Северо-Западный государственный заочный технический университет

**ЦИФРОВЫЕ УСТРОЙСТВА**

**И МИКРОПРОЦЕССОРЫ**

УЧЕБНО-МЕТОДИЧЕСКИЙ

КОМПЛЕКС

Санкт-Петербург

2010

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Государственное образовательное учреждение высшего профессионального образования

«СЕВЕРО-ЗАПАДНЫЙ ГОСУДАРСТВЕННЫЙ ЗАОЧНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра радиотехники

**ЦИФРОВЫЕ УСТРОЙСТВА И МИКРОПРОЦЕССОРЫ**

# УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС

Институт интеллектуальных электронных систем

Специальность

210302.65 – радиотехника

Направление подготовки бакалавра

210300.62 - радиотехника

Санкт-Петербург

Издательство СЗТУ

2010

*Утверждено редакционно-издательским советом университета*

###### УДК 621 396:537.81

**ЦИФРОВЫЕ УСТРОЙСТВА И МИКРОПРОЦЕССОРЫ:** учебно-методический комплекс /сост. О. Л. Соколов, О. С. Голод. - СПб.: Изд-во СЗТУ, 2010. - 169 с.

Учебно-методический комплекс разработан в соответствии с требо-ваниями государственных образовательных стандартов высшего профес-сионального образования по специальности 210302.65- «Радиотехника» и направлению подготовки бакалавра 210300.62- «Радиотехника».

В первой части учебно-методического комплекса рассматриваются: логические операции; логические схемы; способы задания логических функций; способы их минимизации; арифметические основы цифровой техники; особенности построения полусумматоров, полных сумматоров, устройств неравнозначности, равнозначности, шифраторов, дешифраторов, мультиплексоров, демультиплексоров, цифровых компараторов, триггеров, регистров, счетчиков, аналого-цифровых и цифроаналоговых преобразова-

телей, запоминающих устройств.

Во второй части комплекса рассмтриваются микропроцессорный комплект КР580 и микроконтроллер КМ1816, их аппаратные средства, программное обеспечение, системы команд и программирование в микро-процессорных системах.

Учебно-методический комплекс предназначен для студентов Института интеллектуальной электроники, специальности 210302.65 - «Радиотехника», изучающих дисциплину «Цифровые устройства и микропроцессоры». Материал комплекса может быть полезен при курсовом и дипломном проектировании, а также специалистам в области цифровой техники.

Рассмотрено на заседании кафедры радиотехники 25.03.10, одобрено методической комиссией Института радиоэлектроники 25.03.10.

Рецензенты: кафедра радиотехники (зав. кафедрой Л. Я. Родос, канд. техн. наук, проф.); В. Д. Лиференко, д-р техн. наук, проф. кафедры про-мышленной электроники СЗТУ.

Составители: О. Л. Соколов, канд. техн. наук, доц.

О. С. Голод, канд. техн. наук, доц.

© Северо-Западный государственный заочный технический университет, 2010.

© Соколов О. Л. , Голод О. С. , 2010.

1. Информация о дисциплине

## 1.1. Предисловие

Учебно-методический комплекс предназначен для студентов всех форм обучения специальности 210302.65 - "Радиотехника ". Предлагаемый для изучения материал содержит следующие основные разделы дисциплины: логические схемы, основы цифровой техники, комбинационные устройства, цифровые автоматы, аналогово-цифровые и цифроаналоговые преобразова-тели, запоминающие устройства.

Основное внимание уделяется принципу действия, устройству и характеристикам цифровых устройств и микропроцессоров (ЦУ и МП). Дисциплина ОПД.Ф.10 «Цифровые устройства и микропроцессоры» изучается студентами специальности 210302.65. всех форм обучения. В соответствии с учебным планом студент должен прослушать лекции, проделать лабораторный практикум, выполнить контрольную работу и практические занятия, сдать зачет, защитить курсовой проект и сдать экзамен.

Целью изучения дисциплины является изучение принципов построения современных цифровых устройств и микропроцессоров, теоретических основ их анализа, синтеза и исследования.

В результате изучения дисциплины студент должен овладеть основами знаний по дисциплине, формируемыми на нескольких уровнях.

Иметь представление:

- о целях применения цифровых устройств и микропроцессоров;

- об областях применения и перспективах развития теории и практики ЦУ и МП.

Знать:

- основные принципы построения ЦУ и МП;

- методы минимизации логических функций.

Уметь:

-применять изученные методы для решения задач проектирования.

-определять структуру и параметры ЦУ и МП.

Владеть:

- методами анализа ЦУ и МП;

- методами синтеза ЦУ и МП.

Место дисциплины в учебном процессе

Непосредственной базой для изучения данной дисциплины являются: математика, физика, основы теории цепей. Теоретическим фундаментом дисциплины ЦУ и МП является раздел математики – алгебра логики. Знания, полученные студентами по дисциплине, будут использованы при изучении устройств: радиоавтоматики, генерирования и формирования сигналов, приема и обработки сигналов, а также радиотехнических систем различного назна-чения, систем и сетей подвижной радиосвязи, транспортных информационно-управляющих РЭС.

**1.2.** С**одержание дисциплины и виды учебной работы**

##### 1.2.1. Содержание дисциплины по ГОС ОПД.Ф.10

##### Основы алгебры логики и теории переключательных функций; основы теории асинхронных потенциальных и синхронных автоматов; синтез цифровых узлов: триггеры, счетчики, шинные приемопередатчики, сдви-гающие регистры, мультиплексоры, демультиплексоры, сумматоры; применение интегральных схем для проектирования цифровых устройств; микропроцессоры: архитектура, система команд, интерфейсные большие интегральные схемы (БИС) и БИС памяти; проектирование микроконтроллеров на микропроцессорах, разработка программного обеспечения.

##### 1.2.2. Объем дисциплины и виды учебной работы.

|  |  |  |  |
| --- | --- | --- | --- |
| Вид учебной работы | Всего часов | | |
| Форма обучения | | |
| Очная | Очно-заоч-ная | Заочная |
| Общая трудоемкость дисциплины (ОТД) | 170 | | |
| Работа под руководством препо-  давателя (включая ДОТ) | 100 | 100 | 100 |
| В том числе аудиторные занятия:  лекции  практические занятия (ПЗ)  лабораторные работы (ЛР) | 48  8  28 | 12  4  24 | 8  4  12 |
| Самостоятельная работа студента (СР) | 70 | 70 | 70 |
| Промежуточный контроль, количество | 13 | 14 | 14 |
| В том числе: курсовой проект | 1 | 1 | 1 |
| контрольная работа | - | 1 | 1 |
| Вид итогового контроля (зачет, экзамен) | Зачет, экзамен | | |

1.2.3. Перечень видов учебной работы студента, текущего контроля успеваемости и промежуточной аттестации:

-лабораторные работы;

-практические занятия;

-курсовой проект;

-контрольная работа (для студентов очно-заочной и заочной форм обучения);

-тесты по разделам дисциплины;

-зачет;

-экзамен.

**2. РАБОЧИЕ УЧЕБНЫЕ МАТЕРИАЛЫ**

**2.1. Рабочая программа (объем дисциплины 170 часов)**

Введение (2 часа)

[1], с. 3…5; [7], с. 11…14, 413, 414, 453…471

Предмет и задачи дисциплины «Цифровые устройства и микропро-цессоры». Общее состояние отечественной и зарубежной цифровой микро-

схемотехники и наноэлектроники.

Микропроцессорные комплекты, архитектура микропроцессорных сис-тем, принципы организации работы, программное обеспечение, системы команд, подпрограммы, особенности программирования на языках высокого уровня.

Проектирование микроконтроллеров на микропроцессорах, разработка программного обеспечения.

**1.** **ЦИФРОВЫЕ УСТРОЙСТВА**

Раздел 1. Логические схемы (10 часов)

1.1. Алгебра логики (5 часов)

[1], с. 4…6; [2], с. 5…9; [8], с. 16…22

Операции алгебры логики (булевой алгебры). Понятие переклю-чательной функции, сингулярные и бинарные функции. Основные теоремы, аксиомы и тождества, используемые для упрощения логических выражений. Принцип двойственности. Теорема де Моргана.

1.2. Построение логических схем (5 часов)

[1], с. 7…13; [2], с. 9…22; [8], с. 39…52

Способы представления функций: словесное описание, таблица истинности, алгебраическое выражение (структурная формула). Переход от структурной формулы к логической схеме и обратно. Логические базисы, реализующие функционально полную систему. Логические функции: отрицание дизъюнкции и отрицание конъюнкции. Понятие об универсальных логических элементах.

Минимизация логических функций. Понятие о минтермах и макстермах. Совершенная дизъюнктивная нормальная форма (СДНФ). Совершенная конъюнктивная нормальная форма (СКНФ).

Основные методы минимизации (получение тупиковой формы). Алгебраический метод. Метод карт Карно-Вейча. Недоопределенные функции. Табличный метод Квайна-МакКласки.

Раздел 2. Основы цифровой техники (12 часов)

2.1. Системы счисления (6 часов)

[1], с. 13…18; [2], с. 23…28

Представление целых и дробных чисел в разных системах счисления.Двоичная, восьмеричная, десятичная и 16-ричная системы счисления. Перевод чисел из одной системы счисления в другую. Двоично-десятичная система кодирования чисел. Наиболее распространенные виды двоично-десятичного кодирования: двоично-десятичный код 8-4-2-1, код Айкена (2-4-2-1), код «с избытком 3», код Джонсона и код Грея.

2.2. Формы представления чисел (6 часов)

[1], с. 19…20; [2], с. 28…30

Формы представления чисел. Числа с фиксированной точкой. Предста-

вление знака числа. Числа с плавающей точкой. Понятие об обратном и дополнительном кодах.

Раздел 3. Комбинационные устройства (19 часов)

3.1. Полусумматоры, сумматоры,

устройства неравнозначности и равнозначности (7 часов)

[1], с. 21…24; [2], с.31…34; [8], с. 80…83

Задачи синтеза комбинационного устройства. Устройство неравно-значности (сумма по модулю два) и его свойства. Устройство равнозначности, таблица истинности устройства.

3.2. Шифраторы, дешифраторы и кодопреобразователи (6 часов)

[1], с. 24…27, 44, 45; [2], с. 34…35, 38…40; [8], с. 83…85

Полные и неполные дешифраторы (декодеры). Шифраторы и кодопреобразователи, минимизация с помощью карт Карно. Шифратор и дешифратор на программируемых логических матрицах (ПЛМ).

3.3. Мультиплексоры, демультиплексоры и компараторы (6 часов)

[1], с. 28…31, 45,46; [2], с. 35…37; [8], с. 212…214

Демультиплексоры, мультиплексоры, их назначение, вопросы синтеза и каскадирования. Реализация на ПЛМ. Цифровые компараторы.

Раздел 4. Цифровые автоматы (18 часов)

4.1. Асинхронные триггеры (6 часов)

[1], с. 31…35; [2], с. 48…52; [8], с. 89…93

Потенциальные и импульсные сигналы. Операторы перехода. Основные тождества, связывающие потенциальные и импульсные сигналы. Модели асинхронных и синхронных потенциальных автоматов. Триггеры. Асин-хронные потенциальные триггеры RS-типа с инверсными и прямыми входами. Триггеры типа E и JK. Синтез асинхронных потенциальных триггеров. Функции возбуждения. Счетный режим. Т-триггер.

