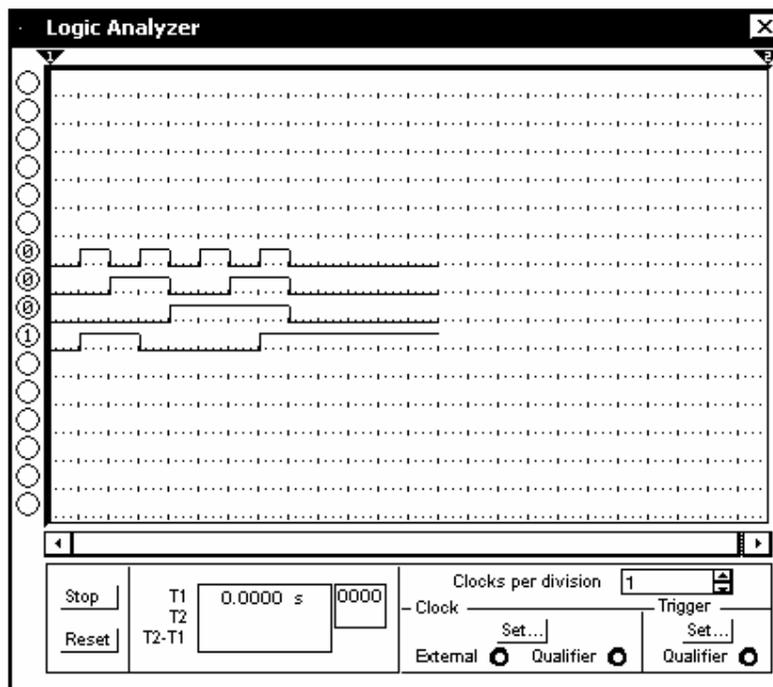


Рис. 3.б – Окно свойств Логического Анализатора

Построение временных диаграмм начинается автоматически при включении всей схемы. В меню «Clock per division» можно задавать масштаб просмотра по оси абсцисс (в данном случае времени). В области «Clock» кнопкой «Set» можно изменить установки внутреннего генератора времени (ось абсцисс), в частности задать частоту анализа («Internal clock rate»). В области «Trigger» можно изменить установки триггера (ось ординат).

Соберите схему изображенную на рис. 3.в.

Запрограммируйте генератор слов так, что бы на его выходе сформировать все возможные комбинации трехразрядного двоичного числа.

Установите ключ *G* в разрешающее положение. Подавая на вход дешифратора различные комбинации двоичного числа *C, B, A* (кнопкой «Step» в генераторе слов), постройте временную диаграмму работы дешифратора с логическими элементами на выходе. Оси диаграммы выберите согласно рис. 3.г.

Получите логическую функцию данного цифрового устройства. Для этого удобнее предварительно построить таблицу истинности, а по ней аналогично пунктам, изложенным выше построить саму логическую функцию.

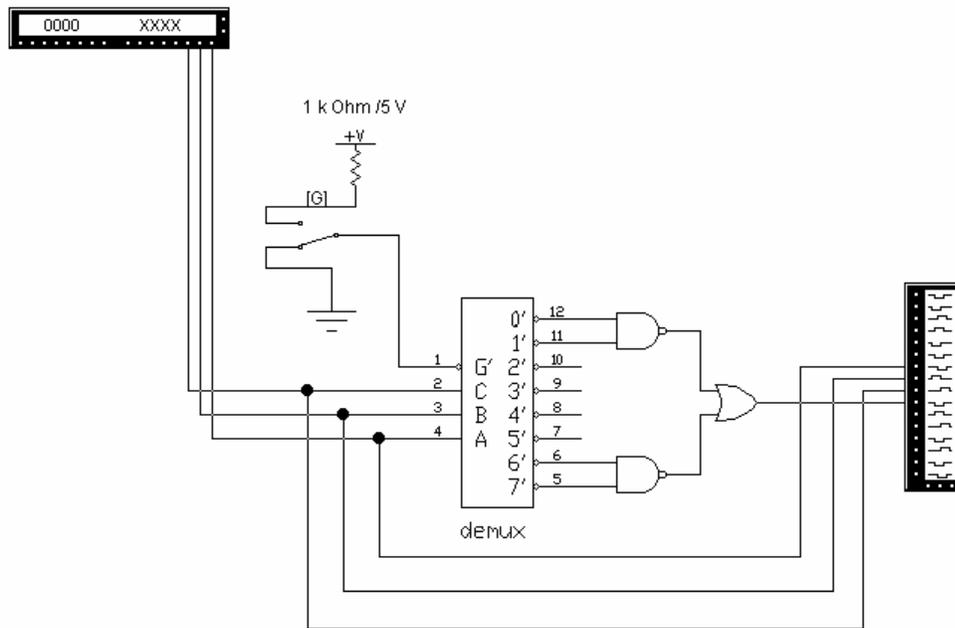


Рис. 3.в – Дешифратор с логическим элементами



Рис. 3.г– Временная диаграмма

Задание 4. Исследование работы мультиплексора

a) Исследование развернутой схемы мультиплексора

Создайте схему изображенную на рис. 4.а.

Здесь, на рис. 4.а приняты следующие условные обозначения: *G* – сигнал разрешения, *d0, d1* – информационные входы, *A* – адресный вход.

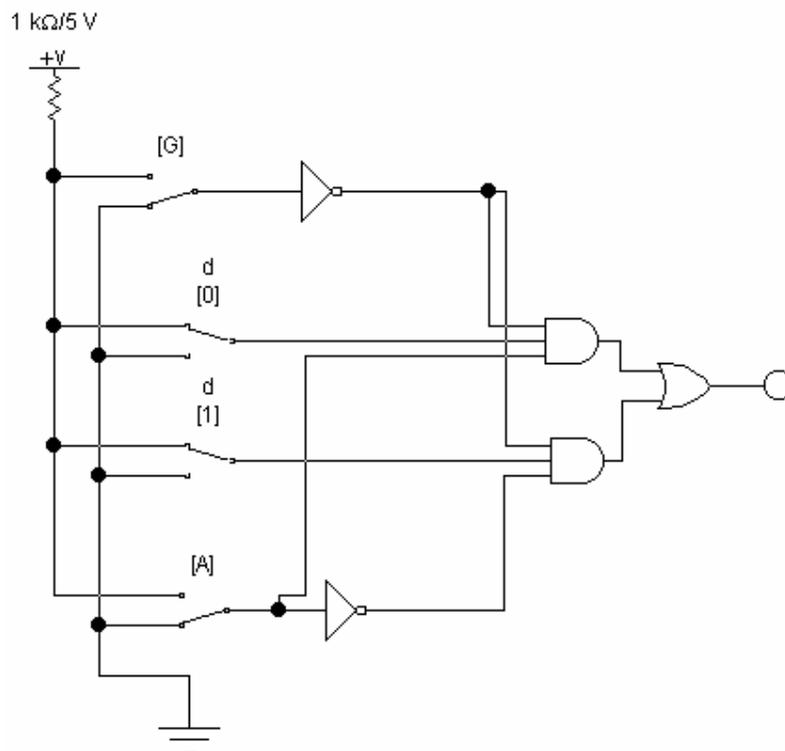


Рис. 4.а – Развернутая схема мультиплексора

Исследуйте поведение схемы мультиплексора, задавая различные сочетания логических уровней на входе схемы, заполнив при этом таблицу истинности, приведенную ниже (таблица 4.а).

Таблица 4.а – Развернутая схема мультиплексора

Входы		Адрес А	Выход Q
d0	d1		
0	0	0	
0	1		
1	0		
1	1		
0	0	1	
0	1		
1	0		
1	1		

б) *Исследование работы схемы мультиплексора 3*8 в основном режиме*

Соберите схему представленную на рис. 4.б.

Запрограммируйте генератор слов так, что бы на адресные входы мультиплексора (C, B, A) подавались все возможные комбинации логических уровней.

Подавая на адресный вход мультиплексора различные комбинации адреса C, B, A (кнопкой «Step» в генераторе слов), заполните таблицу истинности мультиплексора (таблица 4.б).

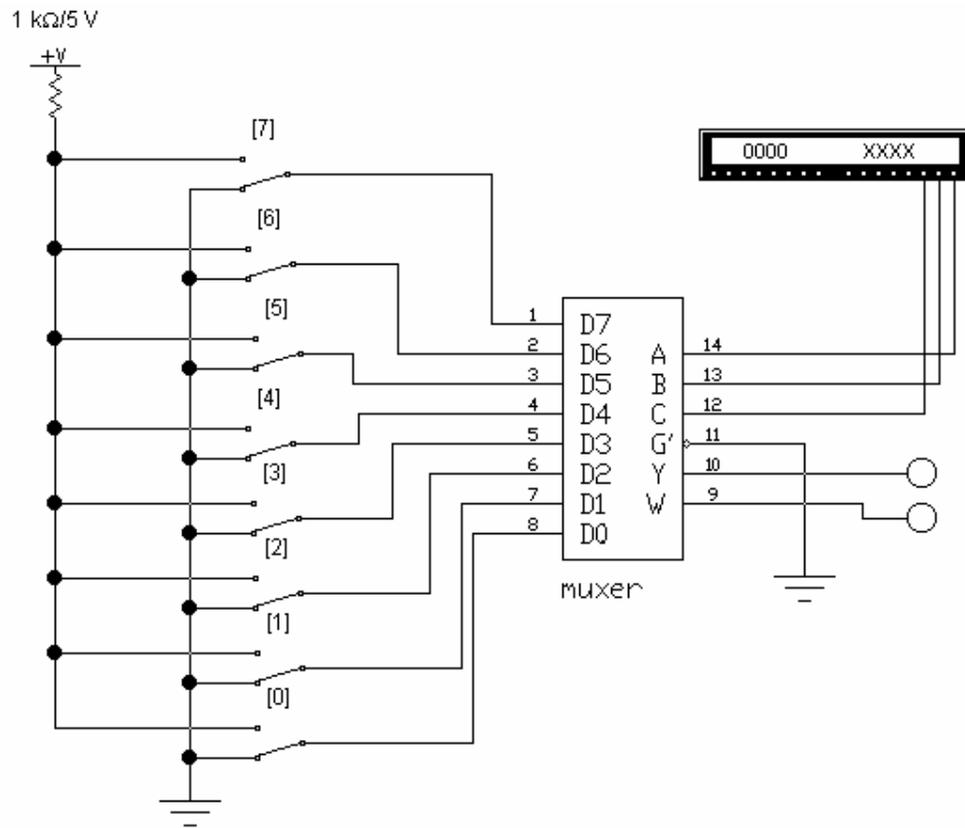


Рис. 4.6 – Схема мультиплексора 3*8

Таблица 4.6 – Таблица истинности мультиплексора

Входы								Выход				
d0	d1	d2	d3	d4	d5	d6	d7	C	B	A	Y	W
								0	0	0		
								0	0	1		
								0	1	0		
								0	1	1		
								1	0	0		
								1	0	1		
								1	1	0		
								1	1	1		

При этом комбинации соответствующих сигналов на входе $D0, D1, \dots, D7$ выбираются в соответствии с вариантом, заданным таблицей 4.в.

По таблице 4.б проследите правильность работы мультиплексора.

Задание 5. Реализация логической функции с помощью мультиплексора.

По заданной таблице истинности (таблица 5.а) реализуйте логическую функцию с помощью мультиплексора. Выбор значений выхода логической функции производится согласно вашему варианту. Реализация логической функ-

ции может быть осуществлена с помощью любого стандартного мультиплексора.

Таблица 4.в – Варианты для входов данных таблицы 4.б

Входы								Вариант
d0	d1	d2	d3	d4	d5	d6	d7	
0	1	0	1	0	1	0	1	1
0	0	1	1	0	0	1	1	2
1	1	1	0	0	0	1	1	3
1	1	1	1	0	0	0	0	4
1	0	0	0	1	0	0	1	5
0	0	0	0	1	1	1	1	6
1	1	0	0	0	0	1	1	7
0	0	0	0	0	1	1	1	8
1	0	1	1	0	1	0	1	9
1	0	0	0	1	0	1	0	10

Таблица 5.а – Варианты таблицы истинности

Значения логических переменных (для всех вариантов)			Варианты задания									
			1	2	3	4	5	6	7	8	9	10
A	B	D	Значения логической функции (для каждого варианта)									
0	0	0	0	0	0	0	1	0	1	0	1	1
0	0	1	1	0	1	1	1	0	0	0	1	1
0	1	0	1	0	1	1	0	0	0	1	0	1
0	1	1	0	0	1	1	0	0	0	1	0	1
1	0	0	1	1	0	1	0	1	1	1	1	1
1	0	1	1	1	0	0	1	1	0	1	1	1
1	1	0	1	0	1	1	0	1	0	0	1	0
1	1	1	0	0	1	1	1	1	0	0	1	1

Создайте схему с вашим вариантом реализации логической функции в EWB, занесите полученную схему в отчет.

С помощью Логического Анализатора постройте временные диаграммы работы вашей логической функции. По ним проверьте правильность функционирования схемы.

Контрольные вопросы

1. Дайте определение шифратору, дешифратору.
2. Чем отличается схема шифратора от схемы дешифратора?
3. Как в простейшем случае реализовать на дешифраторе демультиплексор? Постройте полученную схему в случае использования дешифратора 3*8.
4. Как влияет сигнал управления на работу логической схемы?

5. Как из двух дешифраторов 2×4 сделать один 3×8 ?
6. Как изменить расширенную схему дешифратора (рис. 2.а) так, что бы активным уровнем выходного сигнала данной схемы, была «1»?
7. Дайте определение мультиплексу.
8. Приведите примеры применения мультиплексоров.
9. Как на мультиплексе можно реализовать логическую функцию?
10. Любую ли логическую функцию можно реализовать на мультиплексе?
11. Дайте определение временной диаграммы.
12. Можно ли по произвольной временной диаграмме составить таблицу истинности?
13. Можно ли по таблице истинности составить временную диаграмму?
14. В каком виде записывается логическая функция мультиплекса?

ЛАБОРАТОРНАЯ РАБОТА № 3

Изучение работы триггеров

Цель работы:

1. Изучение структуры и исследование работы асинхронных и синхронных триггеров.
2. Исследование функций переходов и возбуждения основных типов триггеров.
3. Изучение взаимозаменяемости триггеров различных типов.

Приборы и элементы:

- Логические пробники (панель «Indicators/Red probe»)
- Источник напряжения + 5 В (панель «Basic/Pull-Up Resistor»)
- Земля (панель «Sources/Ground»)
- Двухвходовые элементы И, И-НЕ, ИЛИ, ИЛИ-НЕ (панель «Logic Gates/2-Input AND, NAND, OR, NOR Gates»)
- Двухпозиционные переключатели (панель «Basic/Switch»)
- Базовые триггеры RS, JK, D (панель «Digital/»)
- Схемы различных серий (панель «Digital/MUX,DEC»)

Краткие теоретические сведения

Последовательные цифровые устройства часто называют последовательными схемами, последовательными автоматами, дискретными автоматами с памятью, многотактными автоматами. Простейшим примером устройств данного типа являются триггеры.

Триггеры

Триггер – простейшая цифровая схема последовательного типа. У рассмотренных в предыдущих работах комбинационных схем состояние выхода Y в любой момент времени определяется только текущим состоянием входа X . В отличие от них, состояние выхода последовательной схемы (цифрового автомата) зависит еще и от внутреннего состояния схемы Q .

Таким образом, цифровой автомат является не только преобразователем, но и хранителем предшествующей и источником текущей информации

(состояния). Данное свойство обеспечивается наличием в схемах обратных связей.

Триггер имеет два устойчивых состояния: $Q=1$ и $Q=0$, поэтому его иногда называют бистабильной схемой. В каком из этих состояний окажется триггер, зависит от сигналов на информационном входе триггера и от его предыдущего состояния, то есть он имеет память. Можно сказать, что триггер является элементарной ячейкой памяти.

Тип триггера определяется алгоритмом его работы. В зависимости от алгоритма триггер может иметь установочные, информационные и управляющие входы. Установочные входы служат для перевода триггера в одно из определенных состояний, независимо от состояния других входов. Входы управления разрешают запись данных, подающихся на информационные входы. Наиболее распространенными являются триггеры RS , JK , D и T типов.

1. Триггер типа RS

RS-триггер – простейший автомат с памятью, который может находиться в двух состояниях. Триггер имеет два информационных входа: установки S (*set* – установка) и сброса R (*reset* – сброс), на которые подаются входные сигналы от внешних источников. При подаче на вход установки активного логического уровня триггер устанавливается в «1» ($Q=1, \bar{Q}=0$), при подаче активного уровня на вход сброса триггер устанавливается в «0» ($Q=0, \bar{Q}=1$). Если подать на оба входа установки (возбуждения) пассивный уровень, то триггер будет сохранять предыдущее состояние выходов: $Q=0$ ($\bar{Q}=1$) либо $Q=1$ ($\bar{Q}=0$). Каждое такое состояние устойчиво и поддерживается за счет действия обратных связей.

Для триггеров этого типа является недопустимой одновременная подача активного уровня на оба входа установки и сброса, т. к. триггер по определению не может одновременно быть установлен в ноль и единицу. На практике, подача активного уровня на установочные входы приводит к тому, что это состояние не может быть сохранено и невозможно определить, в каком состоянии будет находиться триггер при последующей подаче на установочные входы сигналов пассивного уровня.

На рис. 1 и 2 показаны два вида *RS-триггера*, выполненных на элементах «ИЛИ-НЕ» и «И-НЕ».



Рис. 1 – *RS-триггер* на элементах «ИЛИ-НЕ»



Рис. 2 – RS-триггер на элементах «И-НЕ»

Для схемы на рис. 1 активным уровнем является уровень логической единицы, для схемы на рис. 2 – уровень логического нуля. Схема на рис. 2 получила название *RS-триггера* с инверсными входами или просто – \overline{RS} -триггер.

RS-триггер является основным узлом для построения последовательных схем. Напомним, что название схем такого типа «последовательные» означает, то, что состояние выхода зависит от того, в какой последовательности на входы, подаются сигналы, и каково было предшествующее внутреннее состояние. Так, если в *RS-триггере* (рис. 1) вначале установить комбинацию $R=0, S=1$, а потом перейти к $R=0, S=0$, то состояние выхода будет $Q=1$. Если же вначале установить комбинацию $R=1, S=0$, а потом перейти к $R=0, S=0$, то состояние выхода будет другим – $Q=0$, несмотря на одинаковые комбинации сигналов на входах. Таким образом, при одном и том же входном наборе $R=0, S=0$ выход триггера может находиться в разных состояниях.

Условия переходов триггеров из одного состояния в другое (алгоритм работы) можно описать табличным, аналитическим или графическим способами. Табличное описание работы RS-триггера (рис. 1) представлено в таблице 1 (таблица переходов и таблица функций возбуждения).

Таблица 1 – Таблица переходов и функций возбуждения RS-триггера

Таблица переходов			Таблица функций возбуждений			
R	S	Q_{t+1}	Q_t	Q_{t+1}	R	S
0	0	Q_t	0	0	-	0
0	1	1	0	1	0	1
1	0	0	1	0	1	0
1	1	x	1	1	0	-

В таблице использованы следующие обозначения: « Q_t » – предшествующее состояние выхода; « Q_{t+1} » – новое состояние, устанавливающееся после перехода (возможно $Q_{t+1} = Q_t$); «-» – безразличное значение сигнала: «0» или «1»; «x» – запрещенное состояние.

Аналитическое описание (характеристическое уравнение) можно получить из таблицы 1 по правилам алгебры логики.

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

Зависимость Q_{t+1} от Q_t характеризует свойство запоминания предшествующего состояния.

Таблица 1 показывает, что схема, которая находилась в состоянии $Q=0$, сохраняет это состояние как при воздействии входного набора $R=0, S=0$, так и при воздействии $R=1, S=0$. Если же на вход схемы, находящейся в состоянии $Q=0$, подействовать набором $R=0, S=1$, то она переходит в состояние $Q=1$ и сохраняет его при входных наборах $R=0, S=1$, либо $R=0, S=0$.

Таблицы переходов и функций возбуждения легче запомнить, держа в уме следующее: вход S – вход установки, то есть при подаче на него логической единицы триггер должен перейти в активное состояние; установить триггер в ноль можно только подав, сигнал сброса R ; запрещенным является сигнал одновременного сброса и установки $R=1, S=1$; сигнал $R=0, S=0$ – сохраняет предыдущее состояние.

Схема триггера позволяет запоминать состояние логической схемы, но так как в начальный момент времени может возникать переходный процесс (в цифровых схемах этот процесс называется «опасные гонки»), то запоминать состояния логической схемы нужно только в определённые моменты времени, когда все переходные процессы закончены. Таким образом цифровые схемы требуют синхросигнала. Схемы с сигналом управления (синхронизации) представлены ниже.

2. JK-триггер

Триггер JK-типа имеет более сложную по сравнению с *RS-триггером* структуру и более широкие функциональные возможности. Помимо информационных входов J и K и прямого и инверсного выходов Q и \bar{Q} , *JK-триггер* имеет вход управления C (этот вход также называют тактирующим или счетным), а также асинхронные установочные R и S входы. Обычно активными уровнями установочных сигналов являются нули, как в схеме \overline{RS} -триггера изображенной на рис. 2. Установочные входы имеют приоритет над остальными. Активный уровень сигнала на входе S устанавливает триггер в состояние $Q=1$, а активный уровень сигнала на входе R – в состояние $Q=0$, независимо от сигналов на остальных входах.

Если на входы установки одновременно подать пассивный уровень сигнала, то состояние триггера будет изменяться по фронту импульса на счетном входе в зависимости от состояния входов J и K , как показано в таблице 2.

Таблица 2 – Таблица переходов и возбуждения *JK-триггера*

Таблица переходов			Таблица функций возбуждений			
J	K	Q_{t+1}	Q_t	Q_{t+1}	K	J
0	0	Q_t	0	0	-	0
0	1	0	0	1	0	1
1	0	1	1	0	1	0
1	1	\bar{Q}_t	1	1	0	-

Один из вариантов функциональной схемы *JK-триггера* с входами установки логическим нулем и его условное графическое обозначение приведены на рис. 3. Временные диаграммы его работы при $R=S=1$ приведены на рис. 4.

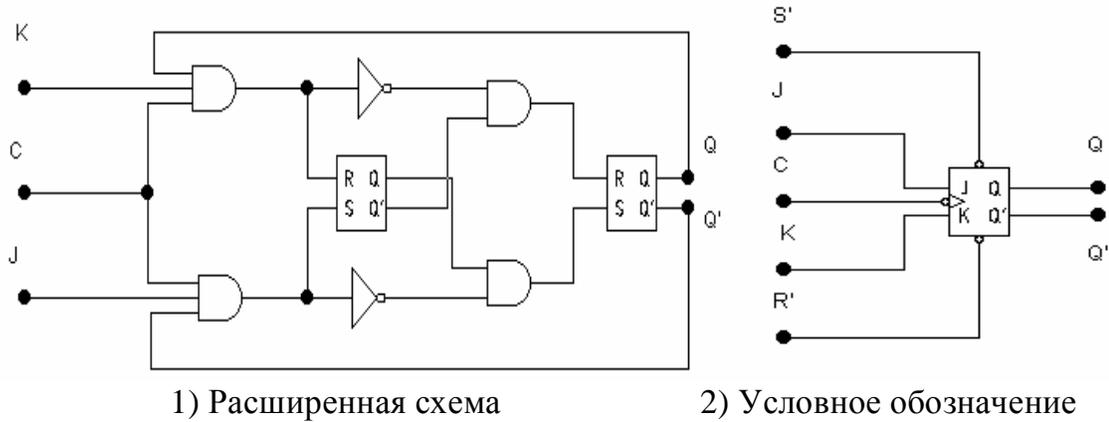


Рис. 3 – Схема *JK-триггера*

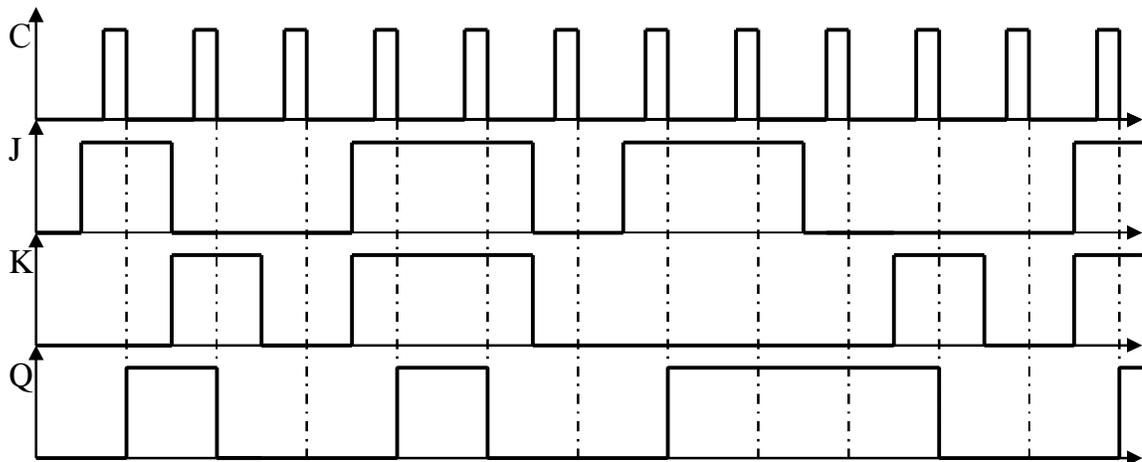


Рис. 4 – Временная диаграмма *JK-триггера*

В данном случае все изменения выхода происходят только в момент отрицательного перепада тактового сигнала.

3. D-триггер

D-триггер имеет один информационный вход *D* (*data* – данные). Информация с входа *D* заносится в триггер по положительному перепаду импульса на входе синхронизации *C* и сохраняется до следующего положительного перепада. Помимо входов *C* и *D* триггер снабжен асинхронными установочными *R* и *S* входами. Установочные входы имеют наивысший приоритет. Они управляют триггером независимо от сигналов на входах *C* и *D*. Функционирование *D-триггера* описывается таблицей 3 переходов и возбуждения и диаграммами рис. 5.

Таблица 3 – Таблица переходов и возбуждения *D-триггера*

Таблица переходов		Таблица функций возбуждений		
D	Q_{t+1}	Q_t	Q_{t+1}	D
0	0	0	0	0
		0	1	1
1	1	1	0	0
		1	1	1

Характеристическое уравнение D-триггера:

$$Q_{t+1} = D_t \quad (2)$$

Уравнение триггера показывает, что состояние триггера на $t+1$ такте равно входному сигналу в момент, предшествующий тактовому перепаду сигнала C . Условное обозначение *D-триггера* представлено на рис 6.1. Функциональная схема *D-триггера* может быть получена из схемы *JK-триггера* путем подключения входа D к входу J через инвертор, рис. 6.2.

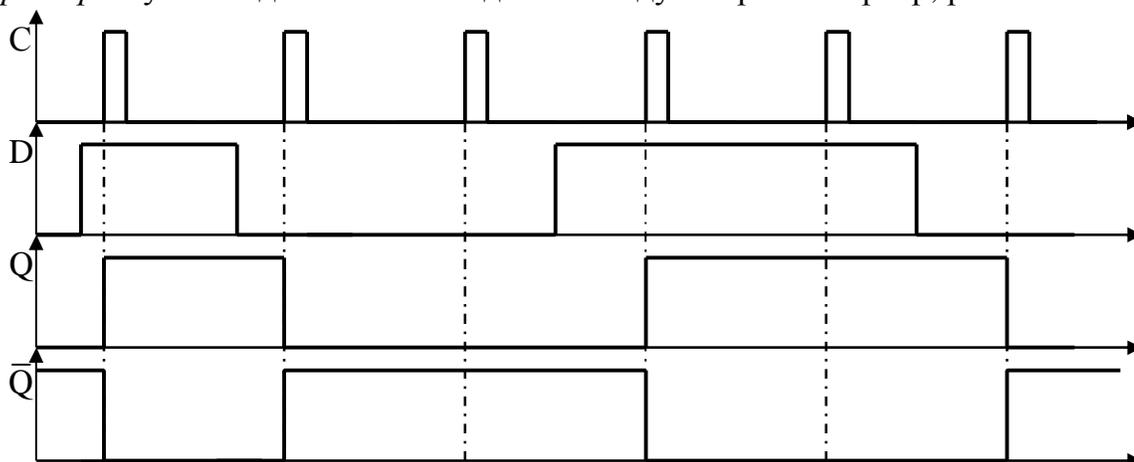


Рис. 5 – Временная диаграмма *D-триггера*

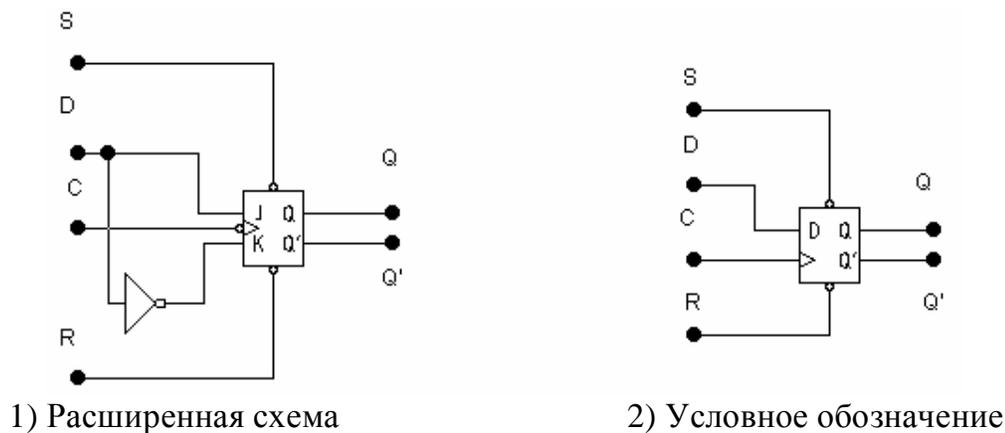


Рис. 6 – Схема *D-триггера*

4. T-триггер (счетный триггер)

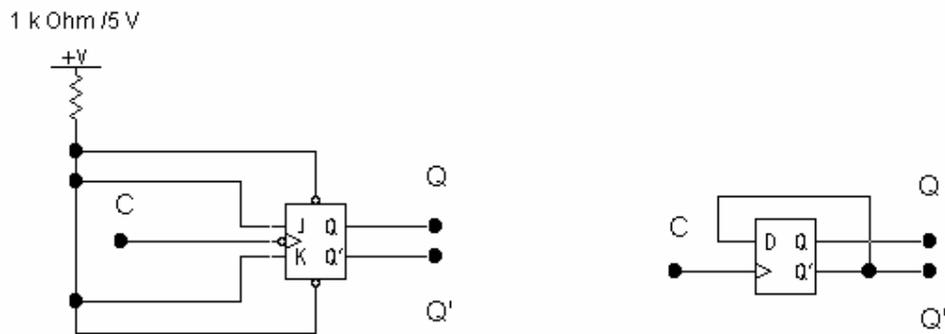
На основе *JK-триггеров* и *D-триггеров* можно построить схемы, осуществляющие так называемый счетный режим. Такие схемы называют *T-*

триггерами или счетными триггерами, что связано со способом их функционирования. На рис. 7 представлены схемы организации *T*-триггера на основе *JK*-триггера (рис. 7.1) и *D*-триггера (рис. 7.2). Счетный режим иллюстрируется временными диаграммами рис. 8.

В *JK*-триггере с входами установки логическим нулем счетный режим реализуется путем подачи констант $J=K=1$ и $R'=S'=1$ и сигнала *T* на вход *C*.

В соответствии с таблицей функционирования при каждом отрицательном перепаде входного сигнала *T* состояние триггера изменяет свое значение на противоположное.

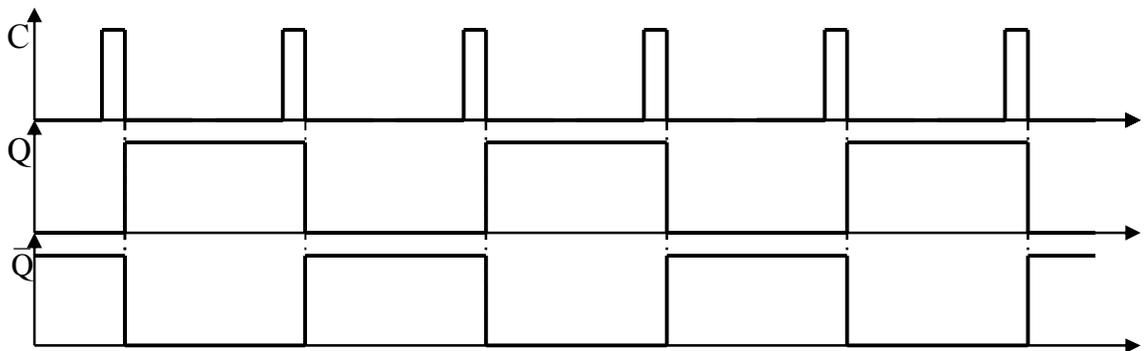
В *D*-триггере счетный режим реализуется при помощи обратной связи (на вход *D* подается сигнал с инверсного выхода). Таким образом, всегда существует неравенство сигнала на входе *D* и сигнала на выходе *Q*: если $Q=1$, $D=0$.



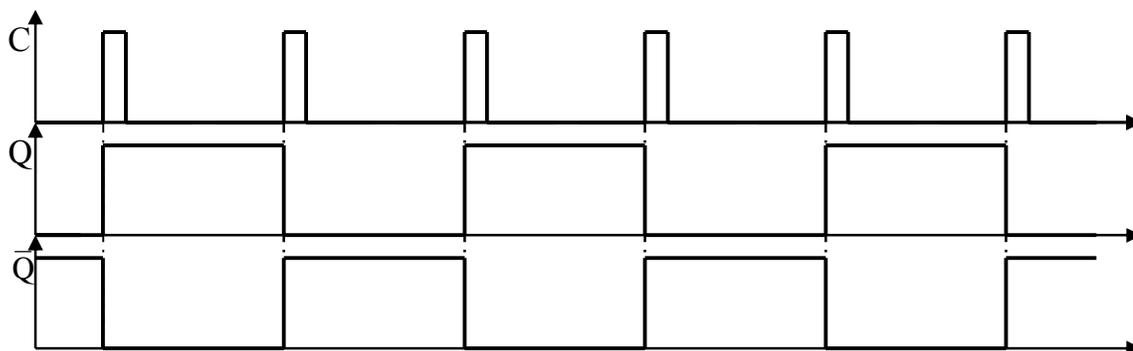
1) На базе *JK*-триггера

2) На базе *D*-триггера

Рис. 7 – Схема *T*-триггера



1) на базе *JK*-триггера



2) на базе *D-триггера*

Рис. 8 – Временная диаграмма *T-триггера*

Следовательно, при каждом положительном перепаде сигнала на счетном входе *C* в соответствии с принципом действия *D-триггера* состояние выхода будет изменяться на противоположное.

Таким образом, на каждые два входных тактовых импульса *T-триггер* формирует один период выходного сигнала *Q*. Следовательно, триггер осуществляет деление частоты f_T на его входе на 2:

$$f_Q = \frac{f_T}{2} \quad (3)$$

Порядок работы

Задание 1. Исследование работы схемы *RS* триггера

а) Исследование *RS* триггера, составленного из двух элементов «ИЛИ-НЕ»
Создайте схему изображенную на рис. 1.а.

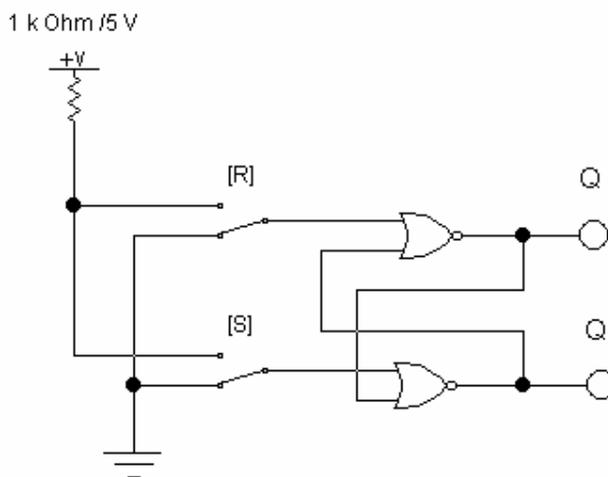


Рис. 1.а – Схема *RS* триггера на базе элементов «ИЛИ-НЕ»

Включите схему. Убедитесь в правильности работы триггера (перехода из одного состояния в другое), проверив следующие утверждения:

при $S=0, R=1$ триггер сбрасывается в состояние $Q=0$;

при $S=0, R=0$ триггер сохраняет свое прежнее состояние $Q=0$;

при $S=1, R=0$ триггер устанавливается в состояние $Q=1$;
 при $S=0, R=0$ триггер сохраняет прежнее состояние $Q=1$;
 при $S=1, R=1$ триггер находится в запрещенном состоянии.

Заполните таблицу возбуждения данного типа триггера (таблица 1.а).

Таблица 1.а – RS триггер на базе элементов «ИЛИ-НЕ»

Q_t	Q_{t+1}	R	S
0		0	0
0		1	0
0		0	1
1		0	0
1		1	0
1		0	1

Примечание: Таблица возбуждения триггера составляется для различных комбинаций R и S входов. При этом предварительно необходимо добиться начального состояния триггера Q_t , помня, что установить триггер в состояние $Q_t=0$ можно подав сигнал $R=1$ ($S=0$), а установить триггер в состояние $Q_t=1$ можно подав сигнал $S=1$ ($R=0$).

б) Исследование \overline{RS} -триггера, составленного из двух элементов «И-НЕ»
 Создайте схему изображенной на рис. 1.б.

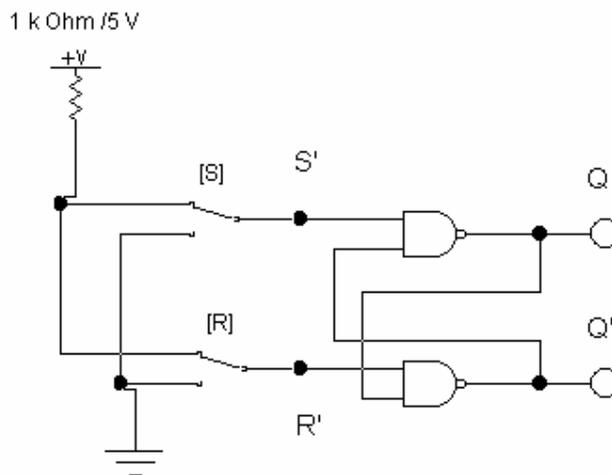


Рис. 1.б – Схема \overline{RS} -триггера на базе элементов «И-НЕ»

Примечание: на схеме инверсные входы \overline{RS} -триггера \overline{R} и \overline{S} обозначаются как R' и S' соответственно. Управляющие клавиши ключей (напомним, что управляющая клавиша задается в закладке «Key» окна свойств ключа, и предназначена для изменения положения ключа) заданны соответственно «R» для входа \overline{R} и «S» для входа \overline{S} .

Включите схему. Убедитесь в правильности работы триггера (перехода из одного состояния в другое), проверив следующие утверждения:

при $\overline{S}=1, \overline{R}=0$ триггер сбрасывается в состояние $Q=0$;

при $\bar{S} = 1, \bar{R} = 1$ триггер сохраняет свое прежнее состояние $Q=0$;
 при $\bar{S} = 0, \bar{R} = 1$ триггер устанавливается в состояние $Q=1$;
 при $\bar{S} = 1, \bar{R} = 0$ триггер сохраняет свое прежнее состояние $Q=1$;
 при $\bar{S} = 0, \bar{R} = 0$ триггер находится в запрещенном состоянии.

Заполните таблицу возбуждения данного типа триггера (таблица 1.б).

Таблица 1.б – $\bar{R}\bar{S}$ -триггер на базе элементов «И-НЕ»

Q_t	Q_{t+1}	$\bar{R} = R'$	$\bar{S} = S'$
0		1	1
0		1	0
0		0	1
1		1	0
1		1	0
1		0	1

Примечание: Таблица возбуждения триггера составляется для различных комбинаций \bar{R} и \bar{S} входов. При этом предварительно необходимо добиться начального состояния триггера Q_t , помня, что установить триггер в состояние $Q_t=1$ можно подав сигнал $\bar{S} = 0$ ($\bar{R} = 1$), а установить триггер в состояние $Q_t=0$ можно подав сигнал $\bar{R} = 0$ ($\bar{S} = 1$).

Задание 2. Исследование работы *JK-триггера*

а) Составление функции возбуждения *JK-триггера*

Соберите схему, изображенную на рис. 2.а.

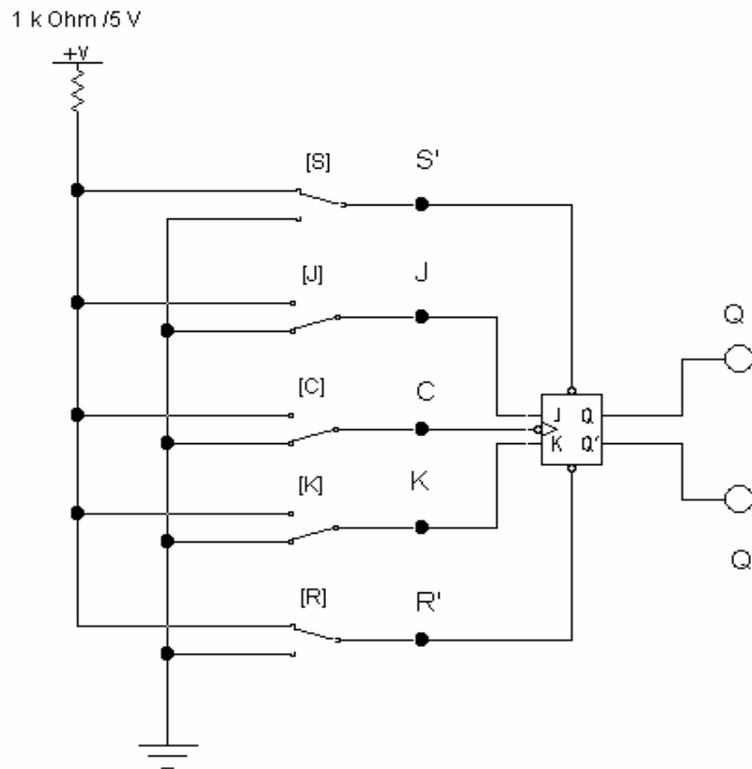


Рис. 2.а – Схема *JK-триггера*

Отметим, что на схеме используется упрощенная микросхема *JK-триггера*, в которой установочные входы \bar{R} и \bar{S} инверсные (на микросхеме такие входы имеют кружок на линии входящей в корпус).

Включите схему. Убедитесь в правильности работы установочных входов триггера, проверив следующие утверждения:

при $\bar{S} = 1, \bar{R} = 0$ триггер сбрасывается в состояние $Q=0$, независимо от входов J, K, C ;

при $\bar{S} = 0, \bar{R} = 1$ триггер устанавливается в состояние $Q=1$, независимо от входов J, K, C .

Таким образом, для установки триггера в требуемое исходное состояние необходимо подать на его вход одну из перечисленных комбинаций, затем установить $\bar{S} = 1, \bar{R} = 1$.

Установив триггер в необходимое исходное состояние, заполните таблицу возбуждения *JK-триггера* (таблица 2.а).

Таблица 2.а – *JK-триггер*

Q_t	Q_{t+1}	J	K
0		0	0
0		0	1
0		1	0
0		1	1
1		0	0
1		0	1
1		1	0
1		1	1

Примечание: переход триггера в новое состояние происходит по заднему фронту импульса C . Таким образом, для получения нового состояния Q_{t+1} необходимо установить начальное состояние триггера, затем необходимую комбинацию входов J и K , и произвести переключение ключа C .

б) Составление временных диаграмм JK-триггера

По таблице возбуждения *JK-триггера* (таблица 2.а) составьте временную диаграмму работы триггера. Оси графика необходимо выбрать аналогично рис. 4 (краткой теории). При этом на оси « C » также откладываются последовательно импульсы синхронизации. Значения, откладываемые на других осях временной диаграммы необходимо выбирать построочно из таблицы 2.а, по следующим правилам:

до появления импульса C значения J и K должны устанавливаться согласно каждой выбранной строке таблице возбуждения;

значение по оси Q выбирается равным Q_t (согласно выбранной строке) до каждого момента исчезновения импульса C (до появления так же называемого – заднего фронта);

после исчезновения импульса C значение оси Q выбирается равным Q_{t+1} (согласно выбранной строке) до момента установки новых значений J и K (перехода к анализу новой строки таблицы возбуждения);

переход к новым значениям J и K производится после исчезновения исследуемого импульса C , но до появления нового (и далее по каждой строке аналогично).

При этом полученные временные диаграммы будут отличными от приведенных на рис. 4, так как в соответствии с таблицей 2.а каждый раз триггер необходимо сбрасывать в состояние Q_i , до анализа исследуемой строки.

Задание 3. Исследование работы D -триггера

а) Исследование D триггера составленного на базе JK -триггера

Создайте схему, изображенную на рис. 3.а.

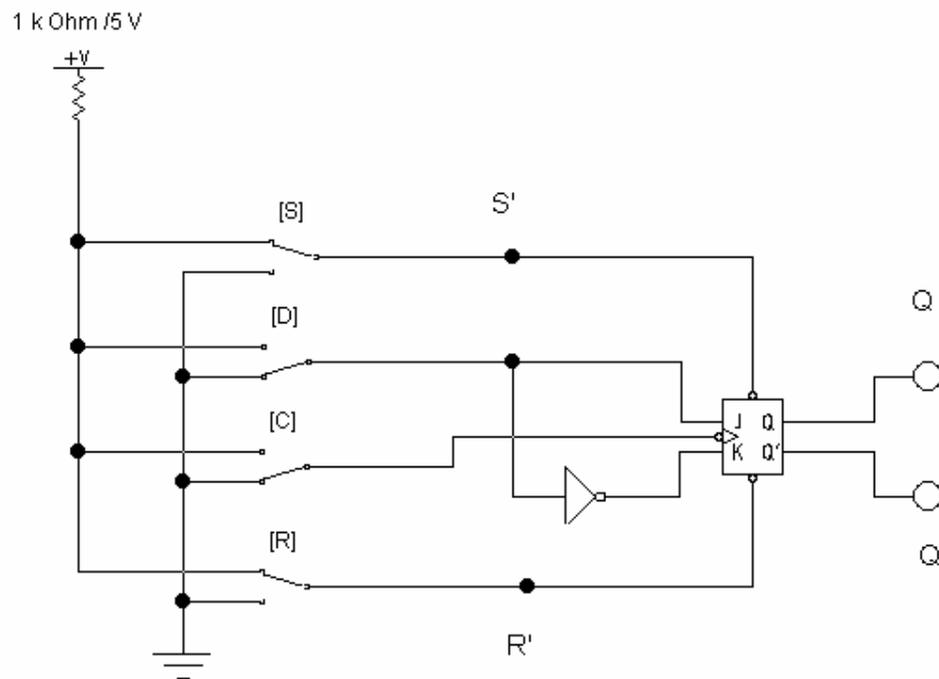


Рис. 3.а – Схема D -триггера на базе JK -триггера

Включите схему. Убедитесь, в правильности функционирования установочных входов:

при $\bar{S} = 1$, $\bar{R} = 0$ триггер сбрасывается в состояние $Q=0$, независимо от входов D , C ;

при $\bar{S} = 0$, $\bar{R} = 1$ триггер устанавливается в состояние $Q=1$, независимо от входов D , C .

Заполните таблицу переходов (таблица 3.а), переведя предварительно установочные входы в разрешающее состояние $\bar{S} = 1$ и $\bar{R} = 1$.

Таблица 3.а – Переходы *D-триггера*

D	Q_{t+1}
0	
1	

б) *Получение функций возбуждения D триггера*

Создайте схему изображенную на рис. 3.б.

Проверите правильность работы установочных входов \bar{R} и \bar{S} . Убедитесь в аналогичности схем 3.а и 3.б с помощью полученной в предыдущем пункте таблицы переходов.

Заполните таблицу возбуждения *D-триггера* (таблица 3.б), предварительно устанавливая необходимое состояние Q_t сигналами \bar{R} и \bar{S} (после установки необходимо перевести данные входы в разрешающие положение $\bar{R} = 1, \bar{S} = 1$).

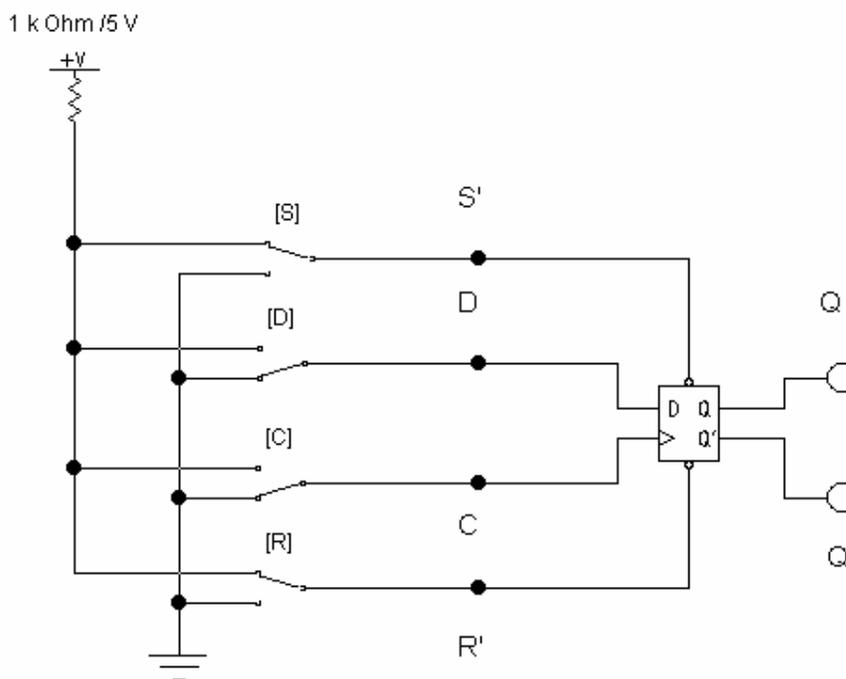


Рис. 3.б – Схема *D-триггера*

Таблица 3.б – Функции возбуждения *D-триггера*

Q_t	D	Q_{t+1}
0	0	
0	1	
1	0	
1	1	

Примечание: необходимо помнить, что переключение триггера в новое состояние происходит по переднему фронту импульса C .

в) Составление временной диаграммы D триггера

По таблице возбуждения D -триггера (таблица 3.б) составьте временную диаграмму работы триггера. Оси графика необходимо выбрать аналогично рис. 5 (краткой теории). При этом на оси « C » так же откладываются последовательно импульсы разрешения. Значения, откладываемые на других осях временной диаграммы необходимо выбирать из таблицы 3.б, аналогично пункту 2.б.

Полученные временные диаграммы будут отличны от диаграмм рис. 5, так как в соответствии с таблицей 3.б каждый раз триггер необходимо сбрасывать в значение Q_t .

Задание 4. Исследование работы T триггера

а) Исследование T -триггера на основе JK -триггера

Соберите T -триггер на основе JK -триггера согласно схеме, изображенной на рис. 4.а.

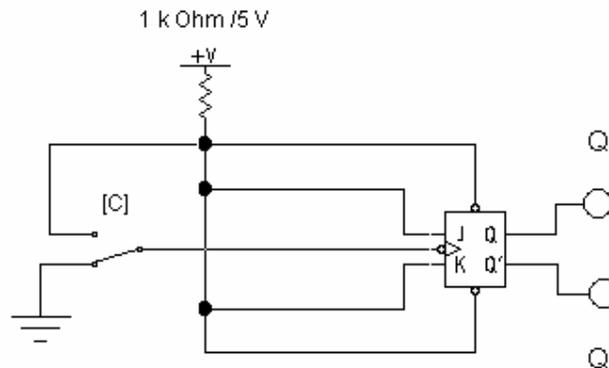


Рис. 4.а – Схема T -триггера на основе JK -триггера

Включите схему. Изменяя состояние входа C соответствующим ключом, постройте временную диаграмму T -триггера. Сравните полученный результат с рис. 7.1.

б) Исследование T -триггера на основе D -триггера

Соберите T -триггер на основе D -триггера согласно схеме, изображенной на рис. 4.б.

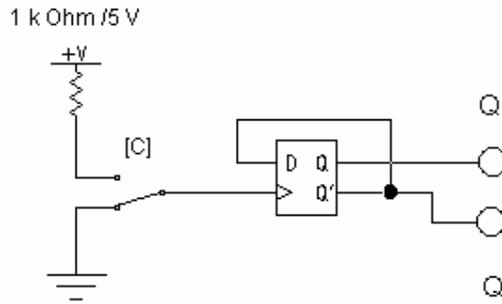


Рис. 4.б – Схема *T-триггера* на основе *D-триггера*

Включите схему. Изменяя состояние входа *C* соответствующим ключом, постройте временную диаграмму *T-триггера*. Сравните полученный результат с рис. 7.2.

Контрольные вопросы

1. Дайте определение триггера.
2. Чем отличается *RS-триггер* с обычными входами от *RS-триггера* с инверсными входами?
3. Что называется запрещенным и неопределенным состоянием триггера?
4. Приведите примеры запрещенного и неопределенного состояния различных триггерных схем.
5. Чем отличается таблица переходов от таблицы возбуждения?
6. Приведите характеристическое уравнение *RS-триггера*.
7. Чем отличается *JK-триггер* от *RS-триггера*?
8. Приведите характеристическое уравнение *JK-триггера*.
9. Зачем нужны установочные входы в триггерах?
10. Каков приоритет установочных входов в триггерах по сравнению с информационными входами?
11. Охарактеризуйте основное применение *D-триггера*?
12. Приведите характеристическое уравнение *D-триггера*.
13. Чем отличается *D-триггер* от *T-триггера*?
14. Чему равна частота следования импульсов в *T-триггере*?

ЛАБОРАТОРНАЯ РАБОТА № 4

Изучение сумматоров, полусумматоров, регистров и счетчиков

Цель работы:

1. Исследование сумматоров и полусумматоров.
2. Изучение структуры и исследование работы суммирующих и вычитающих счетчиков, счетчиков с измененным коэффициентом пересчета.
3. Изучение регистров.

Приборы и элементы:

Генератор слов (панель «Instruments/Word Generator»)
Логические пробники (панель «Indicators/Red probe»)
Источник напряжения + 5 В (панель «Basic/Pull-Up Resistor»)
Земля (панель «Sources/Ground»)
Двухпозиционные переключатели (панель «Basic/Switch»)
Двухвходовые элементы И, И-НЕ, ИЛИ, ИЛИ-НЕ (панель «Logic Gates/2-Input AND, NAND, OR, NOR Gates»)
Сумматор (панель «Digital/Half-Adder»)
D-триггер (панель «Digital/»)
Декодер (панель «Indicators/Decoded 7 segment display»)

Краткие теоретические сведения

Сумматоры и полусумматоры

Широкое применение в цифровой технике находят элементы, выполняющие различные арифметические действия. Операция суммирования – базовая арифметическая операция в двоичной алгебре. Поэтому для дальнейшего изучения цифровой техники необходимо исследовать способы получения сумматоров.

1. Сумматоры по модулю два

Построение двоичных сумматоров обычно начинается с сумматора по модулю 2. Ниже представлена таблица истинности этого сумматора.

Таблица 1 – Таблица истинности сумматора по модулю два

Входы		Выход
X	Y	Out
0	0	0
0	1	1
1	0	1
1	1	0

Из таблицы 1 видно, что логическая функция, выражающая принцип работы сумматора по модулю два, имеет вид:

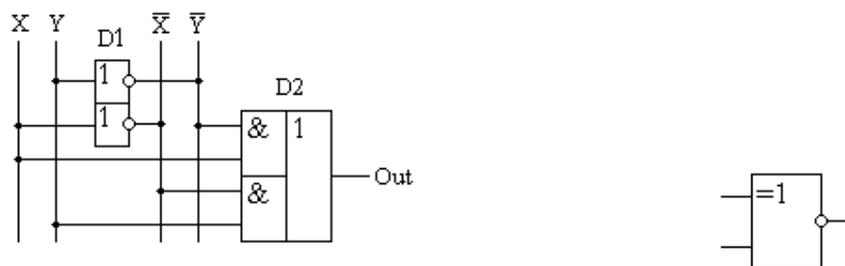
$$F = X \oplus Y = X\bar{Y} \vee \bar{X}Y \quad (1)$$

и представляет собой описанную ранее (формула 7, Лабораторная работа №1) функцию «исключающего ИЛИ».