4.2. Синхронные триггеры (6 часов)

[1], с. 35…37; [2], с. 52…54; [8], с. 96…102

Синхронные триггеры. Синхронные триггеры типа RSC, типа JK и типа D. MS-триггер. Функция возбуждения синхронных триггеров и общая методика их синтеза. Счетный режим в триггерах типа JK и D.

4.3. Регистры и счетчики (6 часов)

[1], с. 37…41; [2], с. 54…60; [8], с. 104…112

Сдвигающие регистры.Классификация сдвигающих регистров. Простые сдвигающие регистры. Реверсивные сдвигающие регистры. Универсальные сдвигающие регистры.

Счетчики.Двоичные и двоично-десятичные счетчики. Каскадирование счетчиков. Реверсивные счетчики, особенности их каскадирования. Счетчики на сдвигающих регистрах. Счетчики с произвольным модулем счета. Счетчики Джонсона.

Раздел 5. Аналого-цифровые и

цифроаналоговые преобразователи (12 часов)

5.1. Аналого-цифровые преобразователи (АЦП) (6 часов)

[1], с. 41…45; [2], с. 61…64; [8], с. 331…337

Классификация АЦП по времени преобразования. АЦП параллельного типа. АЦП с промежуточным преобразованием напряжения во временной интервал. Принципы конвейерной обработки в АЦП. Точность работы АЦП и факторы, влияющие на точность преобразования.

5.2. Цифроаналоговые преобразователи (ЦАП) (6 часов)

[1], с. 45…47; [2], с. 64…66; [6], с. 382…386

Принципы работы ЦАП. Использование матрицы *R-2R.* Методы умно-жения аналоговых сигналов. Получение среднеквадратичного значения сигнала.

Раздел 6. Запоминающие устройства (10 часов)

6.1. Оперативные запоминающие устройства (ОЗУ) (5 часов)

[1], с. 47…52; [2], с. 68…72; [5], с. 63…67

Основные параметры запоминающих устройств (ЗУ). Адресация, информационная емкость, разрядность. Единицы для выражения значений емкости ЗУ. Быстродействие ЗУ. Понятие о времени выборки и цикле записи. Характеристика ЗУ по потребляемой мощности, набору питающих напряже-

ний и времени хранения информации.

Типовая структура ОЗУ матричного вида. Управляющие цепи для обеспечения режима хранения, чтения и записи информации. Определение числа строк и столбцов матрицы элементов памяти. Условное обозначение микросхемы ОЗУ. Временные диаграммы сигналов. Схема наращивания разрядности. Динамические ОЗУ.

6.2. Постоянные запоминающие устройства (ПЗУ) (5 часов)

[1], с. 52…54; [2], с. 72…76; [5], с. 67…74

Классификация ПЗУ по способу занесения информации. ПЗУ, програм-

мируемые маской на предприятии-изготовителе. ПЗУ, программируемые пользователем. Перепрограммируемые ПЗУ (ППЗУ). Способы стирания информации.

**2. МИКРОПРОЦЕССОРЫ**

Раздел 7. Микропроцессорный комплект КР580 (14 часов)

7.1. Архитектура микропроцессорной системы (5 часов)

[1], с. 55…60; [3], с. 5…11; [5], с. 76…79

Трехшинная архитектура микропроцессорной системы***.*** Структурная схема микропроцессорной системы с трехшинной архитектурой, ее основные узлы: центральный процессор, память и внешние устройства. Назначение шин адреса, данных и управления. Основные сигналы управления операциями ввода/вывода.

ОднокристальныеМП*.*Структурная схема однокристальных МП на примере МП КР580ВМ80. Основные узлы МП, буферы шин адреса и данных, регистры общего назначения (РОН), регистр команд, программный счетчик, схема синхронизации и управления, арифметико-логическое устройство (АЛУ), указатель стека.

7.2. Организация работы ЦПЭ КР580ВМ80 (9 часов)

[1], с. 60…62; [3], с. 11…14; [5], с. 90…99

Линии синхронизации. Машинные циклы. Генератор тактовых импуль-сов КР580ГФ24. Структурная схема генератора. Схема подключения гене-ратора к центральному процессорному элементу. Формирование сигнала СТРОБ СОСТОЯНИЯ. Использование сигналов СБРОС и ГОТОВНОСТЬ. Установка МП в режим ожидания на заданное время. Обеспечение шагового режима.

Раздел 8. Аппаратные средства МПК КР580 (14 часов)

8.1. Системный контроллер, шинные формирователи (7 часов)

[1], с. 62…66; [3], с. 23…27

Генераторы, системные контроллеры и шинные формирователи. Генератор тактовых импульсов КР580ГФ24. Структурная схема генератора. Схема подключения генератора к центральному процессорному элементу. Формирование сигнала СТРОБ СОСТОЯНИЯ. Использование сигналов СБРОС и ГОТОВНОСТЬ. Установка МП в режим ожидания на заданное время. Обеспечение шагового режима.

Системный контроллер КР580ВК28/38. Структурная схема системного контроллера, его назначение и принцип действия.

Шинные формирователи КР580ВА86/87. Назначение, структурная схема и принцип действия.

8.2. Параллельный и последовательный интерфейсы, таймер (7 часов)

[1], с. 66…72; [3], с. 27…35

Интерфейс ввода/вывода. Интерфейс параллельного КР580ВВ55 и последовательного КР580ВВ51 ввода/вывода .Структурная схема програм-

мируемого интерфейса.

Раздел 9. Программное обеспечение МПК КР580 (14 часов)

9.1. Программная модель МП системы (7 часов)

[1], с. 72…74, 15; [3], с. 14…15

Программная модель МП типа КР580ВМ80А. Внутренние регистры: регистры данных, регистры управления. Внешние регистры: память, средства ввода/вывода. Принцип программного объединения 8-битных РОН в 16- битные регистровые пары. *H-*пара как основной указатель памяти.

9.2. Способы адресации в МП системе (7 часов)

[1], с. 74,75; [3], с. 15…17

Режимы адресации и система команд МПК КР580ВМ80*.*Прямая адресация. Непосредственная адресация. Регистровая адресация. Косвенная адресация. Возможность программного осуществления индексной адресации.

Раздел 10. Система команд МПК КР580 (16 часов)

10.1. Команды пересылки данных

и арифметико-логические команды (8 часов)

[1], с. 75…79; [3], с. 17…21

Системы команд. Разделение системы команд МП на группы в соответствии с их функциональным назначением. Группы команд пересылки, арифметических и логических операций.

10.2. Команды передачи управления, ввода-вывода и др. (8 часов)

[1], с. 79…81; [3], с. 21…22

Команды передачи управления, команды ввода-вывода, специальные команды. Команды передачи управления, не содержащие условия (безуслов-ный переход) и содержащие условие (условный переход, или переход по условию).

Раздел 11. Программирование в МП системе (14 часов)

11.1. Программирование на машинном языке (7 часов)

[1], с. 83; [3], с. 36…39

Программирование на машинном языке. Формат бланка при програм-

мировании на машинном языке.

11.2. Программирование на ассемблере (7 часов)

1], с. 84…96; [3], с. 39…59.

Директивы ассемблера. Назначение директив и псевдокоманд. Особен-

ности содержимого полей ассемблера при наличии директив. Формат директив ORG, END, EQU, SET, IF, ENDIF, DB, DW, DS.

Макрокоманды.Применение макрокоманд в прикладных программах. Макрокоманды с формальными параметрами. Формат макрокоманды. Формат обращения к макрокоманде. Понятие о расширении макрокоманды.

Структуры данных и организация подпрограмм в МПК КР580ВМ80.

Организация массивов, очередей, стеков.

Понятие одномерного массива. Определение адреса элемента массива. Двумерный массив, способ размещения в памяти.

Многомерные массивы, способ обработки. Структура данных в виде очереди. Понятие длины очереди. Использование очереди при вводе и выводе символьных данных. Принцип организации очереди. Кольцевая организация очереди.

Стек как специальная разновидность одномерного массива. Принцип загрузки элементов данных в стек и извлечение их из стека. Организация стека с двумя концами (полки) по принципу LIFO.

Подпрограммы как средство модульного программирования**.**Общая организация подпрограмм.

Подпрограмма как законченный сегмент (модуль) программы. Порядок вызова подпрограммы и возврата в основную программу. Вложенные подпрограммы. Иерархическая организация подпрограмм.

Особенности программирования на языках высокого уровня*.*Положи-

тельные и отрицательные качества языков высокого уровня при програм-мировании МП систем.

Понятие о компромиссах между аппаратнымии программными сред-ствами.

Раздел 12. Микроконтроллер КМ1816ВЕ48 (14 часов)

12.1. Аппаратное обеспечение КМ1816ВЕ48 (7 часов)

[1], с. 97…101; [3], с. 61…65

Особенности архитектуры микроконтроллера КМ1816ВЕ48. Структурная схема, основные узлы: регистры общего назначения (РОН), внутреннее оперативное запоминающее устройство и стек, внутреннее постоянное запоминающее устройство, программный счетчик, схема синхронизации и управления, АЛУ, внутренний таймер, интерфейс ввода/вывода (ВВ).

Программируемый ВВ с квитированием. ВВ по прерыванию. ВВ с прямым доступом к памяти.

Назначение в составе аппаратных средств микроконтроллера стираемого перепрограммируемого ПЗУ (СППЗУ) программ емкостью 1 Кбайт, воз-можность пределов его расширения. Назначение регистрового ОЗУ данных. Обеспечение прямой адресации внешнего ОЗУ.

Реализация в МК 1816 системы векторного прерывания от двух источников: внутреннего таймера-счетчика событий и внешнего источника.

12.2. Программное обеспечение КМ1816ВЕ48 (7 часов)

[1], с. 102…109; [3], с. 66…76

Особенности программирования КМ1816ВЕ48. Формат команд. Способы адресации в командах микроконтроллера КМ1816. Особенности языка ассемблер для КМ1816. Команды, выполняемые за один машинный цикл и за два машинных цикла. Классификация групп команд по функциональному признаку. Применение специальных символов #, @ при написании программ. Особенности команд операций с таймером.

Заключение(1 час)

[5], с. 10…32; [6], с. 5…12

Улучшение технических параметров цифровых устройств при совершенствовании схем микроэлектроники и наноэлектроники.

Расширение сферы применения микропроцессоров и микроконтрол-

леров в радиотехнических системах различного назначения.