На рис. 1.1 представлена схемная реализация сумматора по модулю два, составленная по таблице истинности 1. На рис. 1.2 приведено условное обозначение этой же схемы в виде одного элемента – «Исключающего ИЛИ».

Однако представленная схема имеет недостаток, который можно увидеть в таблице 1 – не учет переноса при сложении $X = 1$ и $Y = 1$.

Сумматор по модулю 2 выполняет суммирование без учёта переноса. В обычном двоичном сумматоре требуется учитывать перенос, поэтому требуются схемы, позволяющие формировать перенос в следующий двоичный разряд.



1) Расширенная схема

2) «Исключающее ИЛИ»

Рис. 1 – Сумматор по модулю два

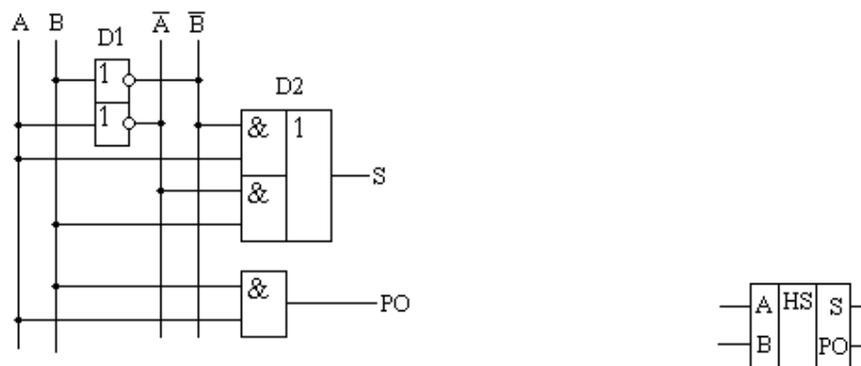
2. Полусумматоры

Таблица истинности полусумматора приведена ниже.

Таблица 2 – Таблица истинности полусумматора

Входы		Выходы	
A	B	S	PO
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

На рис. 2 представлена соответствующая схемная реализация полусумматора на базе логических элементов (рис. 2.1) и в виде одного устройства (рис. 2.2).



1) Расширенная схема

2) Условное обозначение

Рис. 2 – Схема полусумматора

Схема полусумматора формирует перенос в следующий разряд (*PO*), но не может учитывать перенос из предыдущего разряда, поэтому она и называется полусумматором. Для реализации же полного суммирования (пусть пока и одноразрядного) необходимо помимо формирования переноса в следующий разряд учитывать еще и перенос из предыдущего разряда (это нужно для формирования многоразрядных сумматоров).

3. Одноразрядные сумматоры

Таблица истинности полного двоичного одноразрядного сумматора приведена в таблице 3.

Таблица 3 – Таблица истинности одноразрядного сумматора

Входы			Выходы	
PI	A	B	S	PO
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Здесь, помимо формирования переноса в следующий разряд (*PO*) учитывается еще и перенос из предыдущего разряда (*PI*).

На рис. 3 представлена соответствующая схемная реализация сумматора на базе логических элементов (рис. 3.1) и в виде одного устройства (рис. 3.2).

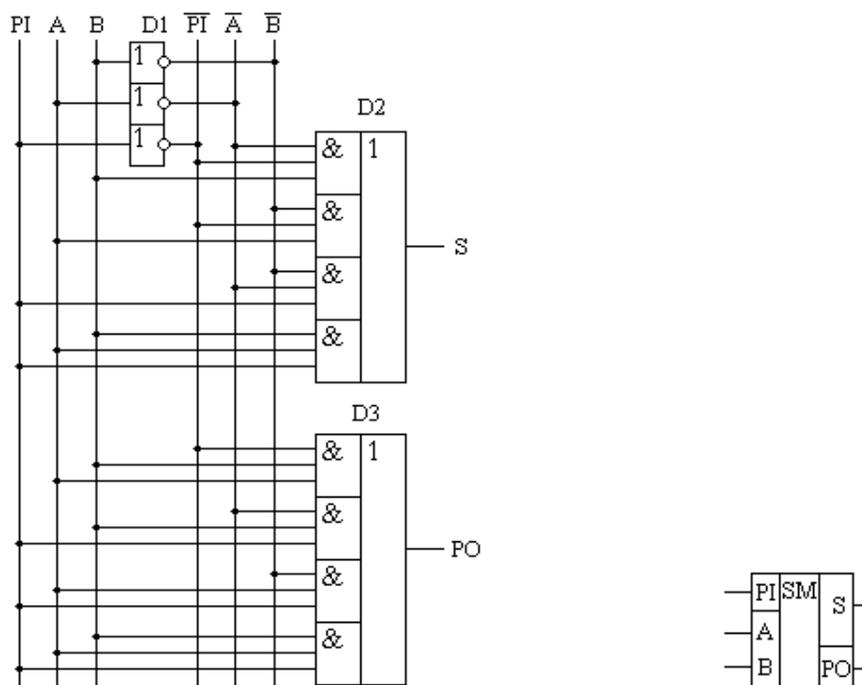
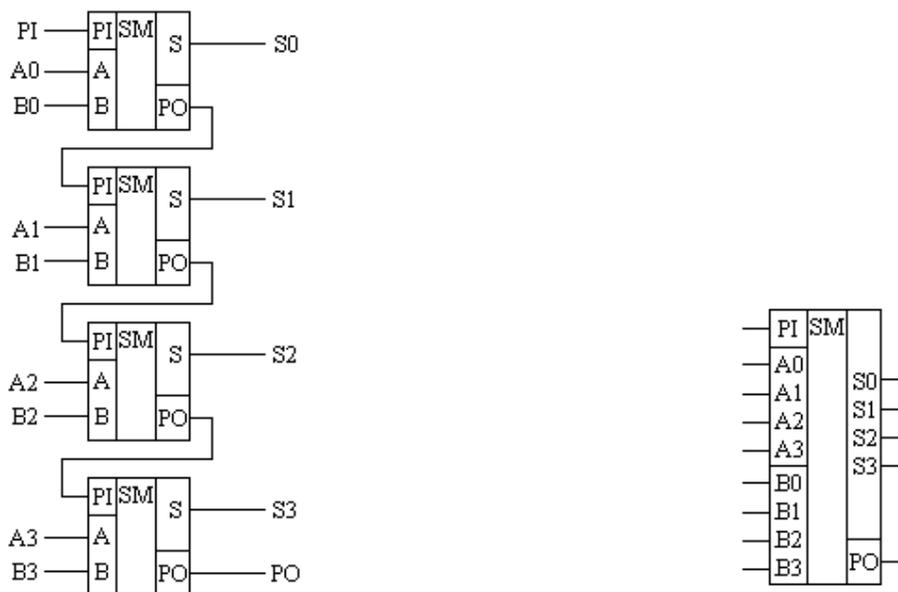


Рис. 3 – Схема двоичного одноразрядного сумматора

4. Многоразрядные сумматоры

Для того чтобы получить многоразрядный сумматор, необходимо соединить входы и выходы переносов соответствующих двоичных разрядов. Схема соединения для трехразрядного сумматора приведена на рис. 4.1. Здесь же приведено условное обозначение данного сумматора, применяемого на схемах (рис. 4.2).



1) Расширенная схема

2) Условное обозначение

Рис. 4 – Схема полного двоичного трехразрядного сумматора

В схеме на рис. 4.1 рассмотрены принципы работы двоичного последовательного сумматора. Данной схеме присущ один недостаток – невысокое быстродействие. В реальных же схемах для увеличения скорости работы применяется отдельная схема формирования переносов для каждого двоичного разряда.

Таблицу истинности для такой схемы легко получить из принципов суммирования двоичных чисел, а затем применить хорошо известные принципы построения схемы по произвольной таблице истинности.

Счетчики

Счетчик – это устройство, предназначенное для подсчета числа входных импульсов. Число, представляющее состояние его выходов с приходом нового импульса изменяется на единицу. Счетчик можно реализовать на нескольких триггерах. В суммирующих счетчиках каждый входной импульс увеличивает число на его выходе на единицу, в вычитающих счетчиках каждый входной импульс уменьшает это число на единицу. Наиболее простые счетчики – двоичные. На рис. 5 представлен суммирующий двоичный счетчик. Диаграммы работы двоичного суммирующего счетчика представлена на рис. 6.

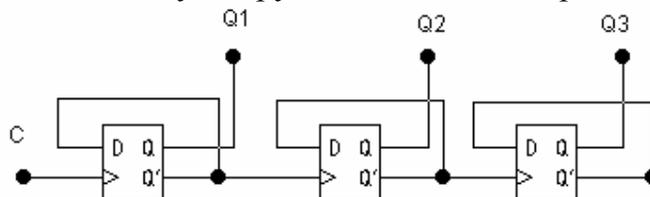


Рис. 5 – Суммирующий двоичный счетчик

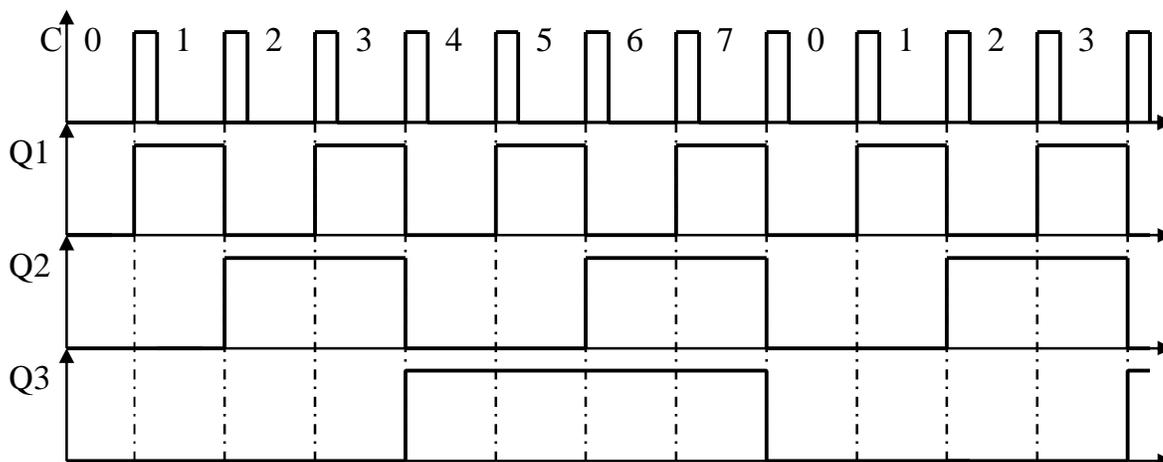


Рис. 6 – Диаграмма работы двоичного суммирующего счетчика

1. Изменение направления счета

Как уже говорилось ранее, счетчики можно реализовать на триггерах. При этом триггеры соединяют последовательно. Выход каждого триггера непосредственно действует на тактовый вход следующего. Для того, чтобы реализовать суммирующий счетчик, необходимо счетный вход очередного триггера подключать к инверсному выходу предыдущего. Для того чтобы изменить направление счета (реализовать вычитающий счетчик), можно предложить следующие способы:

а) Считывать выходные сигналы счетчика не с прямых, а с инверсных выходов триггеров. Число, образуемое состоянием инверсных выходов триггеров счетчика, связано с числом, образованным состоянием прямых выходов триггеров следующим соотношением:

$$N_{np} = 2^n - N_{инв} - 1 \quad (2)$$

где n – разрядность выхода₁ счетчика. В таблице 4 приведен пример связи числа на прямых выходах с числом на инверсных выходах триггеров счетчика.

Таблица 4 – Связь между прямыми и инверсными выходами счетчика

Состояние прямых выходов			Число	Состояние инверсных выходов			Число
Q3	Q2	Q1	N	$\bar{Q}3$	$\bar{Q}2$	$\bar{Q}1$	N
0	0	0	0	1	1	1	7
0	0	1	1	1	1	0	6
0	1	0	2	1	0	1	5

б) Изменить структуру связей в счетчике: подавать на счетный вход следующего триггера сигнал не с инверсного, а с прямого выхода предыдущего, как показано на рис. 7. Временная диаграмма для такого способа реализации счетчика приведена на рис. 8. В этом случае изменяется последовательность переключения триггеров.

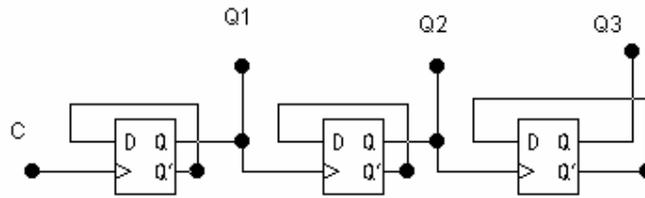


Рис. 7 – Вычитающий счетчик

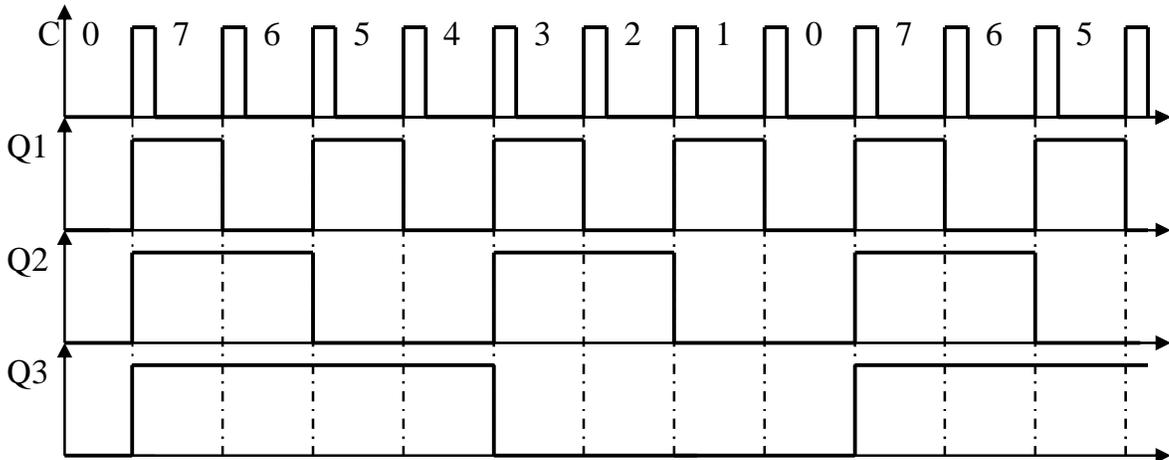


Рис. 8 – Диаграмма работы вычитающего счетчика

2. Изменение коэффициента пересчета

Счетчики характеризуются числом состояний в течение одного периода (цикла). Для схем на рис. 5 и рис. 7 цикл содержит $N=2^3=8$ состояний (от «000» до «111»). Часто число состояний называют *коэффициентом пересчета* $K_{сч}$, который равен отношению числа импульсов N_c на входе к числу импульсов N_{Qcm} на выходе старшего разряда за период:

$$K_{сч} = \frac{N_c}{N_{Qcm}} \quad (3)$$

Если на вход счетчика подавать периодическую последовательность импульсов с частотой f_c , то частота f_Q на выходе старшего, разряда счетчика будет меньше в $K_{сч}$ раз:

$$K_{сч} = \frac{F_c}{F_Q} \quad (4)$$

Поэтому счетчики также называют делителями частоты, а величину $K_{сч}$ – коэффициентом деления. Для увеличения величины $K_{сч}$ приходится увеличивать число триггеров в цепочке. Каждый дополнительный триггер удваивает число состояний счетчика и число $K_{сч}$. Для уменьшения коэффициента $K_{сч}$ можно в качестве выхода счетчика рассматривать выходы триггеров промежуточных каскадов.

Например, для счетчика на трех триггерах $K_{сч}=8$, если взять выход 1 и 2-го триггера, то $K_{сч}=4$. При этом $K_{сч}$ является целой степенью числа 2: 2, 4, 8, 16 и т. д.

Можно реализовать счетчик, для которого $K_{сч}$ – любое целое число. Например, для счетчика на трех триггерах можно сделать $K_{сч}$ от «2» до «7», но при этом один или два триггера могут оказаться лишними. При использовании всех трех триггеров можно получить $K_{сч}=5...7$:

$$2^2 < K_{сч} < 2^3 \quad (5)$$

Счетчик с $K_{сч}=5$ должен иметь 5 состояний, которые в простейшем случае образуют последовательность: $\{0, 1, 2, 3, 4\}$. Циклическое повторение этой последовательности означает, что коэффициент деления счетчика равен «5».

Для построения суммирующего счетчика с $K_{сч}=5$ надо, чтобы после формирования последнего числа из последовательности $\{0, 1, 2, 3, 4\}$ счетчик переходил не к числу «5», а к числу «0». В двоичном коде это означает, что от числа «100» нужно перейти к числу «000», а не «101». Изменение естественного порядка счета возможно при введении дополнительных связей между триггерами счетчика. Можно воспользоваться следующим способом: как только счетчик попадает в нерабочее состояние (в данном случае «101»), этот факт должен быть опознан и повлечь последующую выработку сигнала, который перевел бы счетчик в состояние «000». Рассмотрим этот способ более детально.

Факт попадания счетчика в нерабочее состояние описывается логическим уравнением:

$$\begin{aligned} F &= (101) \vee (110) \vee (111) = Q_3 \cdot \bar{Q}_2 \cdot Q_1 \vee Q_3 \cdot Q_2 \cdot \bar{Q}_1 \vee Q_3 \cdot Q_2 \cdot Q_1 = \\ &= Q_3 \cdot Q_1 \vee Q_3 \cdot Q_2 \end{aligned} \quad (6)$$

Состояния «110» и «111» также являются нерабочими и поэтому учтены при составлении уравнения. Если на выходе эквивалентной логической схемы $F=0$, значит, счетчик находится в одном из рабочих состояний: «0» или «1» или «2» или «3» или «4». Как только он попадает в одно из нерабочих состояний «5» или «6» или «7», формируется сигнал $F=1$. Появление сигнала $F=1$ должно переводить счетчик в начальное состояние «000», следовательно, этот сигнал нужно использовать для воздействия на установочные входы триггеров счетчика, которые осуществляли бы сброс в состояние $Q_1 = Q_2 = Q_3 = 0$. При реализации счетчика на триггерах с входами установки логическим нулем для сброса триггеров требуется подать на входы сброса сигнал $R'=0$, следовательно, логическую функцию F необходимо инвертировать. Для обнаружения факта попадания в нерабочее состояние используем схему, реализующую функцию F и выполненную на элементах И-НЕ. Для этого преобразуем выражение для функции:

$$\bar{F} = \overline{Q_3 \cdot Q_1 \vee Q_3 \cdot Q_2} = Q_3 \cdot (\bar{Q}_1 \cdot \bar{Q}_2) \quad (7)$$

Соответствующая схемная реализация приведена на рис. 9.

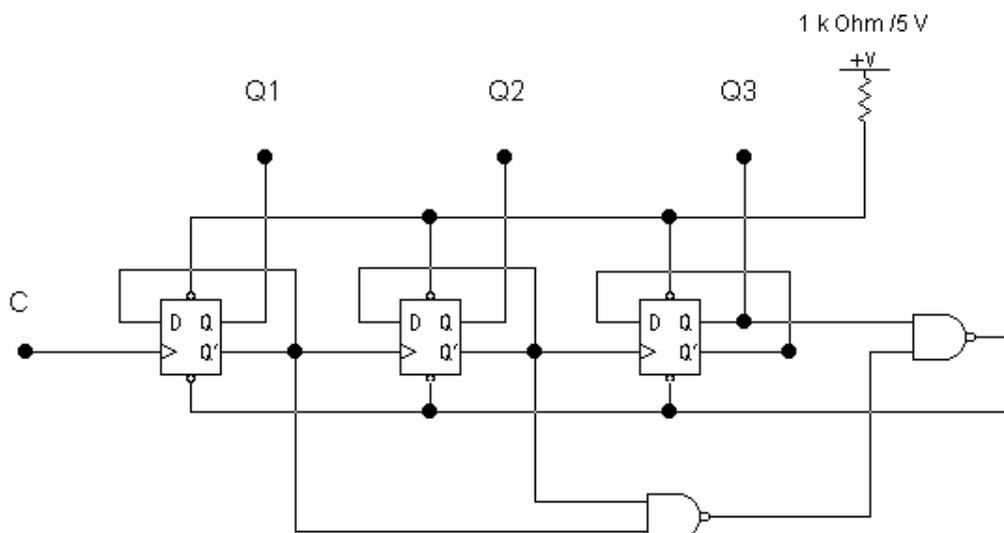


Рис. 9 – Счетчик с измененным коэффициентом пересчета

Счетчик будет работать следующим образом: при счете от «0» до «4» все происходит как в обычном суммирующем счетчике с $K_{сч}=8$. Установочные сигналы равны «1» и естественному порядку счета не препятствуют. Счет происходит по положительному фронту импульса на счетном входе C . В тот момент, когда счетчик находится в состоянии «4» («100»), следующий тактовый импульс сначала переводит счетчик в состояние «5» («101»), что немедленно (задолго до прихода следующего тактового импульса) приводит к формированию сигнала сброса, который поступает на установочный вход R' триггеров. В результате счетчик сбрасывается в «0» и ждет прихода следующего тактового импульса на счетный вход. Один цикл счета закончился, счетчик готов к началу следующего цикла.

Применяя такие схемы с обратной связью для сброса счетчика, нужно иметь в виду, что операция сброса занимает конечное время, поэтому непосредственно перед сбросом счетчика в «0» на выходе первого триггера появляются кратковременные импульсы, или «иголки». Это не имеет значения при подключении счетчика напрямую к индикатору, но при использовании этого выхода счетчика в качестве источника тактовых импульсов могут возникнуть определенные проблемы. Схема, в которой это явление устранено, приведена на рис. 10.

Важным отличием является то, что схема обнаруживает не факт попадания в нерабочее состояние «101», а факт попадания в состояние «100» и в следующем такте вырабатывает сигнал сброса.

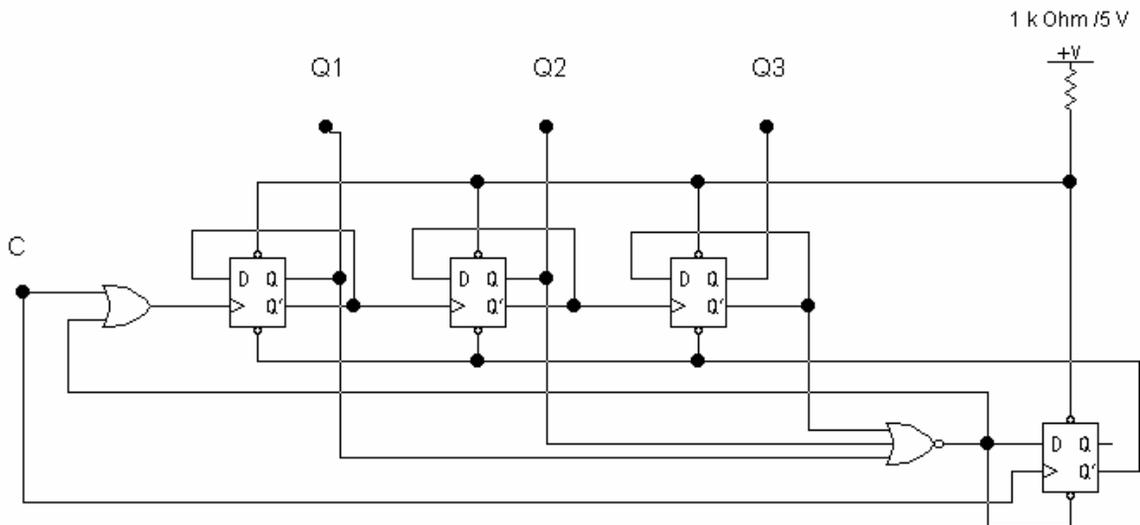


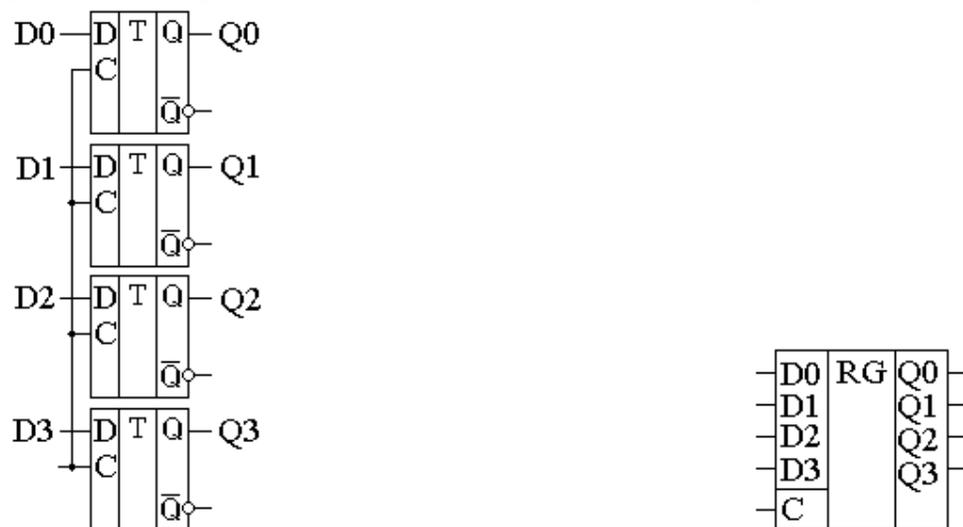
Рис. 10 – Счетчик с измененным коэффициентом пересчета

Регистры

Регистром называется последовательное или параллельное соединение триггеров. Регистры обычно строятся на основе *D-триггеров*. При этом для построения регистров могут использоваться как универсальные *D-триггеры*, так и триггеры-защелки.

1. Параллельный регистр

Параллельный регистр служит для запоминания многоразрядного двоичного слова. При использовании для построения параллельного регистра *триггеров-защелок* регистр называется *регистр-защелка*. Количество триггеров, входящее в состав параллельного регистра определяет его разрядность. При записи информации в параллельный регистр все биты (двоичные разряды) записываются одновременно. Схема четырёхразрядного параллельного регистра приведена на рисунке 11.1, а его обозначение – на рисунке 11.2.



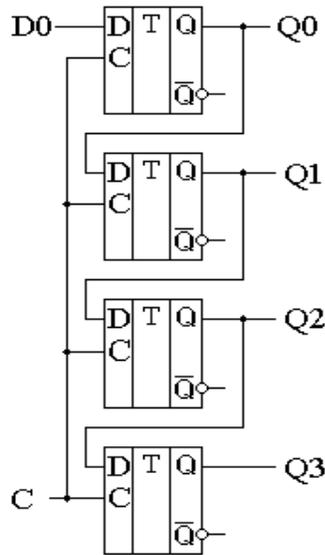
1) Расширенная схема

2) Условное обозначение

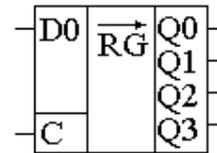
Рис. 11 – Схема параллельного регистра

2. Последовательный регистр

Последовательный регистр (регистр сдвига) обычно служит для преобразования последовательного кода в параллельный и наоборот. Схема регистра, осуществляющего преобразование последовательного кода в параллельный, приведена на рисунке 12.1, а его изображение на принципиальных схемах – на рисунке 12.2.



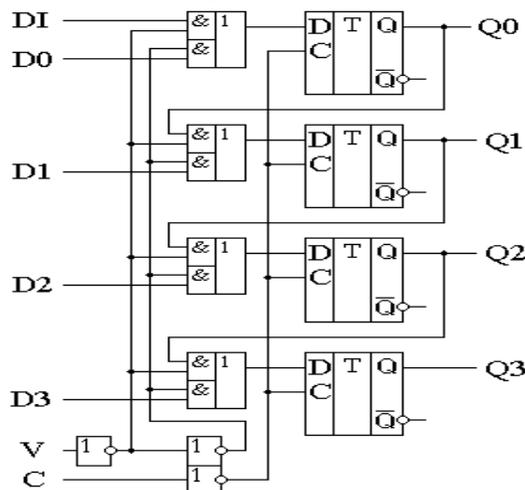
1) Расширенная схема



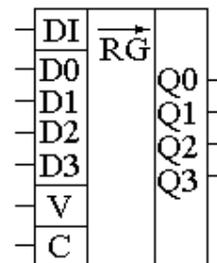
2) Условное обозначение

Рис. 12 – Схема последовательного регистра

Регистры сдвига выполняют обычно как универсальные *последовательно-параллельные микросхемы*. Переключение регистра из параллельного режима в последовательный и наоборот осуществляется при помощи мультиплексо-ра. Схема такого регистра приведена на рисунке 13.1, а его изображение на принципиальных схемах – на рисунке 13.2.



1) Расширенная схема



2) Условное обозначение

Рис. 13 – Схема универсального регистра

Порядок работы

Задание 1. Изучение полусумматоров и сумматоров

а) Изучение работы полусумматора

Соберите схему, приведенную на рис. 1.а.

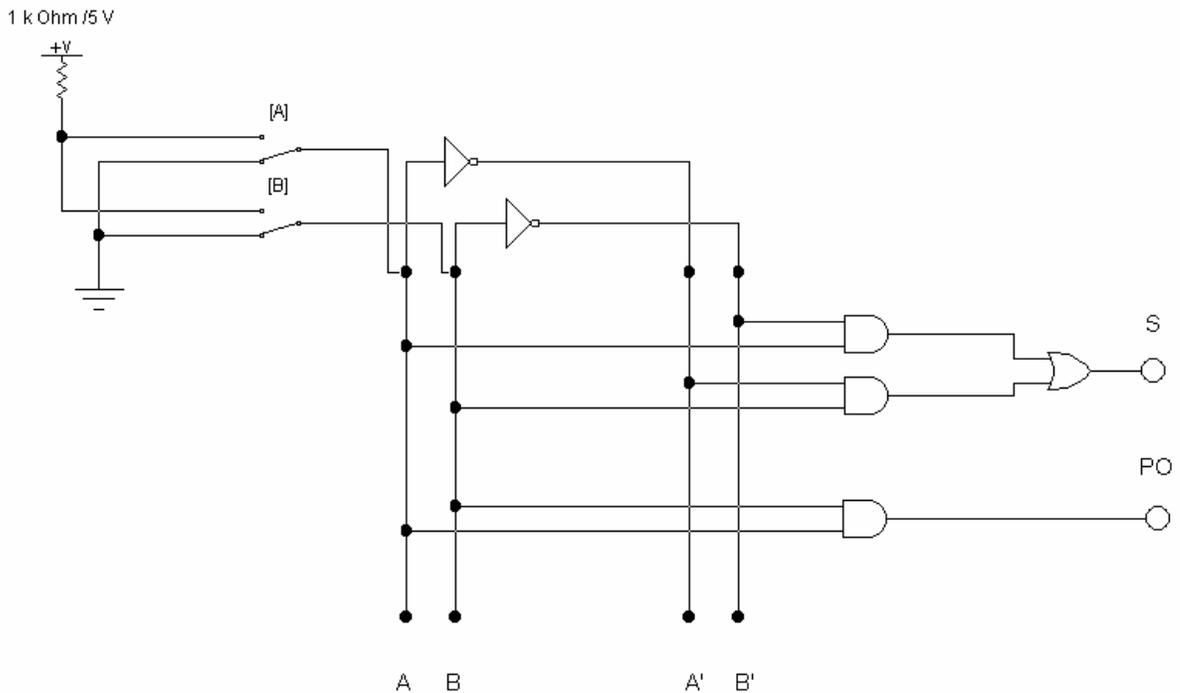


Рис. 1.а – Схема двоичного полусумматора

Задавая различные комбинации логических уровней, на входах полусумматора A и B , заполните таблицу истинности 1.а.

Таблица 1.а. – Таблица истинности полусумматора

Входы		Выходы	
A	B	S	PO
0	0		
0	1		
1	0		
1	1		

Сравните полученную таблицу с таблицей 1 краткой теории, сделайте выводы.

б) Изучение работы одноразрядного сумматора

Соберите схему, приведенную на рис. 1.б.

Задавая различные комбинации логических уровней на входах сумматора A и B , а так же уровень сигнала переноса из предыдущего разряда PI заполните таблицу истинности 1.б.

Сравните полученную таблицу с таблицей 2 краткой теории, сделайте **ВЫВОДЫ**.

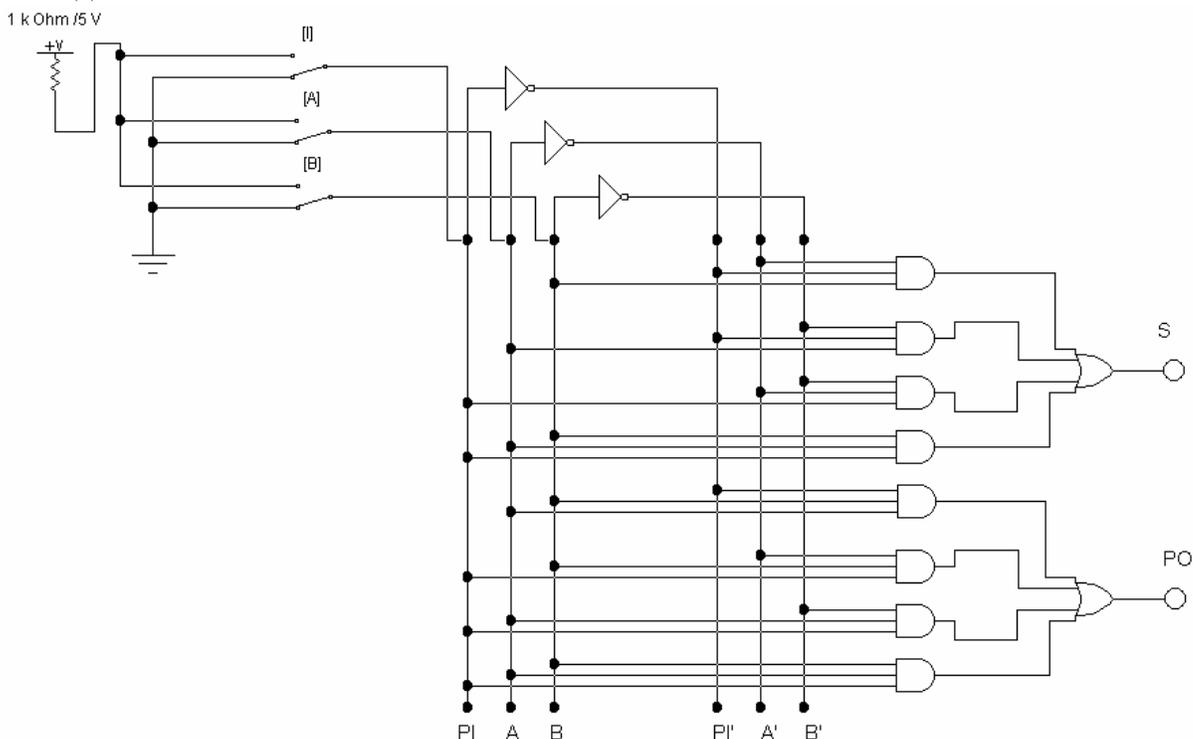


Рис. 1.б – Схема одноразрядного двоичного сумматора

Таблица 1.б. – Таблица истинности одноразрядного сумматора

Входы			Выходы	
PI	A	B	S	PO
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

в) *Изучение работы двоичного трехразрядного сумматора*

Соберите схему, аналогичную рис. 1.в.

Здесь в качестве сумматоров используются специальные блоки EWB «Half-Adder» имеющие аналогично условному обозначению рис. 3.2 два входа A и B , выход сума «S» (в нашем случае на рис. 1.в – « Σ ») и сигнал переноса в следующий разряд «PO» (в нашем случае на рис. 1.в – « C_0 »). В отличие от сумматора, приведенного на рис. 3.2 у данного сумматора нет учета переноса, как из предыдущего разряда, так и в следующий. Для учета переноса из

предыдущих разрядов в схеме рис. 1.в используются дополнительные сумматоры и логические элементы.

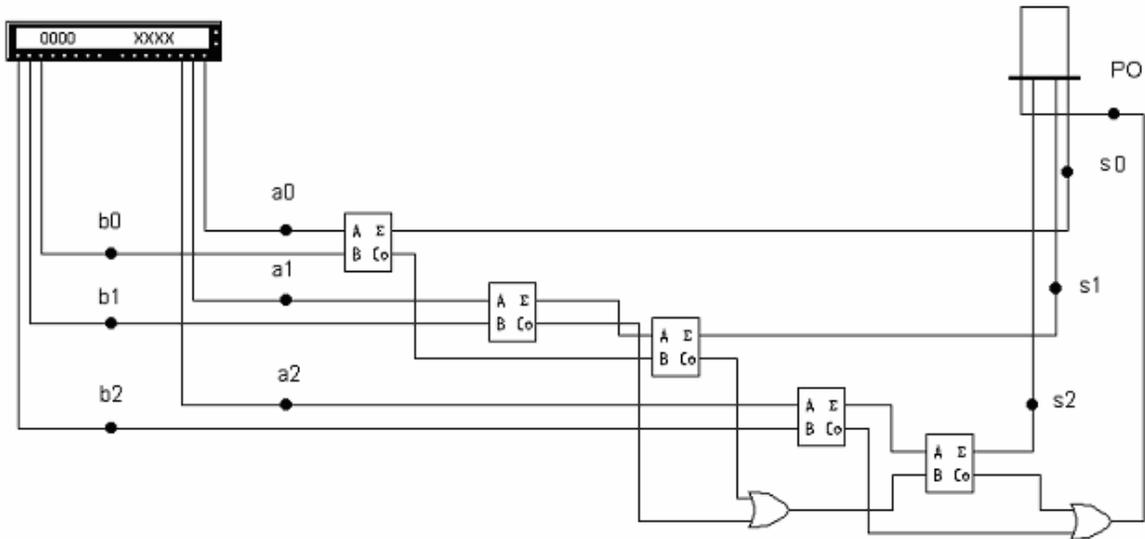


Рис. 1.в – Схема трехразрядного двоичного сумматора

Для проверки правильности функционирования данной схемы необходимо предварительно запрограммировать генератор слов так, что бы на его используемых выходах формировались все возможные комбинации суммируемых двоичных числе «a2 a1 a0» и «b2 b1 b0». Ниже приведена таблица, поясняющая принцип программирования генератора слов (таблица 1.в).

Таблица 1.в – Программирование генератора слов

Шестнадцатеричный код	Двоичный код
0000	0000 0000
0001	0000 0001
0002	0000 0010
0003	0000 0011
0004	0000 0100
0005	0000 0101
0006	0000 0110
0007	0000 0111
2000	0010 0000
2001	0010 0001
2002	0010 0010
2003	0010 0011
2004	0010 0100
2005	0010 0101
2006	0010 0110
2007	0010 0111
4000	0100 0000
...	...
4007	0100 0111
6000	0110 0000
...	...
6007	0110 0111

Продолжение таблицы 1.в

Шестнадцатеричный код	Двоичный код
8000	1000 0000
...	...
8007	1000 0111
A000	1010 0000
...	...
A007	1010 0111
C000	1100 0000
...	...
C007	1100 0111
E000	1110 0000
...	...
E007	1110 0111

Таким образом, из таблицы 1.в видно, что при подключении к трем младшим разрядам генератора слов одного двоичного числа «a2 a1 a0», а к трем старшим разрядам другого двоичного числа «b2 b1 b0» можно просуммировать все возможные комбинации двух трех разрядных чисел.

Примечание: генератор слов удобнее программировать в шестнадцатеричном коде (задание 5.б лабораторной работы №1); семисегментный дисплей («Decoded Seven-Segment Display») сразу декодирует двоичное число (переводит двоичное число в шестнадцатеричное) «s2 s1 s0» и отображает его в удобном для анализа виде.

Запрограммировав генератор слов, необходимо изучить работу трехразрядного двоичного сумматора, заполнив таблицу 1.г.

Таблица 1.г – Суммирование в трехразрядном двоичном сумматоре

Входы, в различных кодах						Выходы, в различных кодах						
2 ^{ый}						10 ^{ый}	16 ^{ый}	2 ^{ый}				10 ^{ый}
a2	a1	a0	b2	b1	b0	b2b1b0	s2s1s0	PO	s2	s1	s0	s2s1s0
0	0	0	0	0	0							
0	0	1	0	0	0							
0	1	0	0	0	0							
0	1	1	0	0	0							
1	0	0	0	0	0							
1	0	1	0	0	0							
1	1	0	0	0	0							
1	1	1	0	0	0							
0	0	0	0	0	1							
0	0	1	0	0	1							
0	1	0	0	0	1							
0	1	1	0	0	1							
1	0	0	0	0	1							
1	0	1	0	0	1							
1	1	0	0	0	1							
1	1	1	0	0	1							

Продолжение таблицы 1.г

Входы, в различных кодах						Выходы, в различных кодах						
2 ^{ый}						10 ^{ый}	16 ^{ый}	2 ^{ый}				10 ^{ый}
a2	a1	a0	b2	b1	b0	b2b1b0	s2s1s0	PO	s2	s1	s0	s2s1s0
0	0	0	0	1	0							
...
1	1	1	0	1	0							
0	0	0	0	1	1							
...
1	1	1	0	1	1							
0	0	0	1	0	0							
...
1	1	1	1	0	0							
0	0	0	1	0	1							
...
1	1	1	1	0	1							
0	0	0	1	1	0							
...
1	1	1	1	1	0							
0	0	0	1	1	1							
...
1	1	1	1	1	1							

Задание 2. Исследование работы суммирующего и вычитающего счетчиков

а) Исследование первой реализации счетчика

Соберите схему изображенную на рис. 2.а.

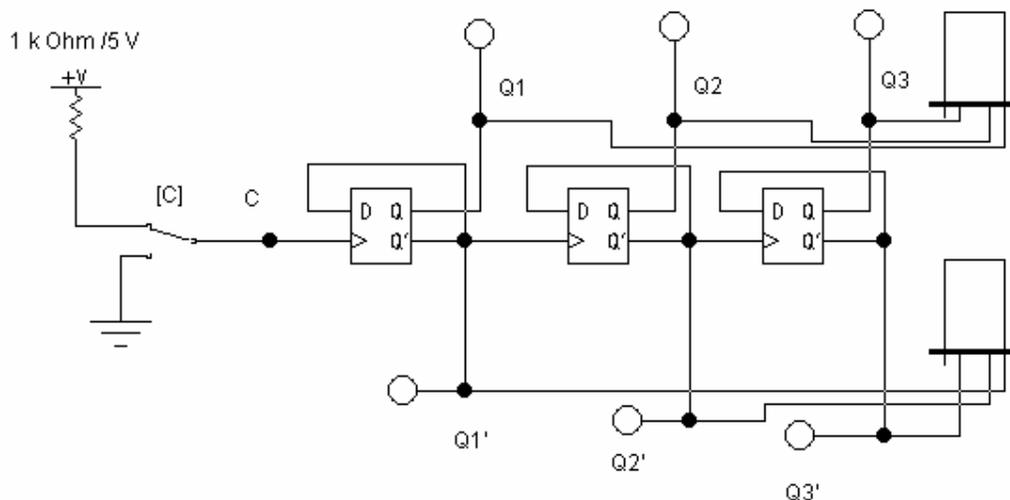


Рис. 2.а – Первая реализация счетчика

Включите схему. Подавая на вход схемы тактовые импульсы при помощи ключа *C* и наблюдая состояние выходов счетчика при помощи семи-сегментного дисплея декодера и логических пробников, составьте временные

диаграммы работы суммирующего (вычитающего) счетчика. Сделайте выводы о работе схемы.

Примечание: в данном случае семисегментный дисплей предназначен для автоматического перевода поступающего на его вход двоичного числа в шестнадцатеричное и отображение последнего.

б) Исследование второй реализации счетчика

Создайте вторую реализацию суммирующего и вычитающего счетчика (см. теорию).

Занесите полученную схему.

Составьте временные диаграммы данной реализации.

Задание 3. Исследование счетчика с измененным коэффициентом пересчета

а) Первый вариант

Соберите схему, изображенную на рис. 3.а.

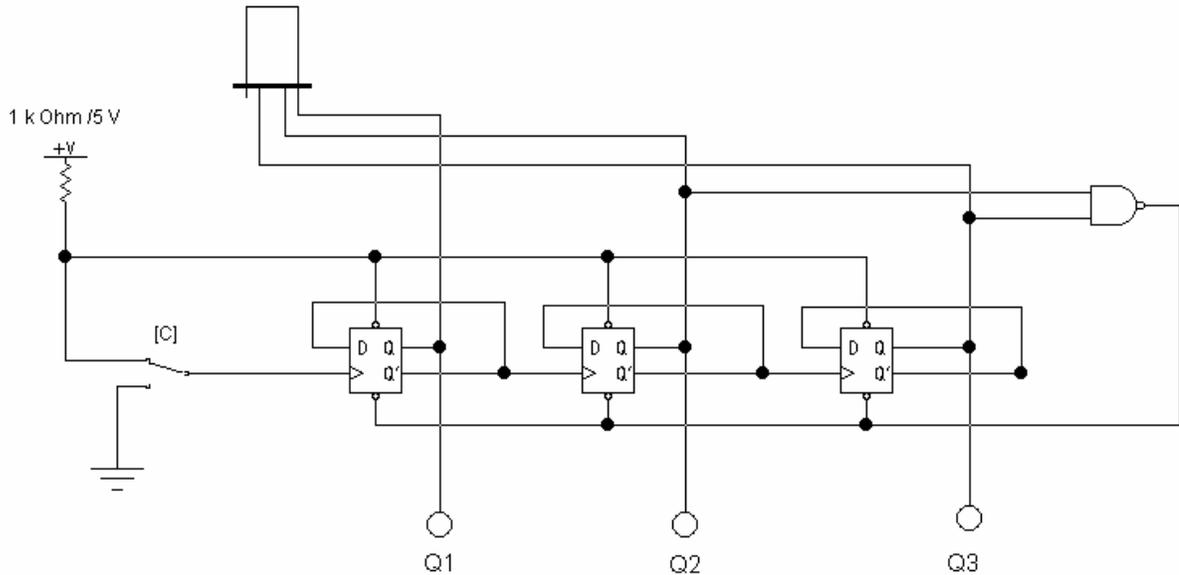


Рис. 3.а – Счетчик с измененным коэффициентом пересчета

Включите схему. Подавая на вход схемы тактовые импульсы при помощи ключа *C* и наблюдая состояние выходов счетчика при помощи логических пробников, составьте временные диаграммы работы счетчика и определите коэффициент пересчета.

б) Второй вариант

Соберите схему изображенную на рис. 3.б.

Включите схему. Подавая на вход схемы тактовые импульсы, при помощи ключа *C* и наблюдая состояние выходов счетчика при помощи логических пробников, составьте временные диаграммы работы счетчика и определите коэффициент пересчета.

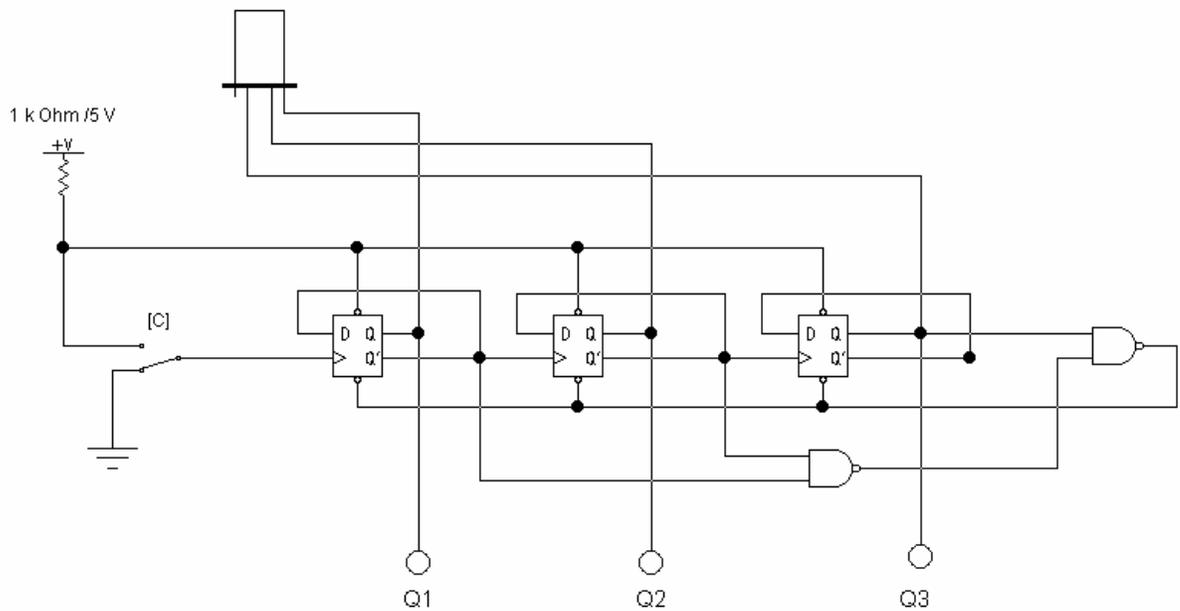


Рис. 3.б – Счетчик с измененным коэффициентом пересчета

Задание 4. Исследование регистров

а) *Исследование параллельного регистра*

Создайте схему, изображенную на рис. 4.а.

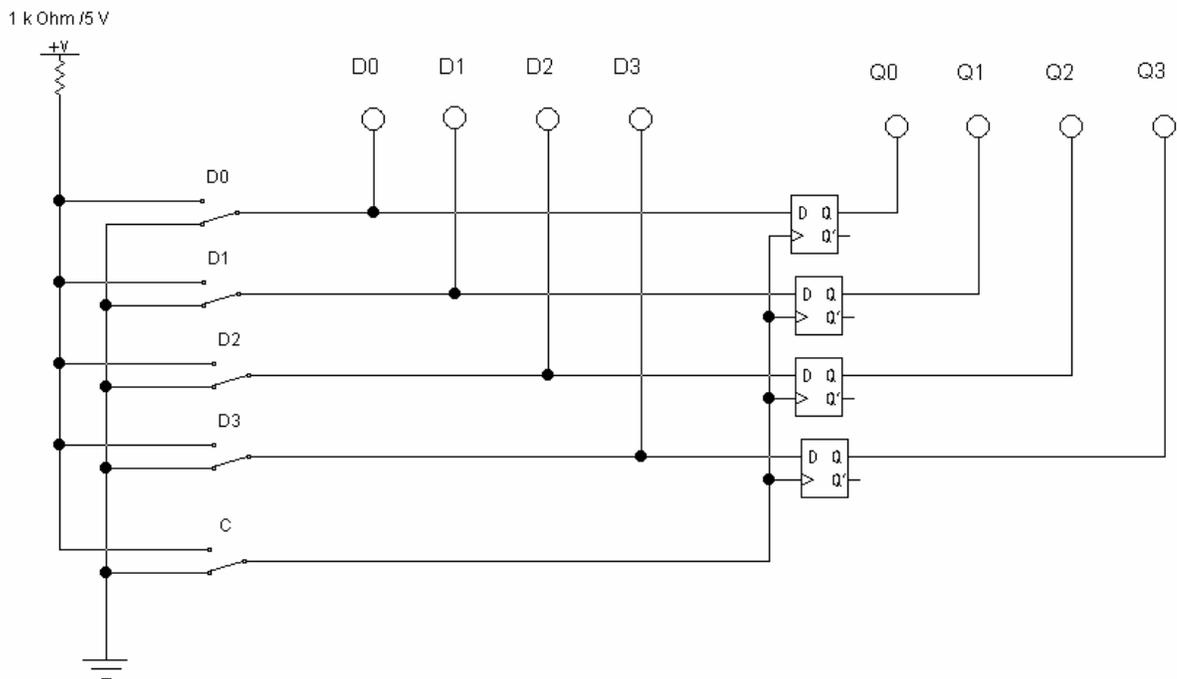


Рис. 4.а – Схема четырехразрядного параллельного регистра

Включите схему. Исследуйте работу полученного двоичного четырехразрядного регистра, заполнив таблицу истинности, приведенную ниже (таблица 4.а).

Таблица 4.а – Параллельный регистр

Входы				Выходы			
D0	D1	D2	D3	Q0	Q1	Q2	Q3
0	0	0	0				
0	0	0	1				
0	0	1	0				
0	0	1	1				
0	1	0	0				
0	1	0	1				
0	1	1	0				
0	1	1	1				
1	0	0	0				
1	0	0	1				
1	0	1	0				
1	0	1	1				
1	1	0	0				
1	1	0	1				
1	1	1	0				
1	1	1	1				

Примечание: запись числа в регистр происходит по переднему фронту разрезающего импульса и сохраняется до появления следующего импульса.

б) Исследование параллельного регистра

Создайте схему, изображенную на рис. 4.б.

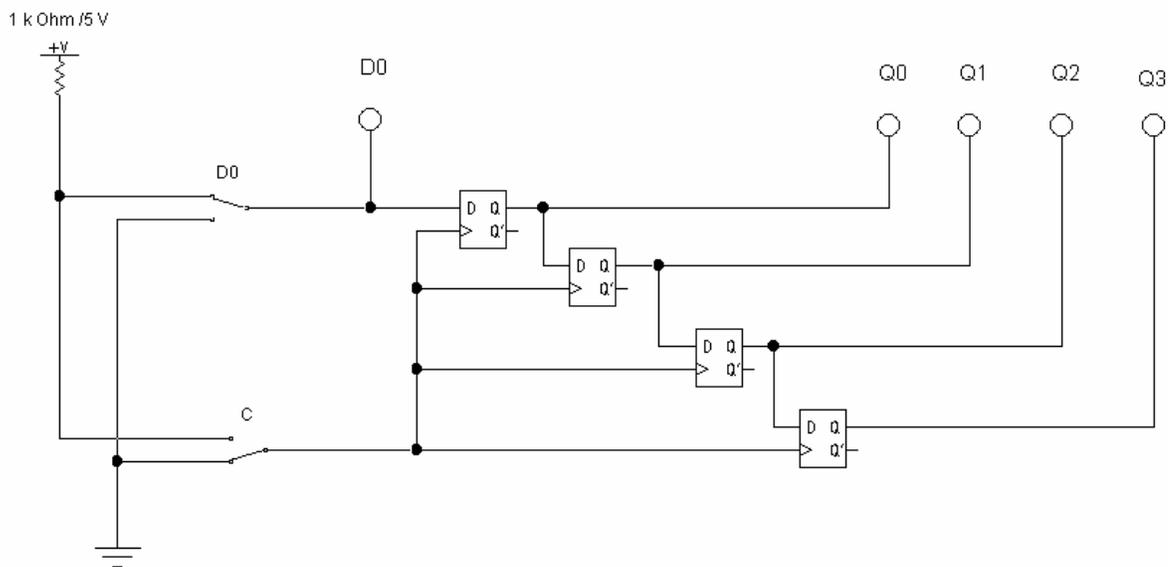


Рис. 4.б – Схема четырехразрядного последовательного регистра

Включите схему. Согласно вашему варианту из таблицы 4.б необходимо выбрать двоичное число, которое требуется получить на выходе регистра сдвига (последовательного регистра).

Задавая необходимые комбинации на входе $D0$ и занося информацию в регистр с помощью сигнала разрешения C , добейтесь получения на выходе $Q0 Q1 Q2 Q3$ требуемого числа.

Постройте временную диаграмму получения в регистре сдвига требуемого числа.

Таблица 4.б – Варианты задания числа для последовательного регистра

Вариант	Число			
	Q0	Q1	Q2	Q3
1	0	0	1	1
2	0	1	0	0
3	0	1	0	1
4	0	1	1	0
5	0	1	1	1
6	1	0	0	0
7	1	0	0	1
8	1	0	1	0
9	1	0	1	1
10	1	1	0	0

Контрольные вопросы

1. Дайте определение сумматора.
2. Опишите принцип работы сумматоров.
3. Чем отличается полусумматор от сумматора?
4. Как можно составить схему четырех разрядного сумматора в EWB?
5. Приведите примеры применения сумматоров.
6. На основе, каких видов триггеров можно получить схему счетчика?
7. Как из суммирующего счетчика получить вычитающий?
8. Что такое коэффициент пересчета счетчика?
9. Как можно вычислить коэффициент пересчета счетчика?
10. Как можно изменить коэффициент пересчета счетчика?
11. Дайте определение регистра.
12. Чем отличается последовательный регистр от параллельного?
13. Приведите примеры использования параллельных регистров.
14. Приведите примеры использования последовательных регистров.

ЛАБОРАТОРНАЯ РАБОТА № 5

Изучение сетевых возможностей Windows NT

Цель работы:

1. Ознакомиться с администрированием Windows NT.
2. Изучить свойства встроенных учетных записей и групп.
3. Ознакомиться с применением средств системной политики и аудита.

4. Рассмотреть сетевые свойства файловой системы NTFS.
5. Научиться управлять доступом к сетевым и локальным ресурсам.