**2.2. Тематический план дисциплины**

очная форма обучения

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № п/п | Наименование  раздела  (темы) | Кол-во час. по дн. фор-ме обу-чения | Виды занятий и контроля | | | | | | | | | | | |
| Лекции | | ПЗ (С) | | ЛР | | Самоc  .  раб  о  т  а | Тесты | Контр. работы | ПЗ | ЛР | КП |
| ауд. | ДОТ | ауд | ДОТ | ауд. | ДОТ |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  | ВСЕГО | 170 | 48 | **-** | 8 | **16** | 28 | **-** | 70 | - | - | - | - | 1 |
|  | Введение | 2 | 2 |  |  |  |  |  |  |  |  |  |  |  |
| 1 | Раздел 1. Логические схемы | 10 | 5 | - | - | **3** | - | - | 2 | №1 | - | - | - | - |
| 1.1 | Алгебра логики | 5 | 3 | - | - | **1** | - | - | 1 | - | - | №1 | - | - |
| 1.2 | Построение логических схем | 5 | 2 | - | - | **2** | - | - | 1 | - |  | 3 | - | - |
| - | -- | - |
| 2 | Раздел 2. Основы цифровой техники | 12 | 4 | - | 2 | **2** | - | **-** | 4 | №2 |
| 5 | - | - |
| 2.1 | Системы счисления | 6 | 2 | - | 1 | **1** | - | - | 2 | - |
| 6 | - | - |
| 2.2 | Формы представления чисел | 6 | 2 | - | 1 | **1** | - | - | 2 | - |
| 3 | Раздел 3 Комбинационные устройства | 19 | 7 | **-** | 2 | **4** | 3 | **-** | 3 | №3 | - | - | - | - |
| 3.1 | Полусумматоры, сумматоры, устройства неравнозначности и равнозначности | 7 | 4 | **-** | - | **2** | 1 | **-** | - | - | - | 7,8,9,10 | №1 | - |
| 3.2 | Шифраторы, дешифраторы и кодопреобразователи | 6 | 1 | - | 1 | **1** | 2 | **-** | 1 | - | - | 11 | №2 | - |
| 3.3 | Мультиплексоры, демультиплексоры и компараторы | 6 | 2 | - | 1 | **1** | - | **-** | 2 |  |  | 12 | №3 |  |
| 4 | Раздел 4. Цифровые автоматы | 18 | 6 | **-** | - | **3** | 9 | **-** | - | №4 | - | - | - | - |
| 4.1 | Асинхронные триггеры | 6 | 2 | **-** | - | **1** | 3 | **-** | - | - | - | 13 | №4 | - |
| 4.2 | Синхронные триггеры | 6 | 2 | **-** | - | **1** | 3 | **-** | - |  | - | 14 | №5 |  |
| 4.3 | Регистры и счетчики | 6 | 2 | **-** | - | **1** | 3 | **-** |  | - | - | 15 | №6 | - |
| 5 | Раздел 5. Аналого-цифровые и цифро-аналоговые преобразователи | 12 | 2 | **-** | - | **4** | - | **-** | 6 | №5 | - | - | - | - |
| 5.1 | Аналого-цифровые преобразователи | 6 | 1 | **-** | - | **2** | - | **-** | 3 | - | - | 16 | - | - |
| 5.2 | Цифроаналоговые преобразователи | 6 | 1 | **-** | - | **2** | - | **-** | 3 | - | - | 17 | - | - |
| 6 | Раздел 6. Запоминающие устройства | 10 | 2 | - | - | - | - | **-** | 8 | №6 | - | - | - | - |
| 6.1 | Оперативные запоминающие устройства | 5 | 1 | - | - | - | - | **-** | 4 | - | - | - | - | - |
| 6.2 | Постоянные запоминающие устройства | 5 | 1 | - | - | - | - | **-** | 4 | - | - | - | - | - |
| 7 | Раздел 7 Микропроцессорный комплект КР580 | 14 | 2 | **-** | 2 | - | 4 | **-** | 6 | №7 | - | 19 | №7 | - |
| - |
| 7.1 | Архитектура микропроцессорной системы | 5 | 1 |  | 1 | - | 2 | - | 1 | - | - |
| - |
| 7.2 | Организация работы ЦПЭ КР580ВМ80 | 9 | 1 | **-** | 1 | - | 2 | - | 5 | - | - |
| 8 | Раздел 8. Аппаратные средства МПК КР580 | 14 | 2 | **-** | - | - |  | **-** | 12 | №8 | - | - | - | - |
| 8.1 | Системный контроллер, шинные формирователи | 7 | 1 | **-** | - | - |  | - | 6 | - | - | - |  | - |
| 8.2 | Параллельный и последовательный интерфейсы, таймер | 7 | 1 | **-** | - | - |  | - | 6 | - | - | - |  | - |
| 9 | Раздел 9 Программное обеспечение МПК КР580 | 14 | 3 | **-** | - | - | 2 | - | 9 | №9 | - |  | №8 | - |
| 9.1 | Программная модель МП системы | 7 | 2 | **-** | - | - | 1 | - | 4 | - | - | - | - |
| - |
| 9.2 | Способы адресации в МП системе | 7 | 1 | **-** | - | - | 1 | - | 5 | - | - | - |
| 10 | Раздел 10. Система команд МПК КР580 | 16 | 8 | - | - | **-** | 6 | - | 2 | №10 | - | - | №9 | - |
| - |
| 10.1 | Команды пересылки данных и арифметико-логические команды | 8 | 5 | - | - | **-** | 3 | - | - | - | - |  |
| 10.2 | Команды передачи управления, ввода-вывода и др. | 8 | 3 | - | - | **-** | 3 | - | 2 | - | - |  |
| 11 | Раздел 11 Программирование в МП системе | 14 | 2 | - | 2 | **-** | - | - | 10 | №11 | - | 24 |  | КП |
|  |
| 11.1 | Программирование на машинном языке | 7 | 1 | - | 1 | **-** | - | - | 5 | - | - |
| 11.2 | Программирование на ассемблере | 7 | 1 | - | 1 | **-** | - | - | 5 | - | - |
| 12 | Раздел 12 Микроконтроллер КМ1816ВЕ48 | 14 | 2 | - | - | - | 4 | - | 8 | №12 | - | - | №10 |  |
|  |
| 12.1 | Аппаратное обеспечение КМ1816ВЕ48 | 7 | - | - | - | - | 2 | - | 5 |  | - | - |
| 12.2 | Программное обеспечение КМ1816ВЕ48 | 7 | 2 | - | - | - | 2 | - | 3 |  | - | - |
|  | Заключение | 1 | 1 |  |  |  |  |  |  |  |  |  |

очно-заочная форма обучения

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № п/п | Наименование  раздела  (темы) | Кол-во час. по дн. фор-ме обучения | Виды занятий и контроля | | | | | | | | | | | |
| Лекции | | ПЗ(С) | | ЛР | | Самос  .  работа  . | Те  с  ты | Контр  .  работы | ПЗ | ЛР | КП |
| аудит. | ДОТ | аудит. | ДОТ | аудит. | ДОТ |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  | ВСЕГО | 170 | 12 | **14** | 4 | **39** | 24 | **7** | 70 | - | 1 | - | - | 1 |
|  | Введение | 2 | 2 |  |  |  |  |  |  |  |  |  |  |  |
| 1 | Раздел 1. Логические схемы | 10 | - | - | - | **8** | - | - | 2 | №1 | - | - | - | - |
| 1.1 | Алгебра логики | 5 | - | - | - | **4** | - | - | 1 | - | - | №1,2 | - | - |
| 1.2 | Построение логических схем | 5 | - | - | - | **4** | - | - | 1 | - |  | 3,4 | - |  |
| 2 | Раздел 2. Основы цифровой техники | 12 | 2 | 2 | - | **2** | - | **-** | 6 | №2 | КР |  |  |  |
| 2.1 | Системы счисления | 6 | 1 | 1 | - | **1** | - | - | 3 | - | 5 |  |  |
| 2.2 | Формы представления чисел | 6 | 1 | 1 | - | **1** | - | - | 3 | - | 6 |  |  |
| 3 | Раздел 3 Комбинационные устройства | 19 | - | **-** | - | **7** | 6 | **3** | 3 | №3 | - | - | - | - |
| 3.1 | Полусумматоры, сумматоры, устройства неравнозначности и равнозначности | 7 | - | **-** | - | **2** | 2 | **2** | 1 | - | - | 7,8,9,10 | №1 | - |
| 3.2 | Шифраторы, дешифраторы и кодопреобразователи | 6 | - | - | - | **2** | 2 | **1** | 1 | - | - | 11 | №2 | - |
| 3.3 | Мультиплексоры, демультиплексоры и компараторы | 6 | - | - |  | **3** | 2 |  | 1 |  |  | 12 | №3 |  |
| 4 | Раздел 4. Цифровые автоматы | 18 | 2 | **1** | - | **6** | 6 | **3** | - | №4 | - | - | - | - |
| 4.1 | Асинхронные триггеры | 6 | 1 | **-** | - | **2** | 2 | **1** | - | - | - | 13 | №4 | - |
| 4.2 | Синхронные триггеры | 6 | 1 | **-** | - | **2** | 2 | **1** | - |  | - | 14 | №5 |  |
| 4.3 | Регистры и счетчики | 6 | - | **1** | - | **2** | 2 | **1** |  | - | - | 15 | №6 | - |
| 5 | Раздел 5. Аналого-цифровые и цифро-аналоговые преобразователи | 12 | - | **2** | - | **4** | - | **-** | 6 | №5 | - | - | - | - |
| 5.1 | Аналого-цифровые преобразователи | 6 | - | **1** | - | **2** | - | **-** | 3 | - | - | 16 | - | - |
| 5.2 | Цифроаналоговые преобразователи | 6 | - | **1** | - | **2** | - | **-** | 3 | - | - | 17 | - | - |
| 6 | Раздел 6 Запоминающие устройства | 10 | - | 1 | - | - | - | **-** | 9 | №6 | - | - | - | - |
| 6.1 | Оперативные запоминающие устройства | 5 | - | 1 | - | - | - | **-** | 4 | - | - |  | - | - |
| 6.2 | Постоянные запоминающие устройства | 5 | - | - | - | - | - | **-** | 5 | - | - | - | - | - |
| 7 | Раздел 7 Микропроцессорный комплект КР580 | 14 | 2 | **1** | 2 | - | 4 | **1** | 4 | №7 | - | 19 | №7 | - |
| 7.1 | Архитектура микропроцессорной системы | 5 | 1 | - | 1 | - | 2 | 1 | - | - | - |
| 7.2 | Организация работы ЦПЭ КР580ВМ80 | 9 | 1 | **1** | 1 | - | 2 | - | 4 | - | - |
| 8 | Раздел 8. Аппаратные средства МПК КР580 | 14 | - | **4** | - | - | - | **-** | 10 | №8 | - | - | - | - |
| 8.1 | Системный контроллер, шинные формирователи | 7 | - | **2** | - | - |  | - | 5 | - | - | - | - | - |
| 8.2 | Параллельный и последовательный интерфейсы, таймер | 7 | - | **2** | - | - | - | - | 5 | - | - | - | - | - |
| 9 | Раздел 9. Программное обеспечение МПК КР580 | 14 | - | **3** | - | - | 4 | - | 7 | №9 | - |  | №8 |  |
|  |
| 9.1 | Программная модель МП системы | 7 | - | **2** | - | - | 2 | - | 3 | - | - |
| 9.2 | Способы адресации в МП системе | 7 | - | **1** | - | - | 2 | - | 4 | - | - |
| 10 | Раздел 10. Система команд МПК КР580 | 16 | 2 | - | - | **8** | - | - | 6 | №  10 | - | - | - | - |
| 10.1 | Команды пересылки данных и арифметико-логические команды | 8 | 1 | - | - | **6** | - | - | 1 | - | - | 21, 22 | - | - |
| 10.2 | Команды передачи управления, ввода-вывода и др. | 8 | 1 | - | - | **2** | - | - | 5 | - | - | 23 | - | - |
| 11 | Раздел 11 Программирование в МП системе | 14 | 2 | - | - | **4** | - | - | 8 | №  11 | - | 24 | - | -  КП |
| -  - |
| 11.1 | Программирование на машинном языке | 7 | 1 | - | - | **2** | - | - | 4 | - | - |
| 11.2 | Программирование на ассемблере | 7 | 1 | - | - | **2** | - | - | 4 | - | - |  |
| 12 | Раздел 12 Микроконтроллер КМ1816ВЕ48 | 14 | - | - | 2 | - | 4 | - | 8 | №  12 | - | 25 | №10 | - |
| 12.1 | Аппаратное обеспечение КМ1816ВЕ48 | 7 | - | - | 1 | - | 2 | - | 4 |  | - | - |
| - |
| 12.2 | Программное обеспечение КМ1816ВЕ48 | 7 | - | - | 1 | - | 2 | - | 4 |  | - |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Заключение | 1 |  |  |  |  |  |  | 1 |  |  |  |  |  |