Краткие теоретические сведения

Концепции Windows NT

Операционная система Windows NT реализована в двух вариантах: Windows NT Server и Windows NT Workstation. Windows NT Server 4.0 – сетевая операционная система с приложениями для Internet, сервисами файлов и печати, службой удаленного доступа, встроенным маршрутизатором, индексированием файлов и управлением сетью. Второй вариант Windows NT – Windows NT Workstation 4.0 во многом напоминает NT Server, но она оптимизирована в качестве операционной системы для рабочей станции. С точки зрения архитектуры и возможностей Windows NT Server является надмножеством Windows NT Workstation и включает в себя все возможности последней.

Средства администрирования Windows NT

Администрирование Windows NT включает как задачи настройки системы непосредственно после инсталляции, так и задачи ежедневной поддержки системы.

Задачи администрирования Windows NT Server и Windows NT Workstation во многом совпадают, однако средства для выполнения этих задач, включенные в каждый из этих продуктов, различаются.

Задачи администрирования могут быть разделены на пять групп:

- Администрирование пользователей и групп пользователей. Включает планирование, создание и поддержку учетной информации пользователей и групп.
- Администрирование средств обеспечения безопасности системы. Включает планирование и реализацию политики безопасности, гарантирующей защиту данных и разделяемых сетевых ресурсов, таких как каталоги, файлы и принтеры.
- Администрирование принтеров. Включает инсталляцию локальных и сетевых принтеров, конфигурирование их для более удобного использования, поиск неисправностей, устранение проблем, возникающих при печати.
- Мониторинг событий и ресурсов. Включает планирование и реализацию политики аудита сетевых событий в целях нахождения брешей в системе защиты, а также мониторинг процессов использования сетевых ресурсов.
- Архивирование и восстановление данных. Включает планирование и выполнение регулярного резервного копирования критических данных.

Средства администрирования Windows NT Workstation используются только для администрирования локального компьютера, а средства администрирования Windows NT Server используются для администрирования любого компьютера сети.

Важнейшими инструментами администрирования пользователей для Windows NT Workstation является *User Manager*, а для Windows NT Server – *User Manager for Domains*.

Другим важным средством администрирования является *Server Manager*. Это средство, включенное в состав Windows NT Server, дает возможность просматривать и управлять компьютерами домена. С помощью Server Manager можно:

- присоединить компьютер к домену;
- создать на удаленном компьютере разделяемый каталог (share) и определить к нему права доступа для пользователей домена;
- просмотреть список работающих на удаленном компьютере сервисов, запускать или останавливать эти сервисы;
- инициировать синхронизацию контроллеров домена, повысит роль BDC (вторичный контроллер домена) до PDC (первичный контроллер домена).

Event Viewer – еще одно средство администрирования, которое оповещает администратора сети обо всех событиях в сети, которые имеют отношение к работоспособности и безопасности системы. Например, такими событиями являются успешный или неуспешный запуск сервиса или приложения, успешная или неуспешная попытка логического входа и т.д.

Backup – средство, которое используется для резервного копирования информации на локальный стример.

Средства администрирования Windows NT Server построены в архитектуре клиент – сервер. После инсталляции Windows NT Server клиентские и серверные части средств администрирования находятся на одном компьютере, но клиентские части могут работать на любых компьютерах сети, работающих под управлением Windows NT Workstation или Windows 9x. Эти клиентские части собраны в одном каталоге Clients\Srvtools дистрибутивного диска Windows NT Server.

Профили пользователей

Профиль пользователя – это набор параметров, определяющих рабочую среду пользователя на том компьютере, на котором он в данный момент работает.

В профиль пользователя входит большое количество переменных:

- состояние рабочего стола (Desktop) – цветовые схемы, обои;
- состояние оболочки (Shell) – элементы графического интерфейса, содержимое папок;
- постоянные сетевые подключения;
- подключаемые сетевые принтеры.

Обычно пользователь сам создает настройки своей рабочей среды, а после выхода его из системы эти настройки сохраняются в файле профиля данного пользователя, а также в Registry. При очередном логическом входе пользователя параметры его рабочей среды берутся из Registry и восстанавливаются для него требуемые установки.

Профиль может также содержать ограничения для пользователя на выполнение некоторых действий. К таким действиям относится возможность использования команды Run, доступ к функциям Control Panel, Settings в меню Start, команды Shut Down и некоторые другие действия, которые могут изме-

нить облик системы или позволят получить пользователю доступ к таким функциям или программам, которые недопустимы с точки зрения администратора.

Для ограничения возможностей пользователя с помощью профиля существует специальная утилита *System Policy Editor* – Редактор Системной Политики, входящая в комплект поставки Windows NT Server 4.0. Эта утилита заменила собой User Profile Editor, использовавшейся для этих же целей в Windows NT 3.51. Профили, создаваемые System Policy Editor, теперь воздействуют не только на пользователей, работающих с Windows NT, но и пользователей Windows 9x.

Администрирование пользователей

1. Пользователи, ресурсы и операции доступа

Администрирование пользователей состоит в создании учетной информации пользователей (определяющей имя пользователя, принадлежность пользователя к различным группам пользователей, пароль пользователя), а также в определении прав доступа пользователя к ресурсам сети – компьютерам, каталогам, файлам, принтерам и т.п.

Создание учетной информации пользователей, называемой так же учетной записью, осуществляется в сети Windows NT утилитой User Manager для локального компьютера и User Manager for Domains для всех компьютеров домена. Права доступа к ресурсам задаются в сети Windows NT различными средствами, в зависимости от типа ресурса. Возможность использования компьютеров Windows NT Workstation в качестве рабочих станций – с помощью User Manager for Domains, доступ к локальным каталогам и файлам (только для файловой системы NTFS, поддерживающей права доступа) – с помощью средств Windows NT Explorer, к удаленным разделяемым каталогам – с помощью Server Manager, доступ к принтерам – из панели Printers.

1.1. Типы пользователей и групп пользователей

В сети Windows NT могут быть определены следующие типы пользователей и групп пользователей:

- *локальный интерактивный пользователь компьютера* (пользователь, который заведен в локальной учетной базе данных компьютера, и который работает с ресурсами компьютера интерактивно);
- *локальный сетевой пользователь компьютера* (пользователь, который заведен в локальной учетной базе данных компьютера, и который работает с ресурсами компьютера через сеть);
- *пользователь домена* (пользователь, который заведен в глобальной учетной базе данных домена на PDC);
- *локальная группа компьютера* (может создаваться на всех компьютерах домена, кроме PDC и BDC, в которых она вырождается в локальную группу домена);
- *локальная группа домена* - состоит из пользователей домена (заводится только на PDC);
- *глобальная группа домена* - состоит из пользователей домена (может входить в локальную группу домена).

Для каждого типа групп имеется некоторый набор встроенных групп: Administrators, Server Operators, Users, Everyone, DomainUsers и др.

Для однозначной идентификации глобальной группы в многодоменной сети, используется составное имя, например Marketing\Managers, где Marketing – имя домена, Managers – имя глобальной группы.

1.2. Типы объектов

- *Каталоги и файлы.* Процедуры задания правил доступа различаются для локальных и разделяемых (share) каталогов и файлов. Операции: read, full control, change, add, ...;
- *Принтеры;*
- *Операционная система.* По отношению к этому типу объектов определяются права по выполнению различных сервисов и утилит: вход, архивирование файлов, изменение конфигурации панелей Program Manager,

1.3. Типы операций доступа

Операции доступа – это действия объектов над субъектами. Операции могут быть либо разрешены, либо запрещены, либо вообще не иметь смысла для данной пары объекта и субъекта.

Все множество операций разделяется на подмножества, имеющие особые названия:

- разрешения (permissions) – это множество операций, которые могут быть определены для субъектов всех типов по отношению к объектам типа файл, каталог или принтер;
- права (user rights) – определяются для объектов типа группа на выполнение некоторых системных операций: создание резервных копий, выключение компьютера (shutdown) и т.п. Права назначаются с помощью User Manager for Domains;
- возможности пользователей (user abilities) – определяются для отдельных пользователей на выполнение действий, связанных с формированием их операционной среды, например, изменение состава программных групп, показываемых на экране дисплея, включение новых иконок в Desktop, возможность использования команды Run и т.п.

Права и разрешения, данные группе, автоматически предоставляются ее членам, позволяя администратору рассматривать большое количество пользователей как единицу учетной информации.

Возможности пользователей определяются профилем пользователя.

2. Встроенные группы пользователей и их права

Права определяются для объектов типа группа на выполнение некоторых системных операций: создание резервных копий, выключение компьютера (shutdown) и т.п. Права назначаются с помощью User Manager for Domains.

Таблица 1 – Права для встроенных локальных групп

Вид прав	Права							
	Ad-minis-trators	Server Opera-tors	Ac-count Opera-tors	Print Opera-tors	Backup Opera-tors	Eve-ryone	Users	Guests
Log on locally (локальный логический вход)	+	+	+	+	+	-	-	-
Access this computer from network (доступ к данному компьютеру через сеть)	+	-	-	-	-	+	-	-
Take ownership of files (установление прав собственности на файлы)	+	-	-	-	-	-	-	-
Manage auditing and security log (управление аудитингом и учетом событий, связанных с безопасностью)	+	-	-	-	-	-	-	-
Change the system time (Изменение системного времени)	+	+	-	-	-	-	-	-
Shutdown the system (Останов системы)	+	+	-	-	+	-	-	-
Force shutdown from remote system (Инициация останова с удаленной системы)	+	+	-	-	-	-	-	-

Продолжение таблицы 1

Вид прав	Права							
	Ad-minis-trators	Server Opera-tors	Ac-count Opera-tors	Print Opera-tors	Backup Opera-tors	Eve-ryone	Users	Guests
Backup files and directories (Резервное копирование файлов и каталогов)	+	+	+	+	+	-	-	-
Restore files and directories (Восстановление со стримера)	+	+	-	-	+	-	-	-
Load and unload device drivers (Загрузка и выгрузка драйверов устройств)	+	-	-	-	-	-	-	-
Add workstation to domain (Добавление рабочих станций к домену)	+	-	-	-	-	-	-	-
Create and manage user accounts (Создание и управление пользовательской учетной информацией)	+	-	+ ¹	-	-	-	-	-
Create and manage global groups (Создание и управление глобальными группами)	+	-	+ ¹	-	-	-	-	-
Create and manage local groups (Создание и управление локальными группами)	+	-	+ ¹	-	-	-	+ ²	-

Продолжение таблицы 1

Вид прав	Права							
	Ad-minis-trators	Server Opera-tors	Ac-count Opera-tors	Print Opera-tors	Backup Opera-tors	Eve-ryone	Users	Guests
Assign user rights (Назначение прав для пользователей)	+	-	-	-	-	-	-	-
Manage auditing of system events (Управление аудитингом системных событий)	+	-	-	-	-	-	-	-
Lock the server (Блокирование сервера)	+	+	-	-	-	+ ³	-	-
Override the lock of the server (Преодоление блокировки сервера)	+	+	-	-	-	-	-	-
Format server's hard disk (Форматирование жесткого диска сервера)	+	+	-	-	-	-	-	-
Create common groups (Создание общих групп)	+	+	-	-	-	-	-	-
Keep local profile (Хранение локального профиля)	+	+	+	+	+	-	-	-
Share and stop sharing directories (Разделение и прекращение разделения каталогов)	+	+	-	-	-	-	-	-

¹Операторы учетной информации (Accounts operators) не могут изменять учетную информацию администраторов, или же изменять глобальную группу Domain Admins или локальные группы Administrators, Server Operators, Account Operators, Print Operators или Backup Operators.

²Хотя члены группы Users имеют право создавать локальные группы домена, они не смогут им воспользоваться, если им не разрешено входить локально в сервер или не разрешено пользоваться утилитой User Manager for Domains.

³Хотя Everyone имеет право блокировать сервер, только пользователи, которые могут также входить локально в этот сервер могут в действительности его заблокировать.

Похожие права можно задать и по отношению к Windows NT Server, не выполняющему роль PDC или BDC – с помощью утилиты User Manager for Domains, а также к Windows NT Workstation с помощью утилиты User Manager.

3. Возможности пользователей

Возможности пользователей – определяются для отдельных пользователей на выполнение немногочисленных действий, касающихся реорганизации их операционной среды:

1. Включение новых программных единиц (иконок) в группу программ панели Program Manager;
2. Создание программных групп Program Manager;
3. Изменение состава программных групп;
4. Изменение свойств программных единиц (например, включение в стартовую группу);
5. Запуск программ из меню FILE в Program Manager;
6. Установление соединений с сетевым принтером, кроме тех (которые уже предусмотрены в профиле пользователя).

Возможности пользователя являются частью так называемого профиля пользователя (User Profile), который можно изменять с помощью утилиты User Profile Editor. Профиль наряду с описанными возможностями включает и установки среды пользователя на его рабочем компьютере, такие как цвета, шрифты, набор программных групп и их состав.

4. Разрешения на доступ к каталогам и файлам

Администратор может управлять доступом пользователей к каталогам и файлам в разделах диска, отформатированных под файловую систему NTFS. Разделы, отформатированные под FAT и HPFS, не поддерживаются средствами защиты Windows NT. Однако можно защитить разделяемые по сети каталоги независимо от того, какая используется файловая система.

Для защиты файла или каталога необходимо установить для него разрешения (permissions). Каждое установленное разрешение определяет вид доступа, который пользователь или группа пользователей имеют по отношению к данному каталогу или файлу. Например, когда вы устанавливаете разрешение Read к файлу MY IDEAS.DOC для группы COWORKERS, пользова-

тели из этой группы могут просматривать данные этого файла и его атрибуты, но не могут изменять файл или удалять его.

Windows NT позволяет использовать набор стандартных разрешений, которые можно устанавливать для каталогов и файлов. Стандартными разрешениями для каталогов являются: No Access, Read, Add, Add&Read, Change и Full Control.

Стандартными разрешениями для файлов являются: No Access, Read, Change и Full Control.

Стандартные разрешения представляют собой группы индивидуальных разрешений. Каждому стандартному разрешению соответствует определенная установка фиксированного набора индивидуальных разрешений. Индивидуальные разрешения могут быть:

Read (R), Write (W), Execute (X), Delete (D),
Change Permission (P), Take Ownership (O).

При установке стандартного разрешения рядом с ним в скобках отображаются заглавные буквы установленных индивидуальных разрешений. Например, при установке для файла стандартного разрешения Read рядом со словом Read появляется аббревиатура RX, которая означает, что стандартному разрешению Read соответствует установка двух индивидуальных разрешений – Read и Execute.

Администратор может с помощью утилиты File Manager устанавливать как стандартные, так и индивидуальные разрешения.

Для того чтобы эффективно пользоваться возможностями механизмов безопасности NTFS, нужно помнить следующее:

- Пользователи не могут пользоваться каталогом или файлом, если они не имеют разрешения на это, или же они не относятся к группе, которая имеет соответствующее разрешение.
- Разрешения имеют накопительный эффект за исключением разрешения No Access, которое отменяет все остальные имеющиеся разрешения. Например, если группа CO-WORKERS имеет разрешение Change для какого-то файла, а группа Finance имеет для этого файла только разрешение Read, и Петров является членом обеих групп, то у Петрова будет разрешение Change. Однако если разрешение для группы Finance изменится на No Access, то Петров не сможет использовать этот файл, несмотря на то, что он член группы, которая имеет доступ к файлу.
- Когда вы создаете в каталоге файлы и подкаталоги, то они наследуют разрешения, которые имеет каталог.
- Пользователь, который создает файл или каталог, является владельцем (owner) этого файла или каталога. Владелец всегда имеет полный доступ к файлу или каталогу, так как может изменять разрешения для него. Пользователи - члены группы Administrators - могут всегда стать владельцами любого файла или каталога.
- Самым удобным путем управления защитой файлов и каталогов является установка разрешений для групп пользователей, а не для отдельных пользователей. Обычно пользователю требуется доступ ко многим файлам.

Если пользователь является членом какой-либо группы, которая имеет доступ к этим файлам, то администратору проще лишить пользователя этих прав, удалив его из состава группы, а не изменять разрешения для каждого файла. Заметим, что установка разрешения для индивидуального пользователя не отменяет разрешений, данных пользователю как члену некоторой группы.

Таблица 2 – Индивидуальные разрешения для каталога

Вид работы	Разрешения						
	R	W	X	D	P	O	FC
Просматривать имена файлов в каталоге	*	O	*	O	*	O	*
Просматривать атрибуты каталога	*	O	*	O	*	O	*
Добавлять файлы и подкаталоги	O	*	O	*	O	*	*
Изменять атрибуты каталога	O	*	O	*	O	*	*
Переходить в подкаталоги каталога	O	*	*	O	*	O	*
Просматривать владельца каталога и разрешения	*	*	*	O	*	O	*
Удалять каталог	O	*	O	*	O	*	*
Изменять разрешения каталога	O	*	O	*	*	O	*
Становиться владельцем каталога	O	*	O	*	O	*	O

Таблица 3 – Индивидуальные разрешения для файла

Вид работы	Разрешения						
	R	W	X	D	P	O	FC
Просматривать данные файла	*	O	O	O	O	O	*
Просматривать атрибуты файла	*	O	*	O	O	O	*
Изменять атрибуты файла	O	*	O	O	O	O	*
Изменять и добавлять данные в файл	O	*	O	O	O	O	*
Выполнять файл, если это программа	O	O	*	O	O	O	*
Просматривать владельца файла и разрешения	*	*	*	O	O	O	*
Удалять файл	O	O	O	*	O	O	*
Изменять разрешения файла	O	O	O	O	*	O	*
Становиться владельцем файла	O	O	O	O	O	*	*

Для файлов имеется следующее соответствие индивидуальных и стандартных разрешений файла:

No Access – Ни одного

Read – RX

Change – RWXD

Full Control – Все разрешения

Стандартные разрешения для каталога представляют собой объединения индивидуальных разрешений для каталога и для файлов, входящих в этот каталог:

No Access (Ни одного) (Ни одного)

List (RX) (Не определены)

Read (RX) (RX)

Add	(WX) (Не определены)
Add&Read	(RWX) (RX)
Change	(RWXD) (RWXD)
Full Control	(Все разрешения) (Все разрешения)

5. Управление профилями пользователей

Когда пользователь локально входит первый раз в какой-либо компьютер, то для него по умолчанию создается профиль. Все настройки среды (цвет фона, обои, шрифты и т.п.) автоматически сохраняются в подкаталоге Profiles системного каталога данного компьютера, например, C:\NT40w\Profiles\username, где *username* – имя пользователя. Профиль хранится в файле с именем ntuser.dat

Администратор также может настраивать профиль пользователя, входя в какой-либо компьютер под именем этого пользователя.

В отличие от профиля пользователя, который устанавливается по умолчанию, существует также Roaming – перемещаемый профиль пользователя, который формирует одну и ту же среду для данного пользователя, независимо от того, с какого компьютера он вошел в сеть.

Перемещаемые пользовательские профили хранятся, централизованно на сервере, а не на локальных компьютерах пользователей.

Администратор может определить для пользователя один из двух типов перемещаемых профилей.

- Индивидуальный перемещаемый профиль, который пользователь может изменять. Любые изменения, которые пользователь внес в свою среду, вносятся в индивидуальный перемещаемый профиль тогда, когда пользователь логически выходит из сети. Когда тот же пользователь входит снова, с сервера загружается последний вариант профиля. Таким образом, если используются перемещаемые индивидуальные профили, то у каждого пользователя имеется свой собственный перемещаемый профиль. Этот профиль хранится в файле ntuser.dat в одном из разделяемых каталогов сервера.
- Обязательный (mandatory) перемещаемый профиль – это заранее сконфигурированный администратором профиль, который пользователь не может изменить. Один обязательный профиль может быть назначен нескольким пользователям. Этот вид профиля целесообразно назначать тем пользователям, которым требуется одинаковая среда, например, операционистам банка. Обязательный профиль должен иметь расширение .man. Индивидуальный профиль можно сделать обязательным, переименовав его из Ntuser.dat в Ntuser.man.

Начиная с версии 4.0, администратору предлагается более мощное средство управления профилями пользователей – System Policy Editor. С его помощью администратор может изменять профиль пользователя, не входя под его именем. При этом он может устанавливать ограничения, которые невозможно было бы установить, входя под именем пользователя, например, запрет на использование команды Run. System Policy Editor может использоваться для формирования как локальных, так и перемещаемых профилей.

Перемещаемый профиль хранится в файле Ntconfig.pol в разделяемом каталоге Netlogon на PDC.

Примечание: Программы-мастера Administrative Wizards позволяют без труда, шаг за шагом, выполнять такие действия, как создание учетных записей пользователей, управление их группами, контроль доступа к файлам и каталогам, установка нового принтера, инсталляция и деинсталляция программ, подключение модема, подготовка пакетов инсталляционных дискет для новых клиентов и контроль за соблюдением лицензионных соглашений для установленных программ. Однако в данной работе использование мастера не рассматривается.

Порядок работы

Задание 1. Изучение встроенных учетных записей и групп

а) Изучение встроенных групп при установке NT Server как контроллер домена

С помощью User Manager for Domains ознакомиться с составом встроенных групп, определив принадлежность (локальная или глобальная). Результаты занести в таблицу 1.а.

Таблица 1.а – Встроенные группы

Имя	Тип	Члены группы
Account Operators		
Administrators		
Backup Operators		
Guests		
Print Operators		
Replicator		
Server Operators		
Users		
Domain Admins		
Domain Guests		
Domain Users		

б) Изучение встроенных групп при установке NT как обычного сервера

С помощью User Manager for Domains ознакомиться с составом встроенных групп. Результаты занести в таблицу 1.б. Отметить какие учетные записи при этом были созданы.

Таблица 1.б – Встроенные группы

Имя	Члены группы
Administrators	
Power Users	
Guests	
Replicator	
Users	

Задание 2. Исследование учетных записей Administrator и Guest
Изучить учетные записи Administrator и Guest, составить подробное описание принадлежности каждой учетной записи к определенным группам и описать права каждой учетной записи.

Задание 3. Изучение системных привилегий
Исследовать список привилегий, заполнить таблицу привилегий встроенных учетных записей и групп (таблица 3).

Таблица 3 – Системные привилегии

Право	Группы и учетные записи, имеющие данное право
Доступ к компьютеру из сети	
Архивирование файлов и каталогов	
Изменение системного времени	
Завершение работы системы	
Принудительное удаленное завершение	
Загрузка и выгрузка драйверов устройств	
Локальный вход в систему	
Овладение файлами или иными объектами	
Управление аудитом и журналом безопасности	
Добавление станций в домен	
Восстановление файлов и каталогов	

Задание 4. Изучение способов внесения новых пользователей в систему

Задайте ряд пользователей с определенными свойствами и правами:

Обычный пользователь – время от времени играет в игры;

Студент – пользователь, использующий некоторые программы, не имеющий право играть в игры и устанавливать программы;

Преподаватель – имеет право запускать все виды программ, может пользоваться всеми сетевыми устройствами, не имеет прав на установку оборудования и программ;

Сформируйте глобальные группы. Определить привилегии новых пользователей путем включения глобальных групп в локальные или прямым назначением привилегий.

Кратко запишите последовательность действий выполняемых в данном задании.

Задание 5. Изучение способов установки прав доступа к файлам и каталогам

В каталоге C:\TEMP создайте произвольные подкаталоги и файлы и назначьте различные права доступа для созданных ранее пользователей. Войдите в систему от имени пользователей и удостоверьтесь в правильном разграничении привилегий и прав.

Задание 6. Изучение разделяемых файловых ресурсов

Создайте разделяемые файловые ресурсы. Проверьте сетевой доступ к разделяемым ресурсам. Выполните передачу права владения файлом. Заполните таблицу прав для разделяемых файловых ресурсов (таблица 6).

Таблица 6 – Права для разделяемых файловых ресурсов

Право	Описание
No Access	
Read	
Change	
Full Control	

Задание 7. Изучение способов создания персонального профиля

С помощью User Profile Editor создать персональный профиль пользователя, назначить домашний каталог. Опишите различные способы и преимущества каждого способа по созданию и расположению домашнего каталога.

Задание 8. Изучение способов редактирования и создания политик

С помощью System Policy Editor изучите способы создания и редактирования политик. Опишите способ, которым можно сделать политику доступной на любой машине в домене.

12.2. КОНТРОЛЬНЫЕ РАБОТЫ

Контрольная работа №1: «Основы представления информации и алгебры логики»

Задание 1. Переведите из одной системы исчисления в другую:

- из двоичной в десятичную, восьмеричную, шестнадцатеричную числа:

«11», «1011», «011100», «01111001»;

из десятичной в двоичную, восьмеричную, шестнадцатеричную числа:

«1», «20», «2.3» «-6»

Задание 2. Сложите, вычтите, умножьте полученные в вышеприведенном задании числа в различных системах исчисления.

Задание 3. Составьте логические функции по таблице истинности:

Переменные			Значения логических функций					
А	В	С	Вариант 1			Вариант 2		
			f1	f2	f3	F1	F2	F3
0	0	0	0	0	0	0	1	1
0	0	1	1	1	1	0	0	1
0	1	0	1	1	0	1	1	0
0	1	1	0	1	0	1	0	0
1	0	0	0	1	1	0	0	0
1	0	1	1	0	0	0	0	1
1	1	0	0	0	0	1	1	0
1	1	1	0	0	1	0	1	0

Задание 4. Упростите полученные в предыдущем задании логические функций (вариант №1 – логические функции f1, f2, f3; вариант №2 – логические функции F1, F2, F3).

Задание 5. Постройте схемы соответствующих упрощенных логических функций (для варианта №1 – упрощенные логические функции f1, f2, f3; для варианта №2 – упрощенные логические функции F1, F2, F3) в произвольном базисе.

Задание 6. Реализуйте упрощенные логические функций, полученные в четвертом задании (для варианта №1 – упрощенные логические функции f1, f2, f3; для варианта №2 – упрощенные логические функции F1, F2, F3) в базисе «И-НЕ»:

4.1. Запишите соответствующие логические выражения,

4.2. Постройте соответствующие схемы.

Задание 7. Реализуйте упрощенные логические функций, полученные в четвертом задании (для варианта №1 – упрощенные логические функции f1, f2, f3; для варианта №2 – упрощенные логические функции F1, F2, F3) в базисе «ИЛИ-НЕ»:

- 7.1. Запишите соответствующие логические выражения,
7.2. Постройте соответствующие схемы.