заочная форма обучения

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № п/п | Наименование  раздела  (темы) | Кол-во час. по дн. фор-ме обучения | Виды занятий и контроля | | | | | | | | | | | |
| Лекции | | ПЗ (С) | | ЛР | | Самост  .  раб.  . | Те  С  ты | Контр  .  раб | ПЗ | ЛР | КП |
| ауд. | ДОТ | аудит. | ДОТ | ауд. | ДОТ |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  | ВСЕГО | 170 | 8 | **16** | 4 | **51** | 12 | **9** | 70 | - | 1 | - | - | 1 |
|  | Введение | 2 | 2 |  |  |  |  |  |  |  |  |  |  |  |
| 1 | Раздел 1. Логические схемы | 10 | - | - | - | **8** | - | - | 2 | №1 | - | - | - | - |
| 1.1 | Алгебра логики | 5 | - | - | - | **4** | - | - | 1 | - | - | №1,2 | - | - |
| 1.2 | Построение логических схем | 5 | - | - | - | **4** | - | - | 1 | - | КР | 3,4 |  |  |
| 2 | Раздел 2. Основы цифровой техники | 12 | - | - | - | **8** | - | **-** | 4 | №2 |  |  |  |
| 5 | - | - |
| 2.1 | Системы счисления | 6 | - | - | - | **4** | - | - | 2 | - |
| 6 | - | - |
| 2.2 | Формы представле-  ния чисел | 6 | - | - | - | **4** | - | - | 2 | - |
| 3 | Раздел 3 Комбинационные устройства | 19 | - | **2** | - | **7** | 3 | **4** | 3 | №3 | - | - | - | - |
| 3.1 | Полусумматоры, сумматоры, устройс-тва неравнозначности и равнозначности | 7 | - | **2** | - | **2** | 1 | **2** | - | - | - | 7,8,9,10 | №1 | - |
| 3.2 | Шифраторы, дешиф-раторы и кодопреоб-разователи | 6 | - |  | - | **2** | 2 | **2** |  | - | - | 11 | №2 | - |
| 3.3 | Мультиплексоры, демультиплексоры и компараторы | 6 | - |  |  | **3** | - |  | 3 |  |  | 12 |  |  |
| 4 | Раздел 4. Цифровые автоматы | 18 | 2 | **4** | - | **6** | 3 | **3** | - | №4 | - | - | - | - |
| 4.1 | Асинхронные триггеры | 6 | 1 | **1** | - | **2** | 1 | **1** | - | - | - | 13 | №4 | - |
| 4.2 | Синхронные триггеры | 6 | 1 | **1** | - | **2** | 1 | **1** | - |  | - | 14 | №5 |  |
| 4.3 | Регистры и счетчики | 6 | - | **2** | - | **2** | 1 | **1** |  | - | - | 15 | №6 | - |
| 5 | Раздел 5. Аналого-цифровые и цифро-аналоговые преобразователи | 12 | - | **2** | - | **7** | - | **-** | 3 | №5 | - | - | - | - |
| 5.1 | Аналого-цифровые преобразователи | 6 | - | **1** | - | **4** | - | **-** | 1 | - | - | 16 | - | - |
| 5.2 | Цифроаналоговые преобразователи | 6 | - | **1** | - | **3** | - | **-** | 2 | - | - | 17 | - | - |
| 6 | Раздел 6 Запоминающие устройства | 10 | - | - | - | 1 | - | **-** | 9 | №6 | - | - | - | - |
| 6.1 | Оперативные запоминающие устройства | 5 | - | - | - | 1 | - | **-** | 4 | - | - | 18 | - | - |
| 6.2 | Постоянные запоминающие устройства | 5 | - | - | - | - | - | **-** | 5 | - | - | - | - | - |
| 7 | Раздел 7 Микропроцессорный комплект КР580 | 14 | 2 | **1** | 2 | - | - | **2** | 7 | №7 | - | 19 | - | - |
| №7 | - |
| 7.1 | Архитектура микропроцессорной системы | 5 | 1 | - | 1 | - | - | 2 | 1 | - | - |
| - | - |
| 7.2 | Организация работы ЦПЭ КР580ВМ80 | 9 | 1 | **1** | 1 | - | - | - | 6 | - | - |
| 8 | Раздел 8. Аппаратные средства МПК КР580 | 14 | - | **4** | - | - | - | **-** | 10 | №8 | - | - | - | - |
| 8.1 | Системный контроллер, шинные формирователи | 7 | - | **2** | - | - |  | - | 5 | - | - | - | - | - |
| 8.2 | Параллельный и последовательный интерфейсы, таймер | 7 | - | **2** | - | - | - | - | 5 | - | - | - | - | - |
| 9 | Раздел 9 Программное обеспечение МПК КР580 | 14 | - | **3** | - | - | 3 | - | 8 | №9 | - |  | №8 | - |
|  |
| 9.1 | Программная модель МП системы | 7 | - | **2** | - | - | 1 | - | 4 | - | - |
| 9.2 | Способы адресации в МП системе | 7 | - | **1** | - | - | 2 | - | 4 | - | - |
| 10 | Раздел 10. Система команд МПК КР580 | 16 | 2 | - | - | **8** | - | - | 6 | №10 | - | - | - | - |
| 10.1 | Команды пересылки данных и рифметико-логические команды | 8 | 1 | - | - | **6** | - | - | 1 | - | - | 21 22 | - | - |
| 10.2 | Команды передачи управления, ввода-вывода и др. | 8 | 1 | - | - | **2** | - | - | 5 | - | - | 23 | - | - |
| 11 | Раздел 11 Программирование в МП системе | 14 | - | - | - | **6** | - | - | 8 | №11 | - | 24 | - | КП |
| 11.1 | Программирование на машинном языке | 7 | - | - | - | **3** | - | - | 4 | - | - |
| 11.2 | Программирование на ассемблере | 7 | - | - | - | **3** | - | - | 4 | - | - |
| 12 | Раздел 12 Микроконтроллер КМ1816ВЕ48 | 14 | - | - | 2 | - | 3 | - | 9 | №12 | - | 25 | №10 | - |
| 12.1 | Аппаратное обеспечение КМ1816 ВЕ48 | 7 | - | - | 1 | - | 1 | - | 5 |  | - | - |
|  |
| 12.2 | Программное обеспечение КМ1816 ВЕ48 | 7 | - | - | 1 | - | 2 | - | 4 |  | - |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | - |
|  | Заключение | 1 |  |  |  |  |  |  | 1 |  |  |

**2.3. Cтруктурно-логическая схема дисциплины**

Цифровые устройства

**Раздел 1**

Логические схемы

**Раздел 3**

Комбинационные устройства

**Раздел 5**

Аналого-цифро-вые и цифро-аналоговые преобразователи

Алгебра логики

Построе-ние логи- ческих схем

**Раздел 2**

Основы цифровой техники

Системы счисления

Формы представ-ления чисел

Полусумматоры, сумматоры, устройства неравнозначности и равнозначности

Шифраторы, дешифраторы и кодопреобразова-тели

**Раздел 4**

Цифровые автоматы

АЦП

ЦАП

**Раздел 6**

Запоминающие устройства

Синхронные триггеры

Асинхронные триггеры

Оперативные запоминающие устройства

Мультиплексоры, демультиплексоры и компараторы

Регистры и счетчики

Постоянные запоминающие устройства

**2.3. Структурно-логическая схема дисциплины**

Микропроцессоры

**Раздел 7**

Микропро-цессорный комплект КР580

**Раздел 9**

Программное обеспечение МПК КР580

**Раздел 11**

Программи-рование в МП системе

Архитектура МП системы

Организация работы ЦПЭ КР580ВМ80

**Раздел 8**

Аппаратные средства МПК КР580

Системный контроллер, шинные формирователи

Параллельный и последовательный интерфейсы, таймер

Программная модель МП системы

Способы адресации в МП системе

**Раздел 10**

Система команд МПК КР580

Программирова-ние на машинном языке

Программирова-ние на ассемблере

**Раздел 12**

Микроконтроллер КМ1816ВЕ48

Команды пере-сылки данных и арифметико-логи-ческие команды

Команды передачи управления, ввода-вывода и другие

Аппаратное обеспечение КМ1816 ВЕ48

Программное обеспечение КМ1816 ВЕ48

**2.4. Временной график изучения дисциплины при использовании информационно-коммуникационных технологий**

|  |  |  |
| --- | --- | --- |
| **№** | Название раздела (темы) | Продолжительность  изучения раздела (темы)  (из расчета – 4 часа в день) |
| 1 | Введение. Раздел 1. Логические схемы | 3дн. |
| 2 | Раздел 2. Основы цифровой техники | 3дн. |
| 3 | Раздел 3. Комбинационные устройства | 5дн. |
| 4 | Раздел 4. Цифровые автоматы | 4,5дн |
| 5 | Раздел 5. АЦП и ЦАП | 3дн |
| 6 | Раздел 6. Запоминающие устройства | 2,5дн |
| 7 | Раздел 7. Микропроцессорный комплект КР580 | 3,5дн |
| 8 | Раздел 8. Аппаратные средства МПК КР580 | 3,5дн |
| 9 | Раздел 9. Программное обеспечение МПК КР580 | 3,5дн |
| 10 | Раздел 10. Система команд МПК КР580 | 4дн |
| 11 | Раздел 11. Программирование в МП системе | 3,5дн |
| 12 | Раздел 12. Микроконтроллер КМ1816ВЕ48  Заключение | 3,5дн |
|  | ИТОГО  В том числе курсовой проект,  контрольная работа | 42,5дн.  12 дн.  2 дн. |