Задание 8. По заданной схеме, представленной на рисунке 12.2, получите логические функции (для варианта №1 – это узлы схемы с обозначением f_1, f_2, f_3 ; для варианта №2 – это узлы схемы с обозначением F_1, F_2, F_3)

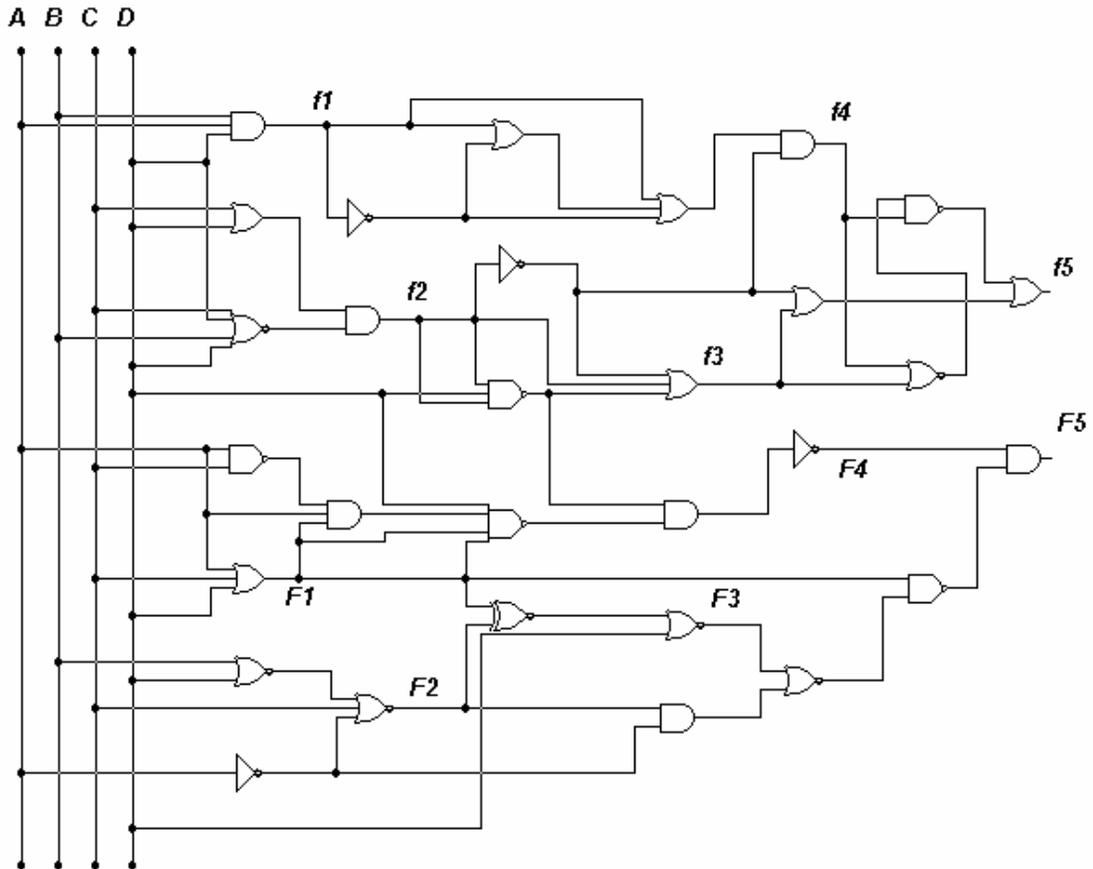


Рисунок 12.2 – Схема к заданию №6

Контрольная работа №2: «Комбинационные устройства средней степени интеграции»

Задание 1. Реализуйте заданные логические функции на мультиплексоре

Вариант №1	Вариант №2
$f_1 = \bar{A} \cdot B \cdot \bar{C} + A \cdot B \cdot \bar{C} + \bar{A} \cdot B \cdot C + A \cdot B \cdot C$	$F_1 = A \cdot B \cdot C + \bar{A} \cdot B \cdot \bar{C} + A \cdot \bar{B} \cdot C + \bar{A} \cdot B \cdot C$
$f_2 = \bar{A} \cdot B \cdot C + A \cdot B \cdot \bar{C} + \bar{A} \cdot B \cdot C + A \cdot B \cdot C$	$F_2 = \bar{A} \cdot B \cdot \bar{C} + \bar{A} \cdot B \cdot C + \bar{A} \cdot B \cdot C + A \cdot B \cdot C$
$f_3 = A \wedge B \wedge (\overline{C \wedge A \vee A \wedge B}) \vee (A \vee B \wedge \bar{C})$	$F_3 = A \wedge C \vee \bar{A} \wedge (\overline{C \vee A}) \vee (C \wedge \bar{A} \wedge B \vee B \wedge \bar{A})$
$f_4 = (A + B\bar{C}) + A(\overline{BC + AC})\bar{C} + \bar{A}\bar{B}\bar{B}$	$F_4 = \overline{AC + BC} + (\bar{A} + C + \bar{AC})(\bar{A} + B)$

- 1.1. запишите таблицу распределения входов и выходов мультиплексора,
1.2. зарисуйте соответствующую схемную реализацию.

Задание 2. Реализуйте заданные логические функции (вариант №1 – логические функции f_1, f_2, f_3, f_4 ; вариант №2 – логические функции F_1, F_2, F_3, F_4) на основе дешифратора и логических элементов следующего типа

- 2.1. логические элементы произвольного типа,
- 2.2. логические элементы «И-НЕ»,
- 2.3. логические элементы «ИЛИ-НЕ»

Задание 3. Разработайте схему на основе дешифратора, формирующую (для варианта №1 – формирующую логическую «1»; для варианта 2 – формирующую логический «0») на указанных выходах Y_i (приведенных в таблице), при определенных комбинациях входных чисел.

Входные числа	Выходы схемы для варианта 1				Выходы схемы для варианта 2			
	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8
Первое	0, hex	2, hex	8, hex	C, hex	4, hex	8, hex	0	3
Второе	1, hex	3, hex	9, hex	D, hex	A, hex	B, hex	7, hex	F, hex

Примечания:

1. считать, что числа на входе задаются в двоичной системе исчисления;
2. для варианта №1 на указанном выходе (Y_1, Y_2, Y_3, Y_4) должна формироваться «1» тогда и только тогда когда на входе схемы одна из перечисленных комбинаций; во всех других случаях на указанном выходе должен устанавливаться «0»;
3. для варианта №2 на указанном выходе (Y_5, Y_6, Y_7, Y_8) должен формироваться «0» тогда и только тогда когда на входе схемы одна из перечисленных комбинаций; во всех других случаях на указанном выходе должна устанавливаться «1».

12.3. ДОМАШНИЕ ЗАДАНИЯ (РГР)

Упражнение 1. Получение логических функций

По заданной таблице истинности (таблица 1.у) получите логическую функцию.

Таблица 1.у – Варианты таблицы истинности

Значения логических переменных (для всех вариантов)				Варианты задания									
				1	2	3	4	5	6	7	8	9	10
A	B	C	D	Значения логической функции (для каждого варианта)									
0	0	0	0	1	0	1	1	0	1	0	0	1	0
0	0	1	0	0	0	1	1	1	1	0	0	1	1
0	1	0	0	0	0	1	1	0	1	0	0	1	0
0	1	1	0	0	1	0	0	0	0	1	0	1	0
1	0	0	0	1	1	1	1	1	0	0	0	1	0
1	0	1	0	0	1	1	0	1	1	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0	1	0	0
1	1	1	0	0	0	1	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0	1	0	1	0	1	1
0	0	1	1	1	0	1	1	1	0	0	0	1	1
0	1	0	1	1	1	0	1	0	1	0	0	0	1
0	1	1	1	0	0	1	1	1	1	1	1	1	1
1	0	0	1	1	0	1	1	0	0	0	1	0	1
1	0	1	1	0	0	1	1	0	0	0	1	0	1
1	1	0	1	1	1	0	1	0	1	1	1	1	1
1	1	1	1	1	1	0	0	1	1	0	1	1	1

Примечание: Значения логических переменных, входящих в вашу логическую функцию заданы столбцами таблицы (*A*, *B*, *C*, *D*), они одинаковы для всех вариантов. Из второй части таблицы необходимо выбрать один столбец (из 10), согласно вашему варианту, который содержит значения вашей логической функции.

Упражнение 2. Построение структурных схем

По заданной логической функции необходимо построить структурную схему.

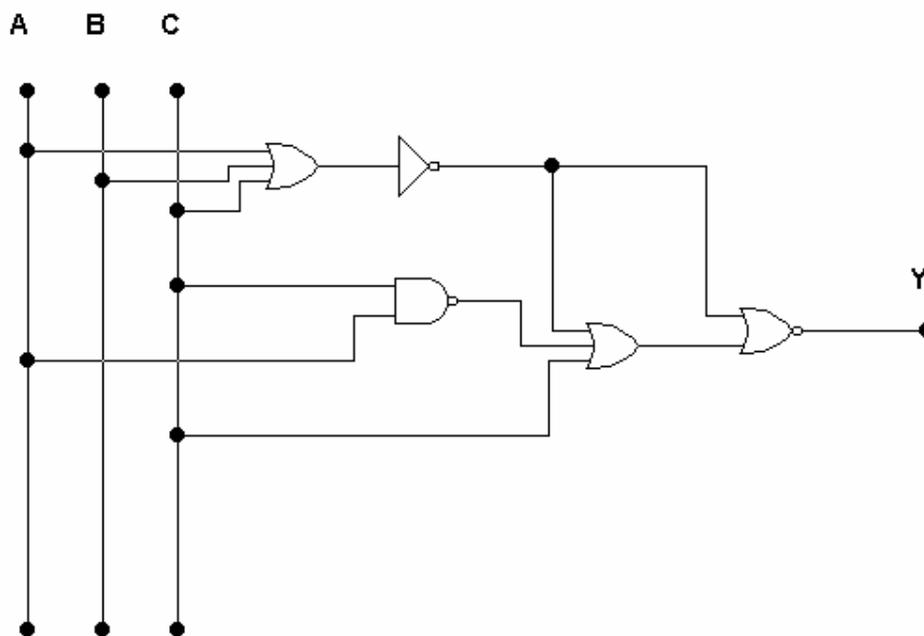
Варианты логических функций, приведены ниже, в таблице 2.у.

Таблица 2.у – Варианты логических функций

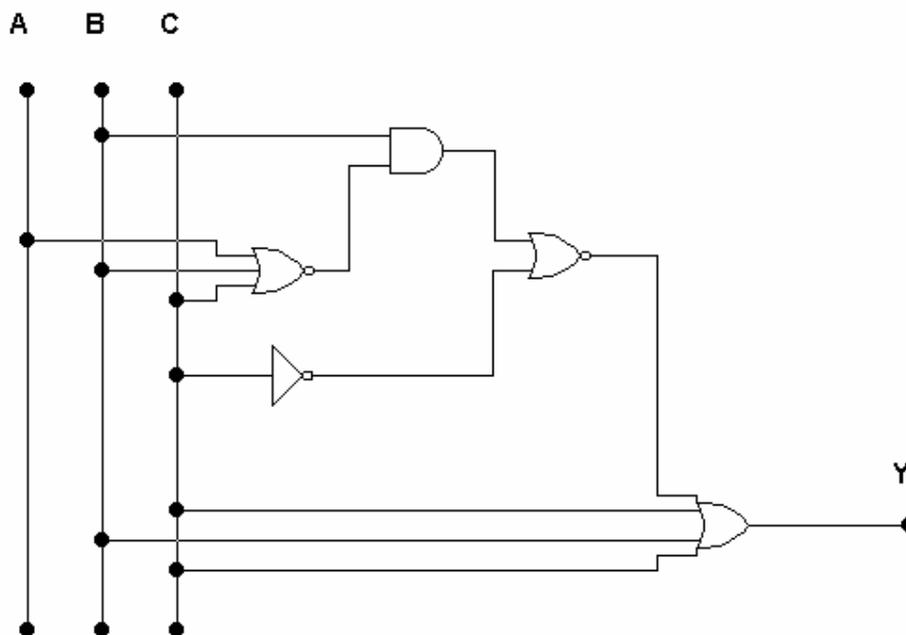
Вариант	Логическая функция
1	$f = \overline{a \vee b} \wedge c \wedge d \vee e$
2	$f = \overline{a \wedge c \vee d \vee e \vee b}$
3	$f = \overline{c} \wedge \overline{a} \vee b \wedge \overline{d} \vee e$
4	$f = \overline{a} \vee \overline{b} \wedge \overline{e} \wedge \overline{d} \vee c$
5	$f = a \vee b \wedge c \vee d \vee e$
6	$f = a \wedge \overline{c} \wedge \overline{d} \vee e \vee \overline{b}$
7	$f = \overline{a \vee b} \vee \overline{c} \vee \overline{d} \wedge \overline{e}$
8	$f = a \wedge b \wedge \overline{d} \vee a \wedge \overline{e} \vee c$
9	$f = \overline{a} \wedge \overline{a} \wedge c \wedge \overline{d} \vee e$
10	$f = \overline{e \vee a} \wedge \overline{a \vee d} \vee e \wedge b$

Упражнение 3. Составление логических функций
 По заданной схеме составьте логическую функцию.

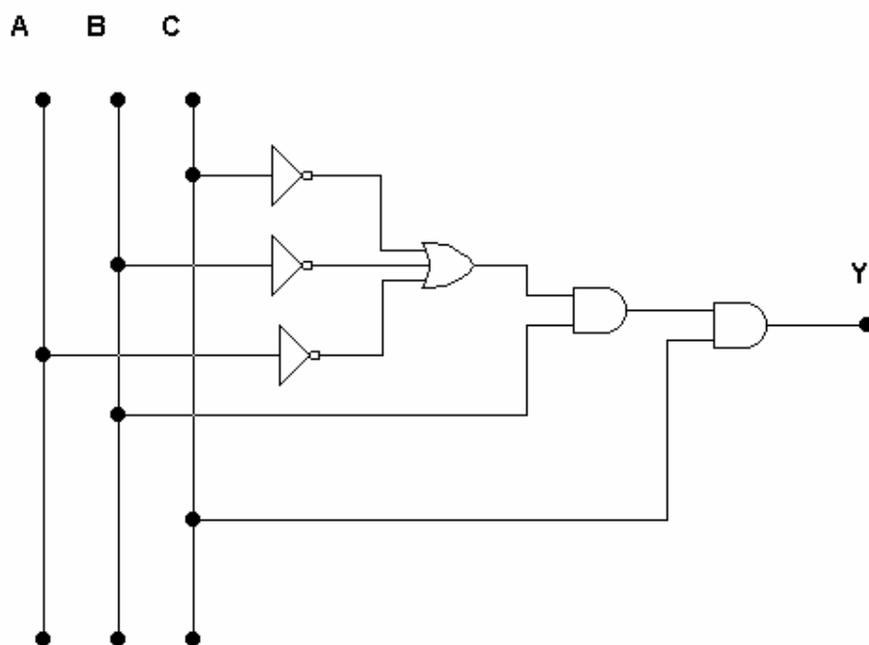
Вариант 1



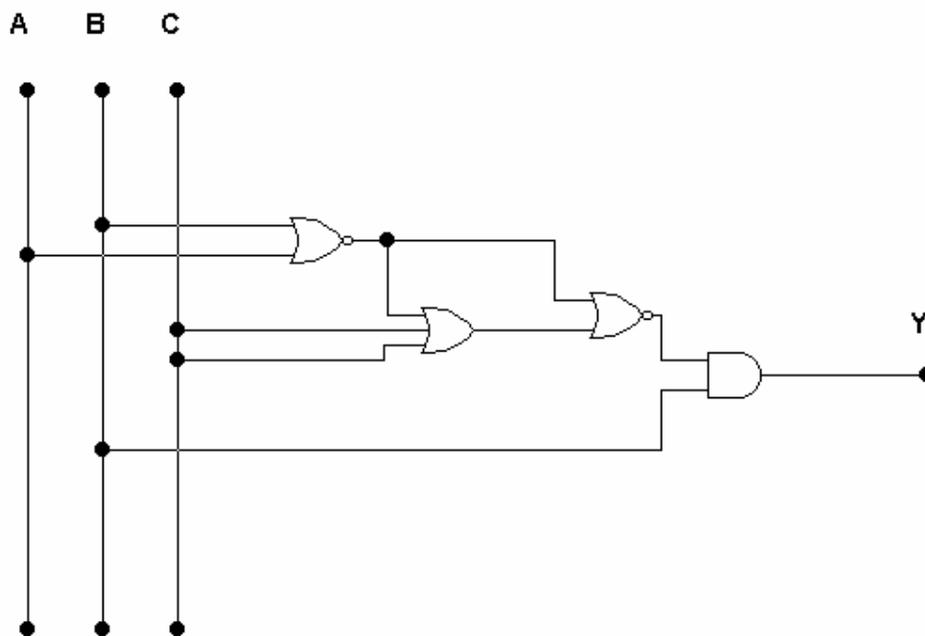
Вариант 2



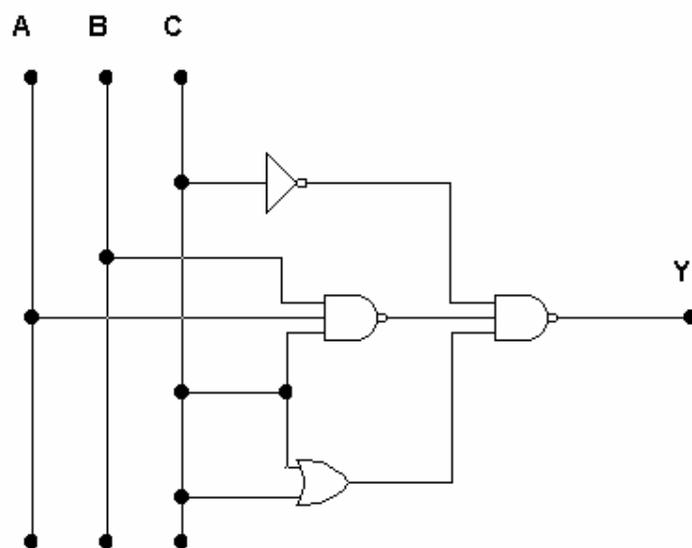
Вариант 5



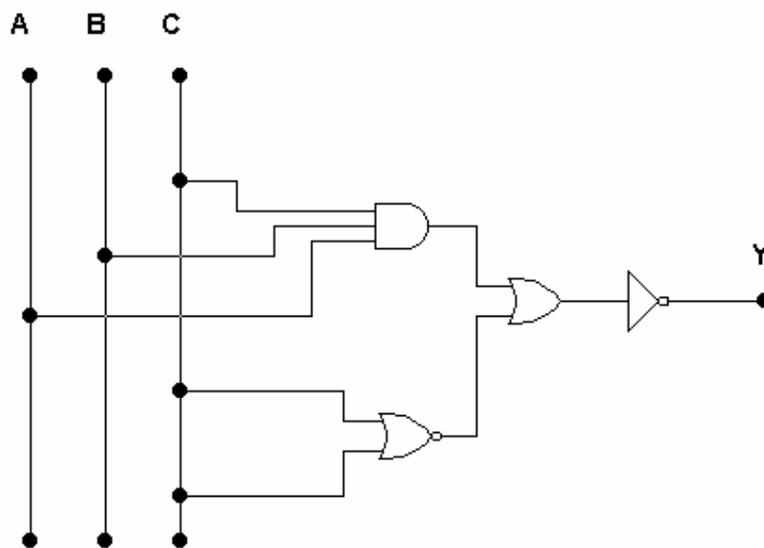
Вариант 6



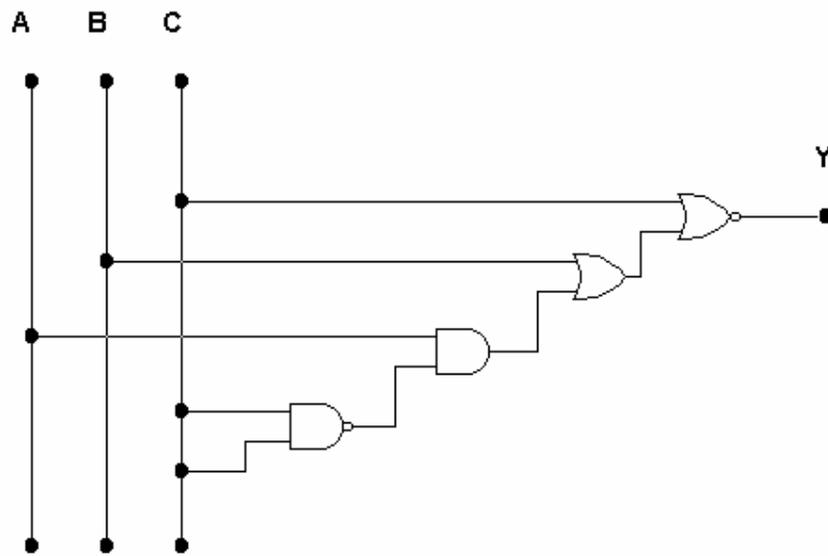
Вариант 7



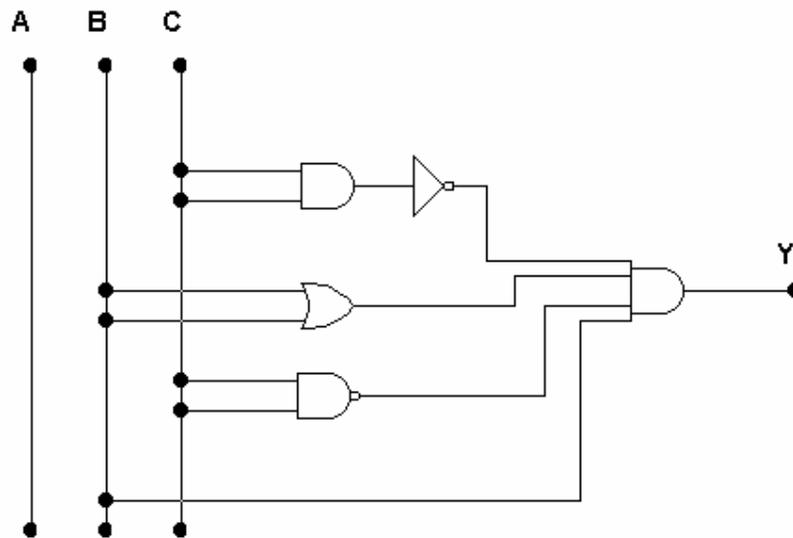
Вариант 8



Вариант 9



Вариант 10



Упражнение 4. Применение дешифраторов.

Сформируйте сигнал выбора определенной микросхемы памяти в микроконтроллере с использованием стандартных дешифраторов.

Микропроцессор, входящий в состав микроконтроллера, имеет 8 разрядную шину адреса, то есть может работать с объемом памяти $2^8=256$ байт (разряды нумеруются $A0 - A7$, начиная с младшего). По каждому адресу хранится информация, объем которой определяется разрядностью шины данный микропроцессора и в данном упражнении не важен. В общем случае вся память делится на определенные блоки, по характеру их использования (например, для ОЗУ, ПЗУ, портов ввода/вывода). Каждый блок памяти имеет начальный и конечный адрес.

Необходимо внутри объема всей доступной памяти сформировать блок, начальный и конечный адреса которых приведены в таблице 4.у. Данный блок будет использоваться для адресации ОЗУ.

Таблица 4.у – Варианты карты памяти для блока ОЗУ

Вариант	Адреса															
	Начальный адрес								Конечный адрес							
	A7	A6	A5	A4	A3	A2	A1	A0	A7	A6	A5	A4	A3	A2	A1	A0
1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
2	0	0	0	1	0	0	0	0	0	0	0	1	1	1	1	1
3	0	0	1	0	0	0	0	0	0	0	1	0	1	1	1	1
4	0	0	1	1	0	0	0	0	0	0	1	1	1	1	1	1
5	0	1	0	0	0	0	0	0	0	1	0	0	1	1	1	1
6	0	1	0	1	0	0	0	0	0	1	0	1	1	1	1	1
7	0	1	1	0	0	0	0	0	0	1	1	0	1	1	1	1
8	0	1	1	1	0	0	0	0	0	1	1	1	1	1	1	1
9	1	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1
10	1	0	0	1	0	0	0	0	1	0	0	1	1	1	1	1

Для выполнения задания необходимо проделать следующее:

проанализировать все адреса, входящие в ваш блок, выделить те разряды шины адреса, значения которых одинаковы для всех адресов вашего блока;

из всех полученных комбинаций выбрать достаточное количество разрядов шины адреса, которые однозначно определяют обращение именно к вашему блоку памяти и не повторяются во всем другом объеме доступной памяти;

полученные разряды шины адреса и будут входами дешифратора, а его выход и есть искомый сигнал выбора определенной микросхемы.

Выберите подходящий дешифратор. Занесите полученную схему в отчет, отметьте номера используемых разрядов шины адреса.

Упражнение 5. Применение мультиплексора

По заданной логической функции постройте схему с использованием мультиплексора.

Таблица 5.у – Варианты логических функций

Вариант	Логическая функция
1	$\bar{C} \cdot B \cdot \bar{A} + \bar{C} \cdot B \cdot A + C \cdot \bar{B} \cdot \bar{A} + C \cdot \bar{B} \cdot A + C \cdot B \cdot \bar{A} + C \cdot B \cdot A$
2	$\bar{C} \cdot \bar{B} \cdot A + \bar{C} \cdot B \cdot \bar{A} + \bar{C} \cdot B \cdot A + C \cdot B \cdot \bar{A} + C \cdot B \cdot A$
3	$\bar{C} \cdot B \cdot \bar{A} + \bar{C} \cdot \bar{B} \cdot A + C \cdot \bar{B} \cdot \bar{A} + C \cdot \bar{B} \cdot A + C \cdot B \cdot \bar{A} + C \cdot B \cdot A$
4	$\bar{C} \cdot \bar{B} \cdot \bar{A} + \bar{C} \cdot \bar{B} \cdot A + C \cdot \bar{B} \cdot \bar{A} + C \cdot \bar{B} \cdot A + C \cdot B \cdot \bar{A} + C \cdot B \cdot A$
5	$\bar{C} \cdot \bar{B} \cdot \bar{A} + \bar{C} \cdot \bar{B} \cdot A + \bar{C} \cdot B \cdot \bar{A} + \bar{C} \cdot B \cdot A + C \cdot B \cdot \bar{A} + C \cdot B \cdot A$
6	$\bar{C} \cdot \bar{B} \cdot \bar{A} + \bar{C} \cdot \bar{B} \cdot A + \bar{C} \cdot B \cdot \bar{A} + C \cdot \bar{B} \cdot \bar{A} + C \cdot B \cdot \bar{A}$
7	$\bar{C} \cdot \bar{B} \cdot \bar{A} + \bar{C} \cdot \bar{B} \cdot A + \bar{C} \cdot B \cdot \bar{A} + \bar{C} \cdot B \cdot A + C \cdot \bar{B} \cdot \bar{A} + C \cdot \bar{B} \cdot A$
8	$\bar{C} \cdot \bar{B} \cdot \bar{A} + \bar{C} \cdot \bar{B} \cdot A + C \cdot \bar{B} \cdot \bar{A} + C \cdot \bar{B} \cdot A + C \cdot B \cdot \bar{A}$
9	$\bar{C} \cdot \bar{B} \cdot \bar{A} + \bar{C} \cdot \bar{B} \cdot A + C \cdot \bar{B} \cdot \bar{A} + C \cdot \bar{B} \cdot A + C \cdot B \cdot \bar{A} + C \cdot B \cdot A$
10	$\bar{C} \cdot \bar{B} \cdot A + \bar{C} \cdot B \cdot \bar{A} + C \cdot \bar{B} \cdot \bar{A} + C \cdot \bar{B} \cdot A + C \cdot B \cdot \bar{A}$

Упражнение 6. Построение временных диаграмм

По заданной таблице истинности некоторой логической функции постройте временную диаграмму.

Считайте, что переключение в новое состояние схемы должно происходить по отрицательному фронту импульса С.

Варианты таблицы истинности приведены в таблице 6.у.

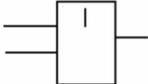
Таблица 6.у – Варианты задания логической функции

Значения логических переменных (для всех вариантов)			Варианты задания									
			1	2	3	4	5	6	7	8	9	10
X	Y	Z	Значения логической функции (для каждого варианта)									
0	0	0	0	0	1	1	0	1	0	0	1	0
0	0	1	0	1	0	0	0	0	1	0	1	0
0	1	0	0	1	1	0	1	1	0	0	0	0
0	1	1	1	0	1	1	0	0	0	1	0	1
1	0	0	0	1	0	1	0	1	1	1	1	1
1	0	1	1	1	0	1	0	1	1	1	1	1
1	1	0	0	0	1	1	0	0	0	1	0	1
1	1	1	0	0	1	0	0	1	0	0	0	0

13. ФОНД ТЕСТОВЫХ И КОНТРОЛЬНЫХ ЗАДАНИЙ ДЛЯ ОЦЕНКИ КАЧЕСТВА ЗНАНИЙ ПО ДИСЦИПЛИНЕ

13.1. Тест по главе 1

1. Одним из способов описания цифрового устройства является...	а) Логическое выражение; б) Логическое тождество; в) Логическая функция; г) Правильного ответа нет;
2. Схема, представляющая собой управляемый переключатель, который подключает к выходу один из входов данных, называется:	а) Шифратор; б) Мультиплексор; в) Дешифратор; г) Сумматор;
3. В двоичной системе исчисления за единицу объема принят...	а) Гбайт; б) Бит; в) Байт; г) Мбайт;
4. Информация после какого-то свершившегося факта или действия называется....	а) Апостериорная неопределенность; б) Синтаксическая адекватность; в) Априорная неопределенность; г) Энтропия;
5. Под процессом автоматизированной обработки информации понимается...	а) Получение; б) Хранение; в) Преобразование; г) Передача;
6. Дисциплина, занимающаяся изучением работы устройств, работающих по принципу включено-выключено называется...	а) Геометрией; б) Булевой алгеброй; в) Математикой; г) Правильных ответов нет;
7. К комбинационным устройствам относятся:	а) Транзистор; б) Шифратор; в) Резистор; г) Сумматор;
8. Соответствие содержания и смысла называется...	а) Синтаксическая адекватность; б) Прагматическая адекватность; в) Семантическая адекватность; г) Правильных ответов нет;

<p>9. Объем работ выполняемых в единицу времени называется...</p>	<p>а) Быстродействие; б) Достоверность; в) Точность; г) Производительность;</p>
<p>10. Простейшее устройство ВМ, выполняющая одну операцию над входными переменными, называется...</p>	<p>а) Узел; б) Элемент; в) Блок; г) Правильных ответов нет;</p>
<p>11. Для случая, когда все состояния системы равновероятны, энтропия определяется по формуле:</p>	<p>а) $H(\alpha) = \log N$ б) $H(\alpha) = -\sum P_i \log N$ в) $I\beta(\alpha) = H(\alpha)$ г) $N = m \cdot n$</p>
<p>12. Какой из триггеров делит частоту входного сигнала пополам?</p>	<p>а) RS-триггер; б) Т-триггер; в) D-триггер; г) JK-триггер;</p>
<p>13. Схемное представление соответствует операции:</p> 	<p>а) Конъюнкция; б) Дизъюнкция; в) Инверсия; г) Эквивалентность;</p>
<p>14. Показателем качества является...</p>	<p>а) Последовательность; б) Точность; в) Адекватность; г) Достаточность;</p>
<p>15. Управляемый переключатель, имеющий один выход и несколько входов называется...</p>	<p>а) Диод; б) Дешифратор; в) Резистор; г) Мультиплексор;</p>
<p>16. Какой комбинационной схемы представлено схемное представление?</p> 	<p>а) D-триггер; б) Т-триггер; в) RS-триггер; г) JK-триггер;</p>

<p>17. По назначению ВМ бывают:</p>	<p>а) Специализированные; б) Функциональные; в) Структурные; г) Универсальные;</p>																		
<p>18. Таблица истинности какого элемента представлена на рисунке?</p> <table border="1" data-bbox="326 464 732 730"> <thead> <tr> <th colspan="2">Входы</th> <th>Выходы</th> </tr> <tr> <th>А</th> <th>В</th> <th>У</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	Входы		Выходы	А	В	У	0	0	0	0	1	0	1	0	0	1	1	1	<p>а) Конъюнкция; б) Инверсия; в) Эквивалентность; г) Дизъюнкция;</p>
Входы		Выходы																	
А	В	У																	
0	0	0																	
0	1	0																	
1	0	0																	
1	1	1																	
<p>19. Как называется соответствие цели и содержания?</p>	<p>а) Синтаксическая адекватность; б) Прагматическая адекватность; в) Семантическая адекватность; г) Правильных ответов нет;</p>																		
<p>20. Переменные, рассматриваемые в алгебре логики, принимают значение:</p>	<p>а) '2' б) '0' в) '5' г) '1'</p>																		
<p>21. . Какие из указанных входов имеются у триггера?</p>	<p>а) Информационные б) Управляющие в) Разрешения г) Установочные</p>																		
<p>22. Абстрактная модель, описывающая функциональные возможности ВМ, и предоставляемые ей услуги называется...</p>	<p>а) Структурная организация ВМ; б) Техническая организация ВМ; в) Функциональная организация ВМ; г) Правильных ответов нет;</p>																		

<p>23.Схемное представление какой функции представлено на рисунке?</p> 	<p>а) И; б) ИЛИ; в) И-НЕ; г) ИЛИ-НЕ;</p>
<p>24. Как называется комплекс программно-технических средств, предназначенных для обработки информации в процессе решения информационных и вычислительных задач?</p>	<p>а) Персональный компьютер; б) Микропроцессор; в) Комбинационная схема; г) Вычислительная машина;</p>
<p>25. Устройство, предназначенное для синхронизации всей системы, называется...</p>	<p>а) Процессор; б) Внешнее запоминающее устройство; в) Таймер; г) Тактовый генератор;</p>
<p>26. Как называется выделение в структуре ВМ, достаточно автономных, функционально и конструктивно независимых устройств?</p>	<p>а) Модульность; б) Точность; в) Доступность; г) Адекватность;</p>
<p>27. Шифратор - комбинационная схема, которая имеет...</p>	<p>а) 2^n выходов; б) n входов; в) 2 выхода; г) 2^n входов;</p>
<p>28. Что выполняет устройство управления УУ?</p>	<p>а) Выборку данных; б) Выборку команд; в) Дешифрирование команд г) Правильных ответов нет;</p>
<p>29. Адекватность – это...</p>	<p>а) Соответствие содержания и смысла; б) Соответствие содержания и цели; в) соответствие содержания информации образу исходного объекта; г) Соответствие формально-структурных показателей;</p>

<p>30. Связанные между собой по принципу работы АЛУ и УУ объединены единым устройством -...</p>	<p>а) Счетчик; б) Дешифратор; в) Вычислительная машина; г) Микропроцессор;</p>
<p>31. Фамилией, какого ученого названо данное правило? $y = \overline{x_1 \cdot x_2} = \overline{x_1} + \overline{x_2}$ $y = \overline{x_1 + x_2} = \overline{x_1} \cdot \overline{x_2}$</p>	<p>а) Глушкова б) Фон Неймана в) Де Моргана г) Карно</p>
<p>32. Структурой называется...</p>	<p>а) Элементарная ячейка памяти, представленная для внешнего хранения промежуточных результатов. б) Совокупность элементов и их связей. в) Контролер прямого доступа к памяти. г) Согласованность параметров элементарных сигналов.</p>
<p>33. В какой структуре множество процессорных элементов объединены с помощью коммутатора?</p>	<p>а) VLIW архитектура. б) RISC архитектура. в) CISC архитектура. г) SIMD архитектура.</p>
<p>34. К какому поколению относятся ВМ с жесткой структурой устройства управления?</p>	<p>а) 1-му. б) 2-му в) 3-му. г) 4-му.</p>
<p>35.Какая из предложенных подсистем осуществляет инициализацию, тестирование, отладку и контроль за эффективностью работы всей системы?</p>	<p>а) Подсистема ввода-вывода. б) Подсистема повышения производительности. в) Подсистема памяти. г) Подсистема управления и обслуживания.</p>
<p>36. Структурной организацией ВМ называется...</p>	<p>а) Абстрактная модель, описывающая функциональные возможности ВМ и представленные ей услуги. б) Физическая модель, устанавливающая состав, порядок и принципы взаимодействия основных функциональных частей ВМ. в) Физическая модель, описывающая возможности ВМ и представляющая ей услуги. г) Правильных ответов нет.</p>

37. Основой ВМ в 70-е XX века являлись ...	а) Полупроводниковые приборы. б) Интегральные схемы. в) Микропроцессоры. г) Вакуумные лампы.
38. Какие ЭВМ разрабатывались на основе микрокомпонентов?	а) Супер-ЭВМ. б) Большие ЭВМ. в) Мини ЭВМ. г) Микро ЭВМ.
39. Важнейшим свойством информации является...	а) Актуальность; б) Простота; в) Адекватность; г) Правильного ответа нет;
40. Предписание, определяющее содержание действий, называется...	а) Адрес; б) Архитектура; в) Программа; г) Команда;
41. На каком этапе VLIW архитектуры команды группируются в пакеты, содержащее, которых соответствует структуре процессора?	а) 3; б) 1; в) 4; г) 2;
42. Функция, представляющая собой зависимость выхода системы от входа $y = f(x)$ называется...	а) Логическим элементом; б) Логическим тождеством; в) Логическим выражением; г) логической функцией;
43. Как называется устройство, которое подключает один вход к нескольким выходам?	а) Счетчик; б) Шифратор; в) Регистр; г) Правильного ответа нет;
44. К преимуществам SIMD архитектуры относятся:	а) Экономичность; б) Выполнение одной или нескольких команд одновременно; в) Большое число внутренних регистров; г) Высокая производительность;

<p>45. В каких системах человек присутствует только на этапе проектирования?</p>	<p>а) Автоматизированные системы; б) Автоматические системы; в) Функциональные системы; г) Подсистема управления;</p>																		
<p>46. Схемное представление какой функции представлено на рисунке?</p> 	<p>а) Эквивалентность; б) Конъюнкция; в) Дизъюнкция; г) Инверсия;</p>																		
<p>47. Таблица истинности какого элемента представлена на рисунке?</p> <table border="1" data-bbox="345 743 730 1014"> <thead> <tr> <th colspan="2">Входы</th> <th>Выход</th> </tr> <tr> <th>А</th> <th>В</th> <th>У</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	Входы		Выход	А	В	У	0	0	1	0	1	1	1	0	1	1	1	0	<p>а) ИЛИ; б) И; в) И-НЕ; г) ИЛИ-НЕ;</p>
Входы		Выход																	
А	В	У																	
0	0	1																	
0	1	1																	
1	0	1																	
1	1	0																	
<p>48. Арифметическое логическое устройство содержит:</p>	<p>а) Триггеры; б) Мультиплексоры; в) Шифраторы; г) Логические преобразователи;</p>																		
<p>49. В каком типе архитектуры ВМ оценивается быстрое действие, производительность, надежность.</p>	<p>а) Программный; б) Функциональный; в) Технический; г) Правильных ответов нет;</p>																		
<p>50. Какая из предложенных характеристик позволяет оценить правильность восприятия информации?</p>	<p>а) Надежность; б) Достоверность; в) Точность; г) Производительность;</p>																		

13.2. Тест по главе 2

1. Элементарная ячейка памяти, предназначенная для временного хранения промежуточных результатов.	a) регистр b) выборка c) чипсет d) свопинг
2. Предназначен для выполнения полного функционального набора арифметических и логических операций. Как правило в него входят: РОН, АЛУ, аккумулятор, буферный регистр и т.д.	a) интерфейсный блок b) блок управления c) операционный блок d) блок заполнения
3. В большинстве реальных моделях МП один из регистров выделяется в качестве главного и называется	a) БФУС b) РОН c) аккумулятор d) регистр команд
4. В состав интерфейсного блока входит	a) счётчик команд b) регистр сдвигатель c) буферный регистр данных d) всё выше перечисленное
5. Регистр предназначенный для хранения результатов	a) буферный регистр b) регистр признаков c) регистр команд d) регистр сдвигатель
6. Предназначен для адресации внутри стека. При выполнении одной операции увеличивается на 1-цу тем самым обеспечивает выполнение следующей команды	a) указатель стека b) счётчик команд c) БФУС d) регистр команд
7. При каком виде адресации сам код команды подразумевает работу с неким адресом и этот код не указывается	a) прямая адресация b) неявная адресация c) косвенная адресация d) индексная адресация
8. Предназначен для организации взаимодействия МП с внешними устройствами, с устройством ввода-вывода, а также для организации обмена между ОП и операционным блоком	a) интерфейсный блок b) блок управления c) операционный блок d) блок заполнения
9. Применяется для временного хранения данных, организация прерывателя, вызова процедур и т.д.	a) сектор b) стек c) очередь d) блок
10. Обеспечивает внутренний обмен информации между регистрами МП, также обеспечивает обмен данных между устройствами ввода-вывода и МП	a) команды пересылки данных b) команды передачи управления c) команды работы со стеком d) команды поразрядного двоичного сдвига
11. При каком виде адресации адрес содержится в коде команды и обычно следует за кодом операции	a) непосредственная адресация b) прямая адресация c) косвенная адресация d) неявная адресация
12. Одна из фаз при выполнении любой команды, которая обеспечивает считывание команд из памяти и пересылку её в МП	a) выборка b) декодирование c) выполнение d) реализация
13. Какой вид адресации позволяет задавать фикс-	a) косвенная адресация

сированные значения непосредственно в адресной части	b) неявная адресация c) непосредственная адресация d) индексная адресация
14. При каком виде адресации команд адрес команд содержится в коде операции. При считывании очередной команды в СК автоматически загружается адрес следующей команды	a) относительная адресация b) абсолютная адресация c) косвенная адресация d) прямая адресация
15. При увеличении ёмкости памяти её быстродействие	a) увеличивается b) уменьшается c) не изменяется d) безразлично
16. Самая быстродействующая память	a) регистровая память b) сверхоперативная память c) управляемая память d) буферная
17. Какой вид адресации используется если необходимо считать список из ячеек памяти расположенный не подряд а с некоторым шагом	a) прямая адресация b) неявная адресация c) косвенная адресация d) индексная адресация
18. Тип памяти реализованной с помощью некоего запоминающего устройства размещённого между основной памятью и процессором, предназначенная для сокращения обращений МП к основной памяти	a) управляемая память b) буферная c) регистровая память d) сверхоперативная память
19. Базовый тип оперативной памяти которая содержит ячейки памяти работающие по принципу конденсатора, наличие или отсутствие заряда	a) SRAM b) DRAM c) SDAM d) FRAM
20. Базовый тип оперативной памяти признаком хранения информации которой является типа открыто, закрыто. Реализуется на основе транзисторных схем	a) FRAM b) SRAM c) DRAM d) SDAM
21. Тип распределения памяти при котором виртуальное адресное пространство разбивается на сегменты, величина которых определяется программистом либо системой	a) страничное распределения b) сегментное распределения c) странично-сегментное распределения d) свопинг
22. Тип распределения памяти при котором некоторые задачи находятся в режиме ожидания, целиком отгружаются на жёсткий диск и в случае необходимости могут быть снова загружены памятью	a) странично-сегментное распределения b) сегментное распределения c) свопинг d) страничное распределения
23. Ресурс обладающий гораздо большим объёмом чем ОП, но для пользователя представленная как единое целое	a) регистровая память b) виртуальная память c) буферная d) управляемая память
24. При каком виде адресации адрес указываемый в команде является указателем ячейки содержащий исполнительный адрес, фактически указывается адрес адреса	a) относительная адресация b) абсолютная адресация c) косвенная адресация d) прямая адресация

25. Принцип обмена информацией между периферийными устройствами и МП. Предполагает что одно из устройств является ведущим, а второе ведомым	<ul style="list-style-type: none"> a) принцип подчинения b) принцип квитиования c) принцип унификации характеристик d) принцип замены
26. Программно управляемая передача которая применяется при взаимодействии с быстродействующими устройствами для обмена с которыми не требуется дополнительный сигнал синхронизации	<ul style="list-style-type: none"> a) синхронная передача b) прямая передача c) асинхронная передача d) непрямая передача
27. Способ обмена данными между периферийными устройствами и вычислительным ядром системы при котором для операции ввода вывода используются специальные сигналы	<ul style="list-style-type: none"> a) Программно управляемая передача b) Передача инфор. с прерыванием c) Передача инфор. в режиме прямого доступа к памяти
28. Центральное устройство предназначенное для управления работой и выполнения арифметических, логических и др. операций	<ul style="list-style-type: none"> a) микропроцессор b) оперативная память c) системная шина d) DART
29. Обеспечивает сопряжение и связь всех устройств	<ul style="list-style-type: none"> a) системная шина b) прямая адресация c) RATS d) кэш-память
30. Предназначена для хранения и оперативного обмена информацией. Содержит ОЗУ и ПЗУ	<ul style="list-style-type: none"> a) оперативная память b) внешнее запоминающее устройство c) основная память d) кэш-память
31. Жёсткий диск, флэшка, CD, дискета и т.д.	<ul style="list-style-type: none"> a) оперативная память b) кэш-память c) основная память d) внешнее запоминающее устройство
32. Тип системной платы определяет	<ul style="list-style-type: none"> a) основная память b) микропроцессор c) оперативная память d) периферийные устройства
33. Шина расширения, имеет 8-ми разрядную шину данных, 20-ти разрядную шину адреса, может работать на частоте 4,77 МГц	<ul style="list-style-type: none"> a) ISA b) PC/XT c) PC/AT d) AGP
34. Шина расширения, имеет 32-х разрядную шину данных и шину адреса, создана в 1989г. Работает на частоте 8-10 МГц. Имеет возможность подключать до 16 внешних устройств	<ul style="list-style-type: none"> a) PC/XT b) IESA c) VLB d) SCSJ
35. Локальная шина, имеет 32-х разрядную шину данных и шину адреса ,допускает подключение 10 внешних устройств. Тактовая частота 33 МГц. Может работать в 64 битных системах	<ul style="list-style-type: none"> a) PCI b) VLB c) PC/AT d) AGP
36. Тип памяти построенная на основе полупроводников, своеобразных конденсаторах	<ul style="list-style-type: none"> a) статическая память b) постоянная память c) динамическая память d) внешняя память

37. Используется для работы с жёстким диском	a) SATA b) RS-232 c) ATA d) IEEE 1284
38. Универсальная периферийная шина с возможностью подключения 126 устройств. Передача информации 12 Мбит/с	a) IEEE 1394 b) PCMCIA c) VLB d) USB
39. Модуль оперативной памяти, работает на частоте 600-800 МГц, имеет пропускную способность 1,6 Гбайт/с, время обращения 5 нс	a) DIM b) RIM c) SIM d) DIP
40. Тип оперативной памяти, пропускная способность при шине 100 МГц 1,6 Гбайт/с, выпускается в конструкции DIM	a) FPMDRAM b) RAMEDO c) DRAM d) DDR SDRAM
41. Тип оперативной памяти, может работать на частоте 800 МГц, позволяет использовать технологию двухканального обмена	a) RAMEDO b) DRAM c) DR DRAM d) FPMDRAM
42. На какой фазе проводится процедура тестового контроля	a) на всех b) выборочно c) не на одной d) возможны все варианты
43. Если причиной неисправности являются ошибки, допущенные при проектировании, некорректный монтаж, нарушение режимов работы то неисправность называется	a) физическая неисправность b) субъективная неисправность c) объективная неисправность d) семеричная неисправность
44. Процесс обнаружения ошибки и определение источника его появления	a) диагностика неисправности b) отладка c) выкладка d) проектирование
45. В процессе отладки неисправности устраняются	a) полностью b) не устраняются вовсе c) возможен их пропуск d) устраняется первая и процесс заканчивается
46. Что должно: 1) управлять исполнением программы, останавливать и т.д. 2) собирать информацию о ходе выполнения программы 3) обеспечивать обмен информации между программой и ВМ 4) моделировать работу отдельных элементов системы	a) средства отладки программ b) диагностика неисправности c) тестовый сегмент d) средства фиксации
47. Программное обеспечение МП разрабатывается на	a) иностранных языках b) алгоритмических языках c) интернациональных языках d) возможны все варианты ответов
48. Исходную программу на ассемблере составляют с помощью	a) Microsoft Word b) блокнота

	c) Bred d) любого текстового редактора
49. Содержит исходную программу, программу в машинных кодах, а также обнаруженные ошибки с указанием страницы где они обнаружены	a) исполнительный файл b) листинг c) объектная модель d) транслятор
50. Сколько существует основных приёмов комплексной отладки микропроцессорных систем	a) 2 b) 7 c) 1 d) 5

13.3. Тест по главе 3

1. Микроконтроллерная система, предназначенная для решения конкретных технических задач.	1. Персональный компьютер 2. Микропроцессор 3. Простейшая вычислительная система 4. Объект управления
2. Некий управляемый совокупностью используемых устройств процесс.	1. Объект управления 2. Вычислительная система 3. Микропроцессор 4. Устройство сопряжения с объектом
3. Это устройство обеспечивает согласование сигналов, как управления, так и информации между объектами управления и микропроцессорными системами.	1. Объект управления 2. Пульт управления 3. Микропроцессор 4. Устройство сопряжения с объектом
4. Это устройство представляет оператору формировать параметры процесса управления и при необходимости вносить коррективы.	1. Объект управления 2. Пульт управления 3. Цифровой регулятор 4. Устройство сопряжения с объектом
5. Это устройство получает информацию об объекте управления от датчиков.	1. Объект управления 2. Пульт управления 3. Цифровой регулятор 4. Устройство сопряжения с объектом
6. Нормальный процесс работы системы, который проходит после её создания.	1. Частота 2. Функционирование 3. Быстродействие 4. Производительность
7. Система функционирующая без участия человека.	1. Автоматическая 2. Механическая 3. Автоматизированная 4. Независимая
8. Система функционирующая под воздействием человека.	1. Автоматическая 2. Механическая 3. Автоматизированная 4. Независимая
9. Как система управления с микроконтроллером обрабатывает информацию во време-	1. Непрерывно 2. Дискретно

ни.	<ol style="list-style-type: none"> 3. По закону \sin 4. По закону \cos
10. Структура трехуровневого аппаратно-программного управляющего комплекса (от низшего к высшему).	<ol style="list-style-type: none"> 1. Микроконтроллер – пульт управления - цифровой регулятор 2. Пульт управления – цифровой регулятор – микроконтроллер 3. Цифровой регулятор – микроконтроллер - пульт управления 4. Микроконтроллер – цифровой регулятор – пульт управления
11. Система управления пространственно приближенная к датчикам и использующая устройства.	<ol style="list-style-type: none"> 1. Локальная 2. Встраиваемая (централизованная) 3. Независимая 4. Зависимая
12. Для обеспечения надежности работы распределённых систем управления функции управления техническим процессом возлагаются на	<ol style="list-style-type: none"> 1. Вычислительные системы высшего уровня 2. Микроконтроллеры низшего уровня 3. Пульт управления 4. Цифровой регулятор
13. Режим в котором система должна обеспечивать формирование управляющих сигналов.	<ol style="list-style-type: none"> 1. Реального времени 2. Виртуального времени 3. Функционирования 4. Отказа
14. Программируемое однокристальное вычислительное устройство с встроенным набором средств ввода и вывода, применяемое для решения задач управления и первичной обработки данных технических системах.	<ol style="list-style-type: none"> 1. Микроконтроллер 2. Пульт управления 3. Цифровой регулятор 4. Устройство сопряжения с объектом
15. Какой из вариантов не является методом увеличения производительности вычислительных систем параллельной обработки данных.	<ol style="list-style-type: none"> 1. Совершенствование элементной базы 2. Повышение входного напряжения 3. Использование методов параллельной обработки (структурный) 4. Математический метод
16. Какой из вариантов не является особым типом ошибок вычислительных систем параллельной обработки.	<ol style="list-style-type: none"> 1. Взаимные блокировки процессов (дэдлоки) 2. Нехватка напряжения (вольтлоки) 3. Не использование альтернатив (ливлоки) 4. Невозможность получить ресурс (голодание)
17. Эта обработка осуществляется на нескольких параллельно работающих устройствах, причём каждый элемент осуществляет обработку соответствующей порции данных от начала и до конца (суперскалярная архитектура).	<ol style="list-style-type: none"> 1. Многостадийная обработка 2. Многоэлементная обработка 3. Многомашинная обработка 4. Повременная
18. В случае этой обработки процесс обработки данных разбивается на несколько	<ol style="list-style-type: none"> 1. Многостадийная обработка 2. Многоэлементная обработка

фаз, которые выполняются последовательно.	<ul style="list-style-type: none"> 3. Многомашина обработка 4. Повременная
19. В чем сходство MISD и SISD систем.	<ul style="list-style-type: none"> 1. Имеют одиночный поток данных 2. Имеют одиночный поток команд 3. Имеют множественный поток команд 4. Имеют множественный поток данных
20. В чем сходство MIMD и SIMD систем.	<ul style="list-style-type: none"> 1. Имеют одиночный поток данных 2. Имеют одиночный поток команд 3. Имеют множественный поток команд 4. Имеют множественный поток данных
21. Какие компьютеры представляют собой регулярную структуру, состоящую из цепочки последовательно соединенных процессоров, образующих процессорный конвейер.	<ul style="list-style-type: none"> 1. MISD 2. SISD 3. MIMD 4. SIMD
22. В каких компьютерах процесс обработки данных делится на последовательно выполняемые этапы, каждый этап выполняется на отдельном процессоре. А единый поток данных поступает на вход всего процессорного конвейера.	<ul style="list-style-type: none"> 1. MISD 2. SISD 3. MIMD 4. SIMD
23. В каком классе вычислительных машин процессоры обмениваются информацией через единую память, при этом возможны конфликты процессоров.	<ul style="list-style-type: none"> 1. Многопроцессорные вычислительные системы 2. Однопроцессорные вычислительные системы 3. Персональный компьютер 4. Многомашина вычислительные системы
24. В каком классе многопроцессорных вычислительных систем информация между модулями передается в режиме разделения времени, то есть на шине присутствует в определённый момент времени информация от одного модуля.	<ul style="list-style-type: none"> 1. С общей шиной 2. С многоходовой памятью 3. Модульной 4. Многомашина
25. В каком классе многопроцессорных вычислительных систем память имеет персональные выходы и управляющие схемы для подключения каждого микропроцессора.	<ul style="list-style-type: none"> 1. С общей шиной 2. С многоходовой памятью 3. Модульной 4. Многомашина
26. В системах данного типа реализуется асинхронный вычислительный процесс, а каждый микропроцессор выполняет свою собственную программу или её участок. В таких системах происходит постоянное распараллеливание вычислений.	<ul style="list-style-type: none"> 1. Многопроцессорные вычислительные системы 2. Однопроцессорные вычислительные системы 3. Персональный компьютер 4. Многомашина вычислительные системы
27. Такие системы относятся к системам со	<ul style="list-style-type: none"> 1. Сетевые

слабой связью состоят из нескольких компьютеров, объединённых на основе сетевых средств.	<ol style="list-style-type: none"> 2. Системы массового параллелизма 3. Одноядерные 4. Многомашинные комплексы
28. Для чего используется многомашинная связь на уровне внешних устройств.	<ol style="list-style-type: none"> 1. Для увеличения входов в памяти 2. Для организации общего объема адресного пространства для внешних запоминающих устройств 3. Для уменьшения объема внешних запоминающих устройств 4. Для организации общего объема данных во внутренних запоминающих устройствах
29. При помощи какого устройства осуществляется взаимодействие вычислительных систем на уровне канал - канал	<ol style="list-style-type: none"> 1. Адаптер 2. Внешняя память 3. Внутренняя память 4. Датчик
30. Для этой системы характерно то, что большинство узлов имеют логически и физически распределённую между процессорами память, а каждый узел выполняет отдельный процесс. В узле хранится результат промежуточных вычислений.	<ol style="list-style-type: none"> 1. Сетевые 2. Системы массового параллелизма 3. Одноядерные 4. Многомашинные комплексы
31. Как в системах массового параллелизма происходит передача данных между узлами.	<ol style="list-style-type: none"> 1. По факту готовности некоего процесса 2. Перед началом выполнения центральной программы 3. Под управлением центральной программы 4. По истечении некоего времени
32. Сеть обмена и распределённой обработки информации, образуемая множеством взаимосвязанных абонентских систем и средствами связи.	<ol style="list-style-type: none"> 1. Вычислительная сеть 2. Телекоммуникационная вычислительная сеть 3. Производственная сеть 4. Сеть микропроцессоров
33. Что не включает в себя телекоммуникационная система.	<ol style="list-style-type: none"> 1. Бытовая техника 2. Физическая среда передачи 3. Аппаратные средства, обеспечивающие взаимодействие абонентских станций 4. Программные средства, обеспечивающие взаимодействие абонентских станций
34. Что не является важнейшей функцией телекоммуникационной системы.	<ol style="list-style-type: none"> 1. Синхронизация взаимодействия абонентских систем при обмене информацией 2. Коммутация соединений 3. Маршрутизация сообщений 4. Создание положительной мотивации
35. Какая последовательность действий, называемая протоколом, верна.	<ol style="list-style-type: none"> 1. Запрос данных – подтверждение запроса – ответ – подтверждение ответа

	<p>та</p> <ol style="list-style-type: none"> 2. Запрос данных – ответ – подтверждение запроса - подтверждение ответа 3. Ответ - запрос данных - подтверждение запроса – подтверждение ответа 4. Подтверждение ответа – подтверждение запроса – ответ – запрос
36. Эти сети, объединяющие абонентов района, города, области. Удаленность абонентов обычно составляет десятки – сотни километров.	<ol style="list-style-type: none"> 1. Глобальные сети (WAN) 2. Региональные сети (MAN) 3. Локальные сети (LAN) 4. Системные сети (SAN)
37. Это высоко производительные вычислительные сети, объединяющие до нескольких сотен узлов, с длиной связей до 100 м, располагаемых чаще всего в специальном машинном зале.	<ol style="list-style-type: none"> 1. Глобальные сети (WAN) 2. Региональные сети (MAN) 3. Локальные сети (LAN) 4. Системные сети (SAN)
38. Логическая и техническая организация телекоммуникационных вычислительных сетей, включающая совокупность сетевых аппаратных и программных средств, методов доступа и используемых протоколов.	<ol style="list-style-type: none"> 1. Архитектура сети 2. Сетевая совокупность 3. Детерминант сети 4. Протокольная сеть
39. Какой из вариантов не является верным типом обработки в логических структурах.	<ol style="list-style-type: none"> 1. Переадресация данных 2. Изменение информационного содержания сообщения 3. Обработка сообщения 4. Информационная обработка
40. Эти интерфейсы обеспечивают связь между соседними иерархическими уровнями внутри одного узла.	<ol style="list-style-type: none"> 1. Протокольный интерфейсы 2. Рассылочные интерфейсы 3. Обслуживающие интерфейсы 4. Интерфейсы доступа
41. Уровень предназначен для организации доступа каждого пользователя к программам удаленных пользователей. На нём предусмотрены такие функции как пересылка файлов, пересылка заданий обращение к базам данных.	<ol style="list-style-type: none"> 1. Сеансовый уровень 2. Прикладной уровень 3. Физический 4. Сетевой
42. Уровень обеспечивает средства, необходимые абонентам для организации, синхронизации и административного управления обменом данными между ними, обеспечивает по запросам процессов создание портов для приёма и передачи сообщений, организацию соединений, а также выполнение универсальных функций управления для протоколов представительного уровня.	<ol style="list-style-type: none"> 1. Сеансовый уровень 2. Прикладной уровень 3. Физический 4. Сетевой
43. Уровень обеспечивает передачу двоичных сигналов через передающую среду. На	<ol style="list-style-type: none"> 1. Канальный уровень 2. Прикладной уровень