**2.5. Практический блок**

2.5.1. Практические занятия

очная форма обучения

|  |  |  |  |
| --- | --- | --- | --- |
| Номер и наименование  темы | Наименования тем практических занятий | Кол-во часов  ауд. | Кол-во часов  ДОТ |
| 1.1. Алгебра логики | Занятие № 1 по теме «Алгебра логики» |  | 1 |
| 1.2. Построение логических схем | Занятие № 3 по теме «Построение логических схем» |  | 44  2 |
| 2.1. Системы счисления | Занятие № 5 по теме «Системы счисления» | 1 | 1 |
| 2.2. Формы представления чисел | Занятие № 6 по теме «Формы представления чисел» | 1 | 1 |
| 3.1. Полусумматоры, сумматоры, устройства неравнозначности и равнозначности | Занятия № 7…10 по теме «Полусумматоры, сумматоры, устройства неравнозначности и равнозначности» |  | 2 |
| 3.2. Шифраторы, дешифраторы и кодопреобразователи | Занятие № 11 по теме «Шифраторы, дешифраторы, кодопреобразователи» | 1 | 1 |
| 3.3. Мультиплексоры, демультиплексоры и компараторы | Занятие № 12 по теме «Мультиплексоры, демультиплексоры и компараторы» | 1 | 1 |
| 4.1. Асинхронные триггеры | Занятие № 13 по теме «Асинхронные триггеры» |  | 1 |
| 4.2. Синхронные триггеры | Занятие № 14 по теме «Синхронные триггеры» |  | 1 |
| 4.3. Регистры и счетчики | Занятие № 15 по теме «Регистры и счетчики» |  | 1 |
| 5.1. Аналого-цифровые преобразователи | Занятие № 16 по теме «Аналого-цифровые преобразователи» |  | 2 |
| 5.2. Цифроаналоговые преобразователи | Занятие № 17 по теме «Цифроаналоговые преобразователи» |  | 2 |
| Раздел 7 Микропроцессорный комплект КР580  7.1. Архитектура микро-  процессорной системы 7.2. Организация рабо-  ты ЦПЭ КР580ВМ80 | Занятие № 19 по теме «Микропроцессорный комплект КР580» | 1  1 |  |
| Раздел 11 Программирование в МП системе  11.1 Программирование на машинном языке  11.2. Программирование на ассемблере | Занятие № 24 по теме «Программирование в МП системе» | 1  1 |  |
|  | Всего часов | 8 | 16 |

очно-заочная форма обучения

|  |  |  |  |
| --- | --- | --- | --- |
| Номер и наименование  темы | Наименования тем практических занятий | Кол-во часов  ауд. | Кол-во часов  ДОТ |
| 1.1. Алгебра логики | Занятие № 1 по теме «Алгебра логики»  Занятие № 2 по теме «Алгебра логики» |  | 4 |
| 1.2. Построение логических схем | Занятие № 3 по теме «Построение логических схем»  Занятие № 4 по теме «Построение логических схем» |  | 44  4 |
| 2.1. Системы счисления | Занятие № 5 по теме «Системы счисления» |  | 1 |
| 2.2. Формы представления чисел | Занятие № 6 по теме «Формы представления чисел» |  | 1 |
| 3.1. Полусумматоры, сумматоры, устройства неравнозначности и равнозначности | Занятия № 7…10 по теме  «Полусумматоры, сумматоры, устройства неравнозначности и равнозначности» |  | 2 |
| 3.2. Шифраторы, дешифраторы и кодопреобразователи | Занятие № 11 по теме «Шифраторы, дешифраторы, кодопреобразователи» |  | 2 |
| 3.3. Мультиплексоры, демультиплексоры и компараторы | Занятие № 12 по теме «Мультиплексоры, демультиплексоры, компараторы» |  | 3 |
| 4.1. Асинхронные триггеры | Занятие № 13 по теме «Асинхронные триггеры» |  | 2 |
| 4.2. Синхронные триггеры | Занятие № 14 по теме «Синхронные триггеры» |  | 2 |
| 4.3. Регистры и счетчики | Занятие № 15 по теме «Регистры и счетчики» |  | 2 |
| 5.1. Аналого-цифровые преобразователи | Занятие № 16 по теме «Аналого-цифровые преобразователи» |  | 2 |
| 5.2. Цифроаналоговые преобразователи | Занятие № 17 по теме «Цифроаналоговые преобразователи» |  | 2 |
| Раздел 7 Микропроцессорный комплект КР580  7.1. Архитектура микро-  процессорной системы 7.2. Организация рабо-  ты ЦПЭ КР580ВМ80 | Занятия № 19 по теме «Микропроцессорный комплект КР580» | 1  1 |  |
| 10.1. Команды пересылки данных и арифметико-логические команды | Занятие № 21 по теме «Команды пересылки данных и арифметико-логические команды»  Занятие № 22 по теме «Команды пересылки данных и арифметико-логические команды» |  | 6 |
| 10.2. Команды передачи управления, ввода-вывода и др. | Занятие № 23 по теме «Команды передачи управления, ввода-вывода и др.» |  | 2 |
| Раздел 11 Программирование в МП системе  11.1 Программирование на машинном языке  11.2. Программирование на ассемблере | Занятие № 24 по теме «Программирование в МП системе» |  | 2  2 |
| Раздел 12 Микроконтроллер КМ1816ВЕ48  12.1 Аппаратное обес-  печение КМ1816ВЕ48  12.2. Программное обе-  спечение КМ1816ВЕ48 | Занятие № 25 по теме «Микроконтроллер КМ1816ВЕ48» | 1  1 |  |
|  | Всего часов | 4 | 39 |

заочная форма обучения

|  |  |  |  |
| --- | --- | --- | --- |
| Номер и наименование  темы | Наименования тем практических занятий | Кол-во часов  ауд. | Кол-во часов  ДОТ |
| 1.1. Алгебра логики | Занятие № 1 по теме «Алгебра логики»  Занятие № 2 по теме «Алгебра логики» |  | 4 |
| 1.2. Построение логических схем | Занятие № 3 по теме «Построение логических схем»  Занятие № 4 по теме «Построение логических схем» |  | 44  4 |
| 2.1. Системы счисления | Занятие № 5 по теме «Системы счисления» |  | 4 |
| 2.2. Формы представления чисел | Занятие № 6 по теме «Формы представления чисел» |  | 4 |
| 3.1. Полусумматоры, сумматоры, устройства неравнозначности и равнозначности | Занятия № 7…10 по теме «Полусумматоры, сумматоры, устройства неравнозначности и равнозначности» |  | 2 |
| 3.2. Шифраторы, дешифраторы и кодопреобразователи | Занятие № 11 по теме «Шифраторы, дешифраторы, кодопреобразователи» |  | 2 |
| 3.3. Мультиплексоры, демультиплексоры и компараторы | Занятие № 12 по теме «Мультиплексоры, демультиплексоры, компараторы» |  | 3 |
| 4.1. Асинхронные триггеры | Занятие № 13 по теме «Асинхронные триггеры» |  | 2 |
| 4.2. Синхронные триггеры | Занятие № 14 по теме «Синхронные триггеры» |  | 2 |
| 4.3. Регистры и счетчики | Занятие № 15 по теме «Регистры и счетчики» |  | 2 |
| 5.1. Аналого-цифровые преобразователи | Занятие № 16 по теме «Аналого-цифровые преобразователи» |  | 4 |
| 5.2. Цифроаналоговые преобразователи | Занятие № 17 по теме «Цифроаналоговые преобразователи» |  | 3 |
| 6.1. Оперативные запоминающие устройства | Занятие № 18 по теме «Оперативные запоминающие устройства» |  | 1 |
| Раздел 7 Микропроцессорный комплект КР580  7.1. Архитектура микро-  процессорной системы 7.2. Организация рабо-  ты ЦПЭ КР580ВМ80 | Занятие № 19 по теме «Микропроцессорный комплект КР580» | 1  1 |  |
| 10.1. Команды пересылки данных и арифметико-логические команды | Занятие № 21 по теме «Команды пересылки данных и арифметико-логические команды»  Занятие № 22 по теме «Команды пересылки данных и арифметико-логические команды» |  | 6 |
| 10.2. Команды передачи управления, ввода-вывода и др. | Занятие № 23 по теме «Команды передачи управления, ввода-вывода и др.» |  | 2 |
| Раздел 11 Программирование в МП системе  11.1 Программирование на машинном языке  11.2. Программирование на ассемблере | Занятие № 24 по теме «Программирование в МП системе» |  | 3  3 |
| Раздел 12. Микроконт- роллер КМ1816ВЕ48  12.1 Аппаратное обес-  печение КМ1816ВЕ48  12.2. Программное обе-  спечение КМ1816ВЕ48 | Занятие № 25 по теме  «Микроконтроллер КМ1816ВЕ48» | 1  1 |  |
|  | Всего часов | 4 | 51 |

2.5.2 Лабораторные работы

очная форма обучения

|  |  |  |  |
| --- | --- | --- | --- |
| Номер и наименование темы | Наименование лабораторнойработы | Кол-вочасов,ауд | Кол-вочасов,ДОТ |
| 3.1. Полусумматоры, сумматоры, устройства неравнозначности и равнозначности | Работа 1. Исследование комбинационных устройств (на стенде и ПК) | 1 |  |
| 3.2. Шифраторы, дешифраторы и кодопреобразователи | Работа 2. Исследование шифраторов, дешифраторов, кодопреобразователей на ПК | 2 |  |
| 4.1.Асинхронные триггеры | Работа 4. Исследование асинхронных триггеров (на стенде и ПК) | 3 |  |
| 4.2. Синхронные триггеры | Работа 5. Исследование синхронных триггеров на ПК | 3 |  |
| 4.3. Регистры и счетчики | Работа 6. Исследование регистров и счетчиков (на стенде и ПК) | 3 |  |
| 7. Микропроцессор-  ный комплект КР580  7.1. Архитектура микропроцессорной системы  7.2. Организация ра-  боты ЦПЭ КР580ВМ80 | Работа 7. Исследование работы контроллера К1-20 в режиме монитора | 2  2 |  |
| 9. Программное обеспечение МПК КР580  9.1. Программная мо-  дель МП системы  9.2. Способы адреса-  ции в МП системе | Работа 8. Исследование контроллера К1-20 при выполнении простых программ | 1  1 |  |
| 10. Система команд МПК КР580  10.1. Команды пере-  сылки данных и ари-  фметико-логические  команды  10.2. Команды пере-  дачи управления,  ввода-вывода и др. | Работа 9. Исследование записи и выполнения программ в микропроцессорной системе на базе микроЭВМ УМПК 80 | 3  3 |  |
| 12. Микроконтроллер КМ1816ВЕ48  12.1 Аппаратное обе- спечение М1816ВЕ48  12.2. Программное обеспечение КМ1816ВЕ48 | Работа10. Исследование програм-мирования на учебной микроЭВМ УМПК 48 | 2  2 |  |
|  | |  |  |
| ИТОГО | | 28 | - |

очно-заочная форма обучения

|  |  |  |  |
| --- | --- | --- | --- |
| Номер и наименование раздела темы | Наименование лабораторнойработы | Кол-вочасов,ауд. | Кол-вочасов,ДОТ |
| 3.1. Полусумматоры, сумматоры, устройс-тва равнозначности, устройства неравно-значности | Работа 1. Исследование комбина-ционных устройств (на стенде и ПК) | 2 | 2 |
| 3.2. Шифраторы, де-шифраторы и кодо-преобразователи | Работа 2. Исследование шифрато-ров, дешифраторов, кодопреобра-зователей на ПК | 2 | 1 |
| 3.3. Мультиплексоры, демультиплексоры и компараторы | Работа 3. Исследование мульти-плексоров, демультиплексоров, компараторов на ПК | 2 |  |
| 4.1.Асинхронные триггеры | Работа 4. Исследование асинхрон-ных триггеров (на стенде и ПК) | 2 | 1 |
| 4.2. Синхронные триггеры | Работа 5. Исследование синхрон-ных триггеров на ПК | 2 | 1 |
| 4.3. Регистры и счетчики | Работа 6. Исследование регистров и счетчиков (на стенде и ПК) | 2 | 1 |
| 7. Микропроцессор-ный комплект КР580  7.1. Архитектура  микропроцессорной системы  7.2. Организация ра-  боты ЦПЭ КР580ВМ80 | Работа 7. Исследование работы контроллера К1-20 в режиме монитора | 2  2 | 1 |
| 9. Программное обеспечение МПК КР580  9.1. Программная мо-  дель МП системы  9.2. Способы адреса-  ции в МП системе | Работа 8. Исследование контрол-лера К1-20 при выполнении про-стых программ | 2  2 |  |
| 12. Микроконтроллер КМ1816ВЕ48  12.1 Аппаратное обе- спечение М1816ВЕ48  12.2. Программное обеспечение КМ1816ВЕ48 | Работа 10. Исследование про-грамммирования на учебной микроЭВМ УМПК 48 | 2  2 |  |
| ИТОГО | | 24 | 7 |