нём реализуются такие функции управления каналом связи, как подключение и отключение, формирование и приём сигналов.	<ol style="list-style-type: none"> 3. Физический 4. Сетевой
44. Дейтаграмма – это	<ol style="list-style-type: none"> 1. Пакет, заголовок которого содержит адрес отправителя. 2. Пакет, заголовок которого содержит адрес получателя и необходимые маршрутные признаки 3. Совокупность маршрутов, необходимая для нормального функционирования канала данных 4. Совокупность заголовков, предназначенных для установления адреса отправителя
45. Устройство, осуществляющее поочерёдное подключение нескольких входных каналов связи на один выходной без изменения скорости передачи.	<ol style="list-style-type: none"> 1. Повторитель 2. Коммутатор 3. Мост 4. Маршрутизатор
46. Устройство, выполняющее соединение на транспортном уровне. Обеспечивает соединение логически не связанных сетей, имеющих одинаковые протоколы на сеансовом уровне и выше, создает нужный логический канал и передаёт сообщение по назначению.	<ol style="list-style-type: none"> 1. Повторитель 2. Коммутатор 3. Мост 4. Маршрутизатор
47. Какой кабель необходимо использовать, если необходимо обеспечить высокую помехозащищенность и наиболее высокую скорость передачи сигналов.	<ol style="list-style-type: none"> 1. Витая пара 2. Волоконно-оптический 3. Инфракрасный 4. Коаксиальный
48. Код используется во многих системах передачи данных, обеспечивает изменение уровня сигнала при передаче каждого бита, что обеспечивает возможность самосинхронизации.	<ol style="list-style-type: none"> 1. Код NRZ 2. Код RZ 3. Манчестерский код 4. Код AMI
49. Какая технология использует толстый (полдюйма) коаксиальный кабель, обеспечивает сегменты длиной до 500 м, с числом подключаемых к сегменту узлов до 100.	<ol style="list-style-type: none"> 1. 10 Base-F 2. 10 Base-5 3. 10 Base-T 4. 10 Base-2
50. Протокол поддерживает передачу сообщений электронной почты между произвольными узлами сети Интернет, содержит механизмы промежуточного хранения почты и повышения надёжности доставки. Обеспечивает группирование сообщений в адрес одного получателя и размножение нескольких копий одного сообщения для передачи в разные адреса. Допускает использование транспортных служб.	<ol style="list-style-type: none"> 1. Telnet 2. FTP 3. SMTP (Simple Mail Transfer Protocol) 4. HTTP (Hyper Text Transfer Protocol)

14. КОМПЛЕКТ ЭКЗАМЕНАЦИОННЫХ БИЛЕТОВ

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №1

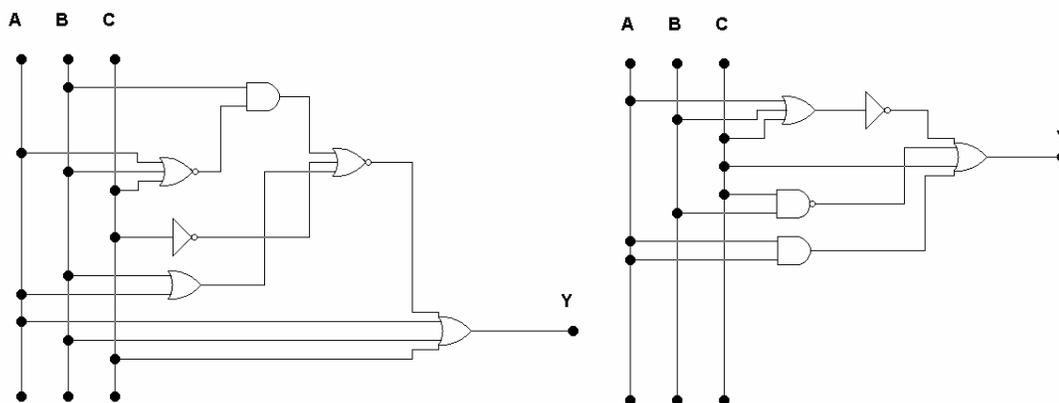
1. Основные понятия о процессе автоматизированной обработки данных. Определения. Меры информации.
2. Персональные компьютеры, особенности архитектуры и применения. Системная плата, внутримашинный и шинный интерфейсы, основные используемые шины.
3. Задачи

- 3.1. Реализовать ЛФ на мультиплексоре

$$f_1 = ABC + A'B'C' + A'B'C + A'BC$$

$$f_2 = AB(C+A)' + B + CA + A'C'$$

- 3.2. Составить логическую функцию по схеме



ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №2

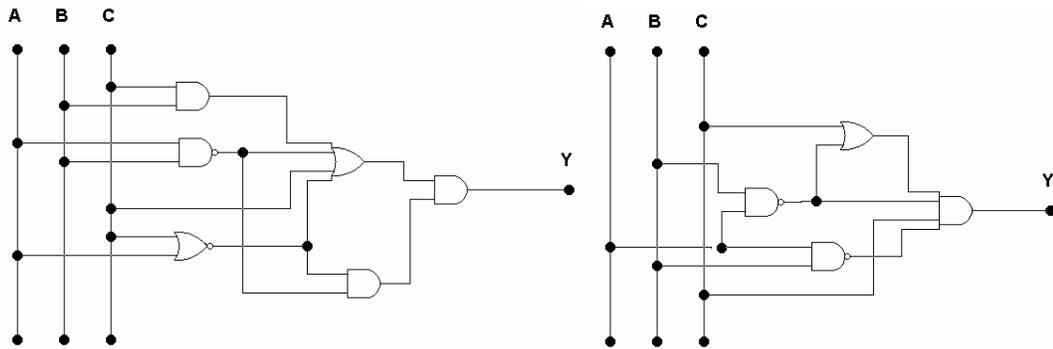
1. Основные понятия о процессе автоматизированной обработки данных. Показатели качества информации. Показатели качества и функционирование системы управления.
2. Персональные компьютеры, особенности архитектуры и применения. Запоминающие устройства.
3. Задачи

- 3.1. Реализовать ЛФ на мультиплексоре

$$f_1 = AB'C + A'B'C' + A'BC + ABC$$

$$f_2 = (A+B)C' + C'AB(B+C)$$

- 3.2. Составить ЛФ по схеме



ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №3

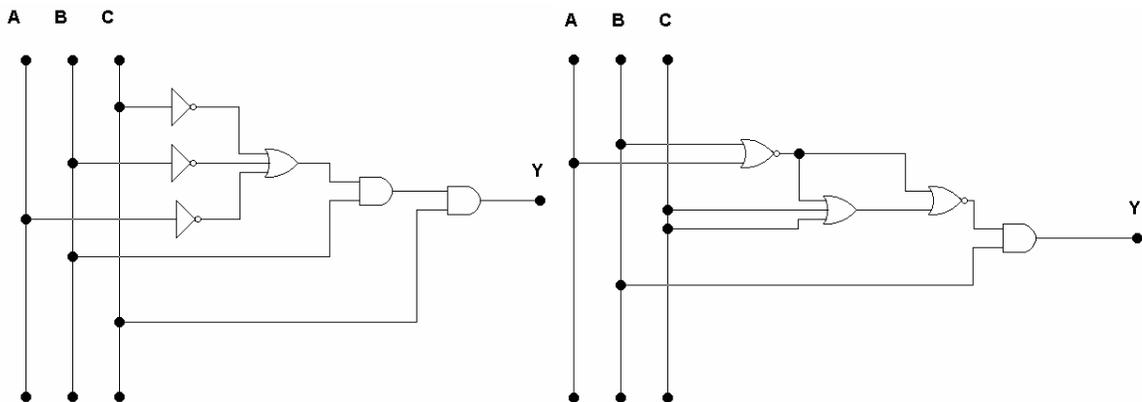
1. Принцип действия ВМ. Основы двоичной системы исчисления, виды систем исчисления, перевод из одной формы записи в другую.
2. Проектирование микропроцессорных систем и средств автоматики на их основе. Уровни проектирования. Классификация ошибок, неисправностей, дефектов.
3. Задачи

3.1. Реализовать ЛФ на мультиплексоре

$$f_1 = AB'C + A'BC' + ABC' + A'B'C'$$

$$f_2 = AC(C+A+B') + (A+B+C)'$$

3.2. Составить ЛФ по схеме



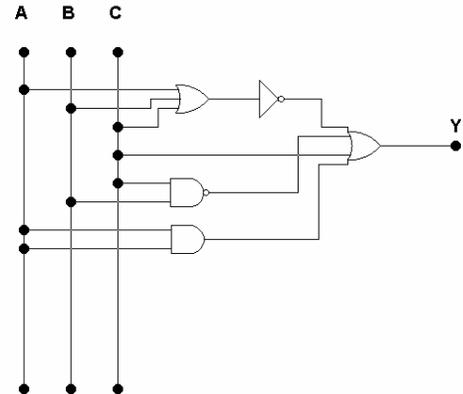
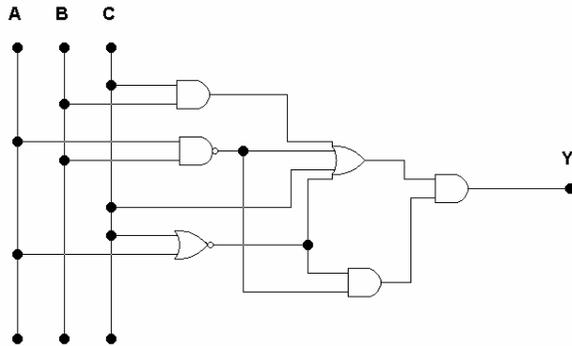
ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №4

1. Принцип действия ВМ. Основы алгебры логики. Базовые логические операции и логические элементы.
 2. Проектирование микропроцессорных систем и средств автоматики на их основе. Обнаружение неисправностей, функции средств отладки.
 3. Задачи
- 3.1. Реализовать ЛФ на мультиплексоре

$$f_1 = ABC + A'BC + AB'C + A'B'C + ABC'$$

$$f_2 = ((A+BC)' + AB + CBA)'$$

3.2. Составить ЛФ по схеме



ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №5

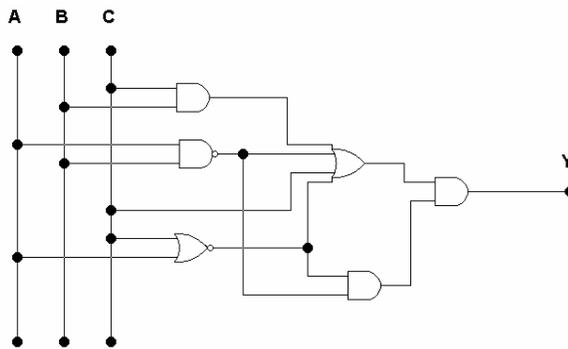
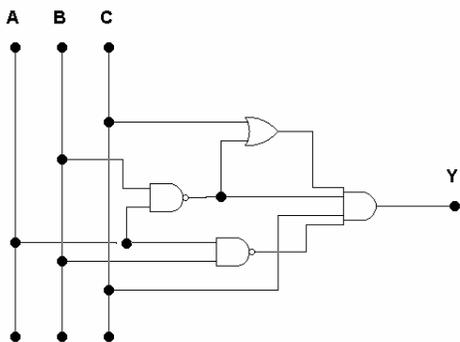
1. Принцип действия ВМ. Основные законы алгебры логики, виды записи логических выражений, переход от одной формы представления работы цифрового устройства к другой.

2. Проектирование микропроцессорных систем и средств автоматики на их основе. Основные этапы проектирования, комплексная отладка.

3. Задачи

3.1. Построить схему 7-сегментного дисплея, отображающего только четные числа от 0 до 9 («0» - считать четным числом)

3.2. Составить ЛФ по схеме



ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №6

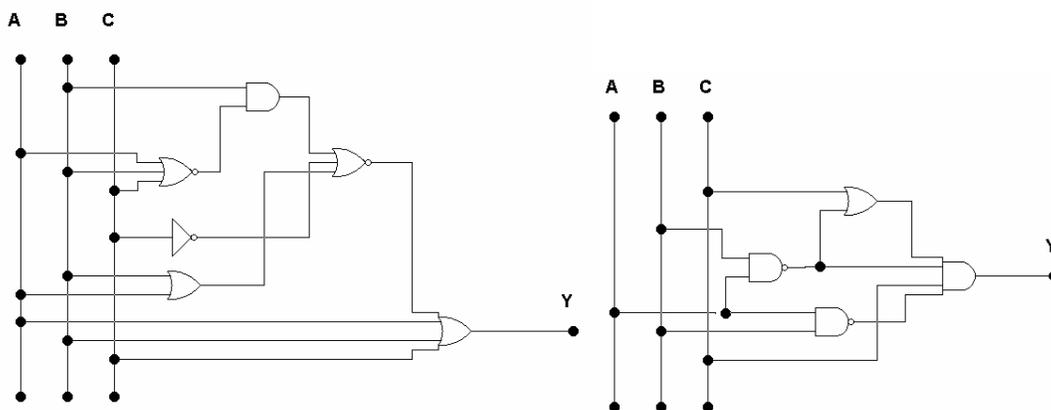
1. Принцип действия ВМ. Представление работы цифрового устройства в различных базисах.

2. Встроенные средства тестирования и отладки при работе системы.

3. Задачи

3.1. Построить схему 7-сегментного дисплея, отображающего только нечетные числа от 0 до 9 («0» - считать нечетным числом)

3.2. Составить ЛФ по схеме



ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №7

1. Принцип действия ВМ. Комбинационные устройства средней степени интеграции: шифраторы, дешифраторы, мультиплексоры, сумматоры, полу-сумматоры.
2. Процесс разработки программного обеспечения: определения основные инструменты, процесс создания.
3. Задачи

3.1. Реализовать ЛФ в базисе И-НЕ

$$f_1 = ABC + A'B'C' + A'BC + A'BC'$$

$$f_2 = AB(C+A)' + B + CA + A'C'$$

3.2. Создать схему, которая обнаруживает (на выходе схемы, в момент обнаружения устанавливается «1») хотя бы один кратковременный спад (установка в «0») сигнала на своем входе. Считать, что первоначально сигнал на входе имел уровень «1».

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №8

1. Принцип действия ВМ. Простейшие последовательностные схемы: триггеры и основные элементы на триггерных схемах (счетчики, регистры)
2. Особенности создания исполняемых программ на языке ассемблер.
3. Задачи

3.1. Реализовать ЛФ в базисах ИЛИ-НЕ

$$f_1 = ABC + A'B'C' + A'BC + A'BC'$$

$$f_2 = AB(C+A)' + B + CA + A'C'$$

3.2. Создать схему, которая обнаруживает (на выходе схемы, в момент обнаружения устанавливается «1») сначала спад (установка в «0»), затем повышение (установка в «1») сигнала на своем входе. Считать, что первоначально сигнал на входе имел уровень «1».

начально сигнал на входе имел уровень «1». На выходе устанавливается «1» только тогда, когда на входе вновь появится «1».

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №9

1. Архитектура и классификация ВМ. Определения, основные характеристики ВМ.
2. Структура программного обеспечения ВМ – предназначение и основные характеристики. Системное программное обеспечение – классификация.
3. Задачи

3.1. Реализовать ЛФ в базисах И-НЕ

$$f_1 = AB'C + A'B'C' + A'BC + ABC$$

$$f_2 = (A+B)'C' + C'AB(B+C)$$

3.2. Создать схему, которая обнаруживает (на выходе схемы, в момент обнаружения устанавливается «1») хотя бы один кратковременный импульс повышения уровня (установка в «1») сигнала на своем входе. Считать, что первоначально сигнал на входе имел уровень «0».

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №10

1. Архитектура и классификация ВМ. Классификация ВМ по принципу действия, элементной базе, по назначению, по мощности. Основные области применения ВМ.
2. Системное обеспечение современных ВМ. Операционные системы, системы автоматизации программирования.
3. Задачи

3.1. Реализовать ЛФ в базисах ИЛИ-НЕ

$$f_1 = AB'C + A'B'C' + A'BC + ABC$$

$$f_2 = (A+B)'C' + C'AB(B+C)$$

3.2. Создать схему, которая обнаруживает (на выходе схемы, в момент обнаружения устанавливается «1») сначала повышение (установка в «1»), затем спад (установка в «0») сигнала на своем входе. Считать, что первоначально сигнал на входе имел уровень «0». На выходе устанавливается «1» только тогда, когда на входе вновь появится «0».

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №11

1. Общие понятия о функциональной и структурной организации ВМ. Обобщенная структура ВМ, ее подсистемы.

2. Комплекс технического обслуживания, пакеты программ дополняющих возможности и системы документации ВМ.

3. Задачи

3.1. Реализовать ЛФ в базисах И-НЕ

$$f_1 = AB'C + A'BC' + ABC' + A'B'C'$$
$$f_2 = AC(C+A+B') + (A+B+C)'$$

3.2. Создать схему, которая обнаруживает (на выходе схемы, в момент обнаружения устанавливается «1») хотя бы один кратковременный спад (установка в «0») сигнала на своем входе. Считать, что первоначально сигнал на входе имел уровень «1».

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №12

1. Аппаратные особенности ВМ первого и второго поколений. Структура простейшего АЛУ.

2. Специальное или прикладное обеспечение современных ВМ. Пакеты прикладных программ, различные системы обработки.

3. Задачи

3.1. Реализовать ЛФ в базисах ИЛИ-НЕ

$$f_1 = ABC + A'BC + AB'C + A'B'C + ABC'$$
$$f_2 = ((A+BC)' + AB + CBA)'$$

3.2. Создать схему, которая обнаруживает (на выходе схемы, в момент обнаружения устанавливается «1») сначала повышение (установка в «1»), затем спад (установка в «0») сигнала на своем входе. Считать, что первоначально сигнал на входе имел уровень «0». На выходе устанавливается «1» только тогда, когда на входе вновь появится «0».

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №13

1. Аппаратные особенности ВМ третьего, четвертого и пятого поколений.

2. Централизованные и распределенные системы обработки данных. Вводные понятия и типовая структура.

3. Задачи

3.1. По заданной таблице истинности получить логическую функцию

Входы			Выходы	
a	b	c	F ₁	F ₂
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	1
1	0	1	1	0
1	1	0	0	0
1	1	1	1	1

3.2. Построить схему суммирующего счетчика с $K_{сч}=3$

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №14

1. Основные принципы построения и функционирования ВМ. Кризис структуры фон-Неймана, ВМ шестого поколения.
2. Централизованные и распределенные системы обработки данных. Организация микроконтроллерных систем.
3. Задачи

3.1. По заданной таблице истинности получить логическую функцию

Входы			Выходы	
a	b	c	F ₁	F ₂
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	0
1	1	0	0	1
1	1	1	0	1

3.2. Построить схему вычитающего счетчика с $K_{сч}=3$

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №15

1. Особенности организации современных ВМ. SISD компьютеры. CISC и RISC архитектура.
2. Типовая структура микроконтроллера, общие сведения.
3. Задачи

3.1. По заданной логической функции получить таблицу истинности

$$f_1 = ABC + A'B'C' + A'B'C + A'BC$$

$$f_2 = AB(C+A)' + B + CA + A'C'$$

3.2. Построить схему суммирующего счетчика с $K_{сч}=5$

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №16

1. Особенности организации современных ВМ. Основы организации суперскалярной обработки данных.
2. Параллельная обработка данных как архитектурный способ повышения производительности.
3. Задачи

3.1. По заданной логической функции получить таблицу истинности

$$f_1 = AB'C + A'B'C' + A'BC + ABC$$

$$f_2 = (A+B)C' + C'AB(B+C)$$

3.2. Построить схему вычитающего счетчика с $K_{сч}=5$

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №17

1. Особенности организации современных ВМ. SIMD компьютеры. Матричная, векторно-конвейерная архитектура, ММХ технология.

2. Классификация систем параллельной обработки данных. Основные архитектуры многопроцессорных вычислительных систем.

3. Задачи

3.1. По заданной логической функции построить схему

$$f_1 = AB'C + A'BC' + ABC' + A'B'C'$$

$$f_2 = ((A+BC)' + AB + CBA)'$$

3.2. Построить схему суммирующего счетчика с $K_{сч}=7$

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №19

1. Организация микропроцессоров. Предназначение, характеристика, состав и принцип работы операционного блока.

2. Многопроцессорные вычислительные системы (с общей шиной, с многовходовой памятью).

3. Задачи

3.1. Упростить логическое выражение

$$f_1 = ABC + ABC' + AB'C + A'B'C$$

$$f_1 = BC + (C+A)' + C'(C'+B')A'$$

3.2. Построить временную диаграмму работы цифрового устройства, таблица истинности которого представлена ниже. Считать, что для функций F_1 и F_2 переключение происходит по заднему фронту импульса C .

Входы			Выходы	
a	b	c	F_1	F_2
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	1
1	0	1	1	0
1	1	0	0	0
1	1	1	1	1

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №20

1. Организация микропроцессоров. Предназначение, характеристика, состав и принцип работы блока управления и интерфейсного блока.
2. Многомашинные вычислительные системы: многомашинные комплексы, системы массового параллелизма.
3. Задачи

3.1. Упростить логическое выражение

$$f_1 = A'B'C + ABC' + AB'C + A'BC'$$

$$f_1 = (C + C'A)(A + B(C + A))' + AB + BC$$

3.2. Построить временную диаграмму работы цифрового устройства, таблица истинности которого представлена ниже. Считать, что для функций F_1 и F_2 переключение происходит по переднему фронту импульса C .

Входы			Выходы	
a	b	c	F_1	F_2
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	0
1	1	0	0	1
1	1	1	0	1

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №21

1. Особенности организации процессоров при использовании внутренних регистров.
2. Вычислительные системы – состояния производства, направления развития высокопроизводительных вычислительных систем, тенденции развития архитектур с общей и разделяемой памятью.
3. Задачи

3.1. Упростить логическое выражение

$$f_1 = ABC + A'BC' + A'B'C + A'BC'$$

$$f_1 = (C + C'A) + (ABC)'B'C$$

3.2. Построить временную диаграмму работы цифрового устройства, таблица истинности которого представлена ниже. Считать, что для функции F_1 переключение происходит по переднему фронту импульса C , для функции F_2 переключение происходит по заднему фронту импульса C .

Входы			Выходы	
a	b	c	F_1	F_2
0	0	0	0	1
0	0	1	1	0
0	1	0	0	0
0	1	1	1	0
1	0	0	0	1
1	0	1	1	1
1	1	0	0	0
1	1	1	1	1

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №22

1. Система команд микропроцессора.
2. Вычислительные системы – развитие архитектур микропроцессоров, направления развития мультипроцессорных систем с распределенной памятью.
3. Задачи

3.1. Упростить логическое выражение

$$f_1 = AB'C' + AB'C + ABC' + ABC$$

$$f_1 = ((BA + CB)(BB + C'C)' + AC)'$$

3.2. Построить временную диаграмму работы цифрового устройства, таблица истинности которого представлена ниже. Считать, что для функций F_1 переключение происходит по заднему фронту импульса C , для функции F_2 переключение происходит по переднему фронту импульса C

Входы			Выходы	
a	b	c	F_1	F_2
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	1	1
1	0	0	1	1
1	0	1	0	0
1	1	0	1	0
1	1	1	0	1

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №23

1. Основные понятия о способах и методах адресации.
2. Общие оценки производительности мультипроцессорных систем при увеличении числа процессоров. Вычислительные системы на кристалле и нанотехнологии.

3. Задачи
3.1. Упростить логическое выражение

$$f_1 = A'B'C + A'BC' + A'BC + AB'C' + AB'C + ABC' + ABC$$

$$f_1 = A + B + (A + ABC + CB)(ABC + ABC')$$

3.2. Построить временную диаграмму работы цифрового устройства, таблица истинности которого представлена ниже. Считать, что для функций F_1 переключение происходит по переднему фронту импульса C , для функции F_2 переключение происходит по заднему фронту импульса C

Входы			Выходы	
a	b	c	F_1	F_2
0	0	0	1	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	0	0
1	1	1	1	1

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №24

1. Организация памяти ВМ. Общая структура и характеристики каждого уровня.
2. Принципы построения телекоммуникационных вычислительных систем: понятие, организация, параметры, классификация, архитектура.
3. Задачи

3.1. Упростить логическое выражение

$$f_1 = A'BC' + A'BC + AB'C + ABC' + ABC$$

$$f_1 = A' + A'B + AC' + A(A+C)' + C$$

3.2. Построить временную диаграмму работы цифрового устройства, таблица истинности которого представлена ниже. Считать, что для функций F_1 переключение происходит по переднему фронту импульса C , для функции F_2 переключение происходит по заднему фронту импульса C

Входы			Выходы	
a	b	c	F_1	F_2
0	0	0	0	1
0	0	1	1	1
0	1	0	0	0
0	1	1	1	1
1	0	0	0	0
1	0	1	1	1
1	1	0	1	0
1	1	1	0	0

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №25

1. Организация памяти ВМ. Организация оперативного, сверх оперативного уровней. Методы управления оперативной памятью.
2. Принципы построения телекоммуникационных вычислительных систем: понятие, организация, параметры, классификация, архитектура.
3. Задачи

3.1. По заданной таблице истинности получить логическую функцию

Входы			Выходы	
a	b	c	F_1	F_2
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	1	0
1	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	1	1	0	1

3.2. Построить схему суммирующего счетчика с $K_{сч}=9$

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №26

1. Системы внешней памяти и особенности организации кэш-памяти. Методы повышения пропускной способности оперативной памяти.
2. Семиуровневая эталонная модель взаимодействия открытых систем (OSI).
3. Задачи

3.1. По заданной таблице истинности получить ЛФ

Входы			Выходы	
a	b	c	F ₁	F ₂
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	1
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

3.2. Построить схему суммирующего счетчика с Ксч=10

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №27

1. Организация обмена данными в ВМ – общие сведения. Обмен данными между периферийными устройствами и вычислительным ядром системы.
2. Коммуникация и маршрутизация при передаче данных.
3. Задачи

3.1. По заданной таблице истинности получить логическую функцию

Входы			Выходы	
a	b	c	F ₁	F ₂
0	0	0	0	0
0	0	1	1	0
0	1	0	0	1
0	1	1	0	0
1	0	0	1	1
1	0	1	1	0
1	1	0	0	1
1	1	1	0	1

3.2. Построить схему суммирующего счетчика с Ксч=11

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №28

1. Персональные компьютеры, особенности архитектуры и применения. Функциональная и структурная организация, характеристики ПК.
2. Локальные вычислительные сети и сеть Ethernet.
3. Задачи

3.1. По заданной логической функции построить схему

$$f_1 = A'BC' + A'BC + AB'C + ABC' + ABC$$

$$f_2 = A' + A'B + AC' + A(A+C)' + C = 1$$

3.2. Построить схему вычитающего счетчика с $K_{сч}=12$

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №29

1. Персональные компьютеры, особенности архитектуры и применения. Системная плата, внутримашинный и шинный интерфейсы, основные используемые шины.

2. Корпоративные сети и сеть Интернет.

3. Задачи

3.1. Реализовать логическую функцию на мультиплексоре

$$f_1 = A'BC' + A'BC + AB'C + ABC' + ABC$$

$$f_2 = A' + A'B + AC' + A(A+C)' + C = 1$$

3.2. Построить схему вычитающего счетчика с $K_{сч}=3$

15. КАРТА ОБЕСПЕЧЕННОСТИ ДИСЦИПЛИНЫ КАДРАМИ ПРОФЕССОРСКО-ПРЕПОДАВАТЕЛЬСКОГО СОСТАВА

Лекционный курс – к.т.н., доцент, Теличенко Денис Алексеевич;

Практические занятия – к.т.н., доцент, Теличенко Денис Алексеевич; к.т.н., старший преподаватель Безруков Николай Сергеевич;

Лабораторные занятия – к.т.н., доцент, Теличенко Денис Алексеевич; к.т.н., старший преподаватель Безруков Николай Сергеевич