заочная форма обучения

|  |  |  |  |
| --- | --- | --- | --- |
| Номер и наименование раздела темы | Наименование лабораторнойработы | Кол-вочасов,ауд. | Кол-вочасов,ДОТ |
| 3.1.Полусумматоры, сумматоры, устрой-ства неравнозначно-сти и равнозначности | Работа 1. Исследование комбина-ционных устройств (на стенде и ПК) | 1 | 2 |
| 3.2. Шифраторы, дешифраторы и кодо-преобразователи | Работа 2. Исследование шифрато-ров, дешифраторов, кодопреобра-зователей на ПК | 2 | 2 |
| 4.1.Асинхронные триггеры | Работа 4. Исследование асинхрон-ных триггеров (на стенде и ПК) | 1 | 1 |
| 4.2. Синхронные триггеры | Работа 5. Исследование синхрон-ных триггеров на ПК | 1 | 1 |
| 4.3. Регистры и счет-чики | Работа 6. Исследование регистров и счетчиков (на стенде и ПК) | 1 | 1 |
| 7.1. Архитектура микропроцессорной системы | Работа 7. Исследование работы контроллера К1-20 в режиме монитора |  | 2 |
| 9. Программное обеспечение МПК КР580  9.1. Программная мо-  дель МП системы  9.2. Способы адреса-  ции в МП системе | Работа 8. Исследование контрол-лера К1-20 при выполнении про-стых программ | 1  2 |  |
| 12. Микроконтроллер КМ1816ВЕ48  12.1 Аппаратное обе- спечение М1816ВЕ48  12.2. Программное обеспечение КМ1816ВЕ48 | Работа 10. Исследование про-граммирования на учебной микроЭВМ УМПК 48 | 1  2 |  |
| ИТОГО | | 12 | 9 |

**2.6. Балльно-рейтинговая система оценки знаний**

Условием успешного завершения изучения дисциплины является выполнение лабораторных работ и практических занятий, предусмотренных учебным планом. Оценка знаний производится по результатам контрольных мероприятий: тестирования; сдачи контрольной работы, теоретического зачета, курсового проекта и экзамена.

Изучение дисциплины происходит в двух семестрах. В первом семестре изучаются разделы 1…6. Контрольным итогом является сдача зачёта.

Тестирование проводится после изучения каждого раздела. Тест по каждому разделу содержит 5 вопросов, правильные ответы на каждый вопрос теста оцениваются в 1 балл. Повторно тесты не даются.

Итак, максимальная оценка в баллах за правильные ответы:

-по практическим занятиям 16;

-по лабораторным работам 6;

-за контрольную работу 10;

- тестирование по разделам 1…6 дисциплины 6\*5 = 30;

- ответы на два вопроса билета на зачёте 15\*2=30;

- за проявленную эрудицию добавляются 8 баллов.

Итого, максимально возможное количество баллов составляет 100.

Для получения зачета необходимо набрать не менее 60 баллов.

Во втором семестре изучаются разделы 7…12. Контрольным итогом является сдача экзамена.

Билеты экзамена содержат два вопроса по разделам 7...12. Каждый правильно освещенный вопрос билета оценивается в 15 баллов, при недочетах балл может снижаться. Максимальное количество баллов составляет 15\*2=30. Билет можно получить повторно (другого содержания), он оценивается на 5 баллов ниже.

Итак, максимальная оценка в баллах за правильные ответы:

-по практическим занятиям 12;

-по лабораторным работам 12;

-по курсовому проекту 10;

- тестирование по разделам дисциплины 6\*5= 30;

- ответы на два вопроса билета 15\*2=30;

- за проявленную эрудицию добавляются 6 баллов.

Итого максимально возможное количество баллов составляет 100.

**Ранжирование результатов:**

|  |  |  |  |
| --- | --- | --- | --- |
| Оценка | Удовлетвори-тельно | Хорошо | Отлично |
| Кол-во набранных баллов | 60-74 | 75-89 | 90-100 |

Для получения удовлетворительной оценки необходимо набрать не менее 60 баллов.

**3. Информационные ресурсы дисциплины**

**3.1. Библиографический список**

Основной:

1. Соколов, О. Л. Цифровые устройства и микропроцессоры: учеб. пособие /О. Л. Соколов, О. С. Голод, - СПб.: Изд-во СЗТУ, 2010. - 129 с.

2. Антонов, О. Г. Цифровые устройства и микропроцессоры. ч. 1. Цифровые устройства: учеб. пособие/ О. Г. Антонов, Е. Ю. Мигунова. – СПб.: Изд-во СЗТУ, 2006. – 83 с.

3. Антонов, О. Г. Цифровые устройства и микропроцессоры. ч. 2. Микропроцессоры: учеб. пособие/ О. Г. Антонов, Е. Ю. Мигунова. – СПб.: Изд-во СЗТУ, 2007. – 85 с.

4. Цифровые устройства и микропроцессоры: методические указания к выполнению лабораторных работ/ О. Л. Соколов, О. С. Голод, - СПб.: Изд-во СЗТУ, 2010. - 71 с.

5. Цифровые устройства и микропроцессоры: рабочая программа, задание на коитрольную работу, методические указания к выполнению контрольной работы, задание на курсовой проект, методические указания к выполнению курсового проекта/ О. Г. Антонов, Е. Ю. Мигунова. – СПб.: Изд-во СЗТУ, 2005. – 51 с.

Дополнительный:

6. Драгунов, В. П. Основы наноэлектроники: учеб. пособие для вузов/ В. П. Драгунов, И. Г. Неизвестный, В. А. Гридчин. – М.: Физматгиз, 2006. – 537 с.

7. Основы цифровой обработки сигналов: учеб. пособие 2-е изд / А. И. Солонин, [ и др.] — СПб.: БХВ-Петербург, 2005. — 753 с.

8. Калабеков, Б. А. Цифровые устройства и микропроцессорные системы: учебник для сред. учеб. заведений/ Б. А. Калабеков. – М.: Горячая линия – Телеком, 2002. – 336 с.

9. Григорьев, В. Л.Программное обеспечение микропроцессорных систем/ В. Л.Григорьев. – М.: Энергоатомиздат, 1983. – 208 с.

10. Интегральные микросхемы**: с**правочник под ред. Б. В. Тарабрина. – М.: Энергоатомиздат, 1984. – 528 с.

Средства обеспечения освоения дисциплины (ресурсы Internet)

11. [http://de/ifo.ru/bk\_netra/pages.php/?tutindex=25&](http://de/ifo.ru/bk_netra/pages.php/?tutindek=25&) index=102

12. <http://krauss/dp/ua/lood/4-1-0-1>

13. [http://krauss/dp/ua/lood/4-1-0-2](http://krauss/dp/ua/lood/4-1-0-1)

**3.2. Опорный конспект (сценарий учебного процесса)**

**Введение**

Дисциплина «Цифровые устройства и микропроцессоры» является общепрофессиональной, базирующейся на дисциплинах: математика, физика, основы теории цепей. Теоретическим фундаментом дисциплины является раздел математики – алгебра логики.

В связи с этим, приступая к её изучению, необходимо восстановить в памяти основные сведения из указанных дисциплин.

Изучение дисциплины происходит в двух семестрах. В первом семестре изучаются разделы 1…6.

Во втором семестре изучаются разделы 7…12.

Основной целью дисциплины является изучение принципов построения современных цифровых устройств и микропроцессоров, теоретических основ их анализа, синтеза и исследования.

Методика и последовательность изучения дисциплины соответствуют перечню тематического плана. Материал каждой темы содержит необходимые математические соотношения, структурные и принципиальные схемы ци-фровых устройств и микропроцессоров, а также временные диаграммы.

**Раздел 1. Логические схемы**

1.1. Алгебра логики

Основные понятия и определения изложены в учебных пособиях [1, 2].

В разделе рассматриваются основные теоремы, аксиомы и тождества, используемые для упрощения логических выражений. Принцип двойственности. Теорема де Моргана.

Способы представления функций: словесное описание, таблица истинности, алгебраическое выражение (структурная формула). Переходы от структурной формулы к логической схеме и обратно. Логические базисы, реализующие функционально полную систему. Логические функции: отри-цание дизъюнкции и отрицание конъюнкции. Понятие об универсальных логических элементах.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы, обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие №1 по теме «Алгебра логики**»**, а также выполнить практическое занятие №2 по теме «Алгебра логики**»,** если это предусмотрено Вашим тематическим планом.

1.2. Построение логических схем

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме рассматриваются совершенная дизъюнктивная нормальная форма (СДНФ) и совершенная конъюнктивная нормальная форма (СКНФ).

Основные методы минимизации (получение тупиковой формы). Алге-браический метод. Метод карт Карно-Вейча. Недоопределенные функции. Табличный метод Квайна-МакКласки.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы, обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие №3 по теме «Построение логических схем**»,** а также выполнить практическое занятие №4 по теме «Построение логических схем**»,** если это предусмотрено Вашим тематическим планом.

После проработки теоретического материала раздела 1 необходимо ответить на вопросы теста № 1.

Максимальное количество баллов, которые Вы можете получить после изучения раздела 1, составляет 5 баллов за тестирование и 4 балла за правильное выполнение практических занятий.

**Раздел 2. Основы цифровой техники**

2.1. Системы счисления

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме рассматриваются двоичная, восьмеричная, десятичная и 16-ричная системы счисления. Перевод чисел из одной системы счисления в другую. Двоично-десятичная система кодирования чисел. Наиболее распространенные виды двоично-десятичного кодирования: двоично-десятичный код 8-4-2-1, код Айкена (2-4-2-1), код «с избытком 3», код «два из пяти», код Джонсона и код Грея.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы, обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие №5 по теме «Системы счисления**».**

2.2.Формы представления чисел

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме рассматриваются числа с фиксированной точкой. Представление знака числа. Числа с плавающей точкой. Понятия об обратном и допол-нительном кодах.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы; обратиться к практикуму; разобрать примеры решения задач, выполнить практическое занятие № 6 по теме «Формы представления чисел».

После проработки теоретического материала раздела 2 необходимо выполнить контрольную работу, если это предусмотрено Вашим тематическим планом, и ответить на вопросы теста № 2.

Максимальное количество баллов, которые Вы можете получить после изучения раздела 2, составляет 10 баллов за правильное выполнение контрольной работы, 5 баллов за тестирование и 1 балл за правильное выполнение практических занятий.

**Раздел 3. Комбинационные устройства**

3.1. Полусумматоры, сумматоры,

устройства неравнозначности и равнозначности

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме изучаются полусумматоры и полные сумматоры, устройство неравнозначности (сумма по модулю два) и его свойства. Устройство равнозначности. Устройство «Запрет».

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы, обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практические занятия №№ 7…10 по теме «Сумматоры, полусумматоры, устройства неравно-значности и равнозначности**».**

После этого следует проделать лабораторную работу №1, содержание которой приведено в «Методических указаниях к выполнению лабораторных работ» УМК[4].

3.2. Шифраторы, дешифраторы и кодопреобразователи

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме изучаются полные и неполные дешифраторы (декодеры). Шифраторы и кодопреобразователи.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы, обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие № 11 по теме «Шифраторы, дешифраторы и кодопреобразователи**».**

После этого следует проделать лабораторную работу №2, содержание которой приведено в «Методических указаниях к выполнению лабораторных работ» УМК [4].

3.3. Мультиплексоры, демультиплексоры и компараторы

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме изучаются мультиплексоры, демультиплексоры, их назначение, вопросы синтеза и каскадирования. Цифровые компараторы.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы; обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие № 12 по теме «Мультиплексоры, демультиплексоры и компараторы».

После проработки теоретического материала раздела 3 необходимо ответить на вопросы теста № 3.

После этого следует проделать лабораторную работу № 3, содержание которой приведено в «Методических указаниях к выполнению лабораторных работ» УМК [4], если это предусмотрено Вашим тематическим планом.

Максимальное количество баллов, которые Вы можете получить после изучения раздела 3, составляет 5 баллов за тестирование, 3 балла за правильное выполнение лабораторных работ и 5 баллов за правильное выполнение практических занятий.

**Раздел 4. Цифровые автоматы**

* 1. Асинхронные триггеры

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме рассматриваются потенциальные и импульсные сигналы. Операторы перехода. Основные тождества, связывающие потенциальные и импульсные сигналы. Асинхронные потенциальные триггеры RS-типа с инверсными и прямыми входами. Триггеры типа E и JK. Синтез асинхронных потенциальных триггеров. Функции возбуждения. Счетный режим. Т-триггер.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы, обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие № 13 по теме «Асинхронные триггеры**»**.

После этого следует проделать лабораторную работу № 4, содержание которой приведено в «Методических указаниях к выполнению лабораторных работ» УМК [4].

4.2. Синхронные триггеры

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме рассматриваются синхронные триггеры типа RSC, типа JK и типа D. MS-триггер. Функция возбуждения синхронных триггеров и общая методика их синтеза. Счетный режим в триггерах типа JK и D.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы; обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие № 14 по теме «Синхронные триггеры**».**

После этого следует проделать лабораторную работу № 5, содержание которой приведено в «Методических указаниях к выполнению лабораторных работ» УМК[4].

4.3. Регистры и счетчики

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме рассматриваются счетный режим в триггерах типа JK и D, сдвигающие регистры,классификация сдвигающих регистров, простые сдвигающие регистры, реверсивные сдвигающие регистры, универсальные сдвигающие регистры.

Счетчики.Двоичные и двоично-десятичные счетчики. Каскадирование счетчиков. Реверсивные счетчики, особенности их каскадирования. Счетчики на сдвигающих регистрах. Счетчики с произвольным модулем счета. Счетчики Джонсона.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы; обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие № 15 по теме «Регистры и счетчики**».**

После проработки теоретического материала раздела 4 необходимо ответить на вопросы теста № 4.

После этого следует проделать лабораторную работу № 6, содержание которой приведено в «Методических указаниях к выполнению лабораторных работ» УМК [4].

Максимальное количество баллов, которые Вы можете получить после изучения раздела 4, составляет 5 баллов за тестирование, 3 балла за правильное выполнение лабораторных работ и 3 балла за правильное выполнение практических занятий.

**Раздел 5. Аналого-цифровые и цифроаналоговые преобразователи**

5.1. Аналого-цифровые преобразователи (АЦП)

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме рассматриваются классификация АЦП по времени преобразования, АЦП параллельного типа, АЦП с промежуточным преобразованием напряжения во временной интервал, принципы конвейерной обработки в АЦП, точность работы АЦП и факторы, влияющие на точность преобразования.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы; обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие № 16 по теме «Аналого-цифровые преобразователи**».**

5.2. Цифроаналоговые преобразователи (ЦАП)

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме рассматриваются принципы работы ЦАП, использование матрицы R-2R, методы умножения аналоговых сигналов, получение сре-днеквадратичного значения сигнала.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы; обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие № 17 по теме «Цифроаналоговые преобразователи**».**

После проработки теоретического материала раздела 5 необходимо ответить на вопросы теста № 5.

Максимальное количество баллов, которые Вы можете получить после изучения раздела 5, составляет 5 баллов за тестирование и 2 балла за правильное выполнение практических занятий.

**Раздел 6. Запоминающие устройства**

6.1. Оперативные запоминающие устройства (ОЗУ)

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме рассматриваются основные параметры запоминающих устройств (ЗУ); адресация; информационная емкость; разрядность; единицы для выражения значений емкости ЗУ; быстродействие ЗУ; понятие о времени выборки и цикле записи; характеристика ЗУ по потребляемой мощности, набору питающих напряжений и времени хранения информации.

Типовая структура ОЗУ матричного вида. Управляющие цепи для обеспечения режима хранения, чтения и записи информации. Определение числа строк и столбцов матрицы элементов памяти. Условное обозначение микросхемы ОЗУ. Временные диаграммы сигналов. Схема наращивания разрядности. Динамические ОЗУ.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы; обратиться к практикуму; разобрать примеры решения задач, а затем выполнить практическое занятие № 18 по теме «Оперативные запоминающие устройства**»**, если это преду-смотрено Вашим тематическим планом.

6.2. Постоянные запоминающие устройства (ПЗУ)

Основные понятия и определения изложены в учебных пособиях [1, 2].

В теме рассматриваются классификация ПЗУ по способу занесения информации; ПЗУ, программируемые маской на предприятии-изготовителе; ПЗУ, программируемые пользователем; перепрограммируемые ПЗУ (ППЗУ); способы стирания информации.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы, а затем необходимо ответить на вопросы теста № 6.

Максимальное количество баллов, которые Вы можете получить после изучения раздела 6, составляет 5 баллов за тестирование и 1 балл за правильное выполнение практического занятия.

**П. МИКРОПРОЦЕССОРЫ**

**Раздел 7. Микропроцессорный комплект КР580**

7.1. Архитектура микропроцессорной системы

Устройства, назначение и сигналы микропроцессорной системы рассмотрены в учебных пособиях [1, 3].

В теме изучаются трехшинная архитектура микропроцессорной системы,структурная схема микропроцессорной системы с трехшинной архитектурой, ее основные узлы: центральный процессор, память и внешние устройства. Назначение шин адреса, данных и управления. Основные сигналы управления операциями ввода/вывода.

ОднокристальныеМП.Структурная схема однокристальных МП на примере МП КР580ВМ80. Основные узлы МП, буферы шин адреса и данных, регистры общего назначения (РОН), регистр команд, программный счетчик, схема синхронизации и управления, арифметико-логическое устройство (АЛУ), указатель стека.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

7.2. Организация работы ЦПЭ КР580ВМ80

Линии синхронизации, машинные циклы рассмотрены в учебных посо-биях [1, 3].

В теме изучаются генератор тактовых импульсов КР580ГФ24, струк-турная схема генератора, схема подключения генератора к центральному процессорному элементу, формирование сигнала СТРОБ СОСТОЯНИЯ, использование сигналов СБРОС и ГОТОВНОСТЬ, установка МП в режим ожидания на заданное время, обеспечение шагового режима.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

Выполнить практическое занятие № 19 по разделу «Микропроцессор-ный комплект КР580».

После проработки теоретического материала раздела 7 необходимо ответить на вопросы теста № 7.

По разделу следует проделать лабораторную работу № 7, если это предусмотрено Вашим тематическим планом. Содержание работы № 7 приведено в «Методических указаниях к выполнению лабораторных работ» УМК [4].

Максимальное количество баллов, которые Вы можете получить после изучения раздела 7, составляет 5 баллов за тестирование, 3 балла за правильное выполнение лабораторной работы и 2 балла за правильное выполнение практических занятий.

**Раздел 8. Аппаратные средства МПК КР580**

8.1. Системный контроллер, шинные формирователи

Системный контроллер КР580ВК28/38 рассмотрен в учебных пособиях [1, 3].

В теме изучаются структурная схема системного контроллера, его назначение и принцип действия.

Шинные формирователи КР580ВА86/87. Назначение, структурная схема и принцип действия.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

8.2. Параллельный и последовательный интерфейсы, таймер

Интерфейс ввода/вывода рассмотрен в учебных пособиях [1, 3].

В теме изучаются интерфейс параллельного КР580ВВ55 и после-довательного КР580ВВ51 ввода/вывода, структурная схема програм-мируемого интерфейса.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

После проработки теоретического материала раздела 8 необходимо ответить на вопросы теста № 8.

Максимальное количество баллов, которые Вы можете получить после изучения раздела 8, составляет 5 баллов за тестирование.

**Раздел 9. Программное обеспечение МПК КР580**

9.1. Программная модель МП системы

Программная модель МП типа КР580ВМ80А рассмотрена в учебных пособиях [1, 3].

В теме изучаются внутренние регистры: регистры данных, регистры управления, внешние регистры, память, средства ввода/вывода.

Принцип программного объединения 8-битных РОН в 16-битные реги-стровые пары. *H-*пара как основной указатель памяти.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

9.2. Способы адресации в МП системе

Режимы адресации и система команд МПК КР580ВМ80 рассмот-рены в учебных пособиях [1, 3].

В теме изучаются прямая адресация, непосредственная адресация, регистровая адресация, косвенная адресация, возможность программного осуществления индексной адресации.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

Выполнить практическое занятие № 20 по разделу «Программное обеспечение МПК КР580**»**, если это предусмотрено Вашим тематическим планом.

После проработки теоретического материала раздела 9 необходимо ответить на вопросы теста № 9.

По разделу следует проделать лабораторную работу № 8, содержание которой приведено в «Методических указаниях к выполнению лабораторных работ» УМК [4].

Максимальное количество баллов, которые Вы можете получить после изучения раздела 9, составляет 5 баллов за тестирование, 3 балла за правильное выполнение лабораторной работы и 2 балла за правильное выполнение практического занятия.

**Раздел 10. Система команд МПК КР580**

10.1. Команды пересылки данных и арифметико-логические команды

Системы команд рассмотрены в учебных пособиях [1, 3].

В теме изучаются разделение системы команд МП на группы в соответствии с их функциональным назначением, группы команд пересылки, арифметических и логических операций.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

Обратиться к практикуму, выполнить практические занятия №21 и № 22 теме «Команды пересылки данных и арифметико-логические команды».

10.2. Команды передачи управления, ввода-вывода и др.

Команды передачи управления, команды ввода-вывода, специальные команды рассмотрена в учебных пособиях [1, 3].

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

Выполнить практическое занятие № 23 по теме «Команды передачи управления, ввода-вывода и др.».

После проработки теоретического материала раздела 10 необходимо ответить на вопросы теста № 10.

По разделу следует проделать лабораторную работу № 9, если это предусмотрено Вашим тематическим планом. Содержание работы № 9 приведено в «Методических указаниях к выполнению лабораторных работ» УМК[4].

Максимальное количество баллов, которые Вы можете получить после изучения раздела 10, составляет 5 баллов за тестирование, 3 балла за правильное выполнение лабораторных работ и 6 баллов за правильное выполнение практических занятий.

**Раздел 11. Программирование в МП системе**

11.1. Программирование на машинном языке

Программирование на машинном языке, формат бланка при про-граммировании на машинном языке рассмотрены в учебных пособиях [1, 3].

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

11.2. Программирование на ассемблере

Директивы ассемблера рассмотрены в учебных пособиях [1, 3].

В теме изучаются назначение директив и псевдокоманд, макрокоманды, применение макрокоманд в прикладных программах.

Структуры данных и организация подпрограмм в МПК КР580ВМ80.

Организация массивов, очередей, стеков.Понятие одномерного мас-сива. Многомерные массивы, способ обработки. Структура данных в виде очереди.

Стек как специальная разновидность одномерного массива.

Подпрограммы как средство модульного программирования**.**Общая организация подпрограмм. Подпрограмма как законченный сегмент (модуль) программы.

Особенности программирования на языках высокого уровня.

Понятие о компромиссах между аппаратными и программными средствами.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы, выполнить практическое занятие № 24 по теме «Программирование в МП системе**».**

После проработки теоретического материала раздела 11 необходимо ответить на вопросы теста № 11.

По разделу следует выполнить курсовой проект, задание на который приведено в «Методических указаниях к выполнению курсового проекта» УМК [4].

Максимальное количество баллов, которое Вы можете получить после изучения раздела 11, составляет 5 баллов за тестирование, 10 баллов за правильное выполнение курсового проекта и 2 балла за правильное выполнение практического занятия.

**Раздел 12. Микроконтроллер КМ1816ВЕ48**

12.1. Аппаратное обеспечение КМ1816ВЕ48

Особенности архитектуры микроконтроллера КМ1816ВЕ48, струк-турная схема, основные узлы рассмотрены в учебных пособиях [1, 3].

В теме изучаются назначение в составе аппаратных средств микроконтроллера стираемого перепрограммируемого ПЗУ (СППЗУ) про-грамм емкостью 1 Кбайт, возможность пределов его расширения, назначение регистрового ОЗУ данных, обеспечение прямой адресации внешнего ОЗУ.

Реализация в МК 1816 системы векторного прерывания от двух источников: внутреннего таймера-счетчика событий и внешнего источника.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

12.2. Программное обеспечение КМ1816ВЕ48

Особенности программирования КМ1816ВЕ48 рассмотрены в учебных пособиях [1, 3].

В теме изучаются формат команд, способы адресации в командах микроконтроллера КМ1816, особенности языка ассемблер для КМ1816, команды, выполняемые за один машинный цикл и за два машинных цикла. Классификация групп команд по функциональному признаку. Применение специальных символов #, @ при написании программ. Особенности команд операций с таймером.

После проработки теоретического материала темы следует ответить на контрольные вопросы, приведенные в конце темы.

Выполнить практическое занятие № 25 по разделу «Микроконтроллер КМ1816ВЕ48**»**, если это предусмотрено Вашим тематическим планом.

После проработки теоретического материала раздела 12 необходимо ответить на вопросы теста № 12.

По разделу следует проделать лабораторную работу № 10, содержание которой приведено в «Методических указаниях к выполнению лабораторных работ» УМК [4].

Максимальное количество баллов, которые Вы можете получить после изучения раздела 12, составляет 5 баллов за тестирование , 3 балла за правильное выполнение лабораторной работы и 2 балла за правильное выполнение практического занятия.

Заключение(2 часа)

Расширение сферы применения микропроцессоров и микроконтроллеров, улучшение технических параметров цифровых устройств и микропроцессоров при совершенствовании схем микроэлектроники и наноэлектроники.

**3.3. Глоссарий**

Адресное пространство – совокупность ячеек памяти, к которым потенциально может адресоваться МП.

Адрес страницы – старшие восемь линий шины адреса.

Адрес в странице (или строка) – младшие восемь линий шины адреса.

Аккумулятор – регистр, используемый для выполнения арифметико – логических операций.

Алгебра логики – часть математической логики, называемая исчис-лением высказываний.

Аналитический способ задания функции – способ задания логической функции по правилам и теоремам булевой алгебры.

Аналого-цифровой преобразователь – устройство для преобразования сигналов из аналоговой формы в цифровую.

Арифметико-логическое устройство – сложная двухвходовая комби-национная схема, рассчитанная на параллельное преобразование двух 8-битных операндов.

Асинхронный триггер – простейший конечный автомат, состояние которого определяется входными сигналами установки и сброса.

Буферные схемы канала адреса и данных – предназначены для разделения внешних и внутренних ША и ШД и обеспечивают допустимую нагрузку по току.

Ввода / вывода (В/В) средства – порты ввода и порты вывода, которые представляют собой буферные регистры, имеющие определенные адреса и выполняющие задачу сопряжения МП системы с периферийными ус-тройствами (интерфейс В/В).

Восьмеричная система счисления – система с основанием восемь.

Время выборки – интервал времени между моментом подачи сигнала выборки и появлением данных на выходе устройства.

Входной сигнал сброса (инициализации) ***RESET –*** заставляет МП начать выполнение программы с нулевой ячейки памяти.

Входной сигнал готовности ***READY (RDY) –*** сигнал,поступающий от других элементов системы и приостанавливающий действие МП до тех пор, пока этот элемент не будет готов к обмену данными с МП.

Выходной сигнал ожидания ***WAIT –*** свидетельствующийо том, что МП приостановлен.

Выходной сигнал приема данных ***DBIN*** (Data Bus In) – определяет направление передачи по ШД в микропроцессор*.*

Выходной сигнал записи (выдачи) – определяет направление передачи по ШД по МП.



Выходной сигнал синхронизации ***SYNC –*** показывает начало каждого машинного цикла, в течение которого МП адресуется к внешнему регистру, обменивается с ним данными и производит внутренние преобразования данных.

Входной сигнал запроса шины (захвата) ***HOLD –*** сообщает МП, что быстродействующее периферийное устройство должно использовать ША и ШД для прямого обмена данными с памятью без участия МП.

Горнера схема – метод преобразования числа из одной системы счисления в другую.

Данных шина – шина, по которой производится обмен любой информацией между всеми компонентами микропроцессорной системы: команды, операнды, результаты операций, вводимые и выводимые данные.

Двоично-взвешенная резистивная цепь – схема с суммированием напряжений или токов.

Двоичный счетчик с последовательным переносом – счетчик суммирующего типа.

Декадный счетчик – счетчик с коэффициентом пересчета 10.

Демультиплексор – устройство, коммутирующее один входной сигнал на несколько выходов.

Дешифратор – устройство для распознавания кодовых комбинаций.

Директивы ассемблера – это команды, относящиеся к классу псевдокоманд, т. е. таких команд, которые не могут породить объектную программу.

Дискретизация – разбиение интервала времени на заданные моменты времени.

Емкость запоминающего устройства – количество информации, которое может храниться в нём.

Задание логической функции числовым способом – задание, при котором номера наборов соответствуют единичным значениям функции.

Закон ассоциативности – устанавливает порядок раскрытия и постановки скобок среди переменных.

Закон коммутативности – устанавливает порядок записи переменных, который не влияет на результат.

Запрещенные сочетания входных сигналов – неустойчивые сочетания входных сигналов триггера.

Идентификаторы внутренних регистров –идентификаторы РОН с соответствующими двоичными значениями от 000 до 111, встроенные в языке ассемблер МП.

Импликант – конъюнкция переменных с отрицанием или без него.

Информационный уровень – уровень напряжения, при котором принимаются логический «0» или логическая «1».

Исключающее ИЛИ – логическая функция, принимающая единичное значение только при несовпадении информационных значений входных сигналов.

Истинность – высказывание, в котором содержится смысл утверждения.

Истинности таблица – таблица, которая дает краткую запись значений логической функции при различных сочетаниях аргументов.

Карта Карно – таблица, имеющая ячейки для всех возможных минтермов функции.

Квантование – замена исходного аналогового напряжения ближайшими фиксированными уровнями.

Кодопреобразователь – устройство, преобразующее один код в другой.

Комбинационная схема – схема, полученная из структурной формулы и выполняющая заданную логическую функцию.

Конечный автомат – последовательностное устройство, содержащее комбинационное и запоминающее устройства.

Логическая функция – функция и аргументы которой могут принимать только два значения «1» и «0».

Логическая схема – схема, реализующая высказывание алгебры логики.

Логический базис – система логических элементов, достаточная для построения любых комбинационных устройств.

Логическое умножение – операция конъюнкции, смысл которой истинен, если одновременно истинны все входящие в неё переменные.

Макрокоманда – группа команд с особой мнемоникой, не входящей в систему команд МП, используемая для сокращения длины входной программы и ускорения программирования.

Максимальная группа – группа, объединяющая наибольшее число единиц по карте Карно.

Матрица микросхем ОЗУ – матрица, составленная из элементов памяти, расположенных вдоль строк и столбцов.

Метод минимизации Квайна-МакКласки – алгоритм минимизации, основанный на использовании операций неполного склеивания и поглощения.

Метка – символическое наименование адреса.

Мнемоника (код) – символическое наименование кода операции, отражающее семантику программы.

Микропроцессорный комплект – комплекс, содержащий три основных компонента: центральный процессорный элемент (ЦПЭ), память микро-процессора и средства ввода / вывода.

Микропроцессор – единственный активный компонент системы, реализующий функции управления выполнением команд программы, выбора команд, считывания операнд, преобразования их в соответствии со смыслом команд, определения адреса следующей команды, управления обменом информацией между компонентами системы, реагирования на внешние сиг-налы.

Минимизация недоопределенных функций – метод произвольного доопределения запрещенных наборов функций на карте Карно при выполнении минимизации.

Минтерм – произведение переменных, для которых значение функции истинно.

Модуль счета – число, характеризующее количество устойчивых состояний счетчика.

Мультиплексор – комбинационная схема, имеющая несколько адресных и информационных входов и один выход.

Набор аргументов – число аргументов, определяющих значение логической функции.

Наращивание разрядности – увеличение емкости памяти ОЗУ при увеличении разряда слова.

Номер набора – номер клетки в двоичном коде карты Карно при задании логической функции числовым способом.

Операнды – адреса памяти, внутренние регистры МП, адреса портов ввода и вывода, числовые и символьные константы.

Операция дизъюнкции – операция логического сложения, иначе операция ИЛИ.

Операция инверсии – операция логического отрицания, иначе операция НЕ.

Память микропроцессорной системы – устройство, состоящее из однотипных ячеек с разрядностью, равной длине слова МП, т. е. 1 байт (8 битов).

Параллельные регистры – ряд триггеров с общими сигналами управления и синхронизации и индивидуальными информационными входами для приема разрядов слова.

Перепрограммируемые постоянные запоминающие устройства – устройства памяти, хранящие записанную информацию неопределенно долго при отключенном питании, а также допускающие стирание записанной информации и запись новой.

Подпрограмма – это законченный модуль программы, который может вызываться при программировании многократно.

Позитивная логика – логической единице соответствует значение сигнала, превышающее верхний пороговый уровень, а логическому нулю соответствует значение сигнала, которое меньше, чем нижний пороговый уровень.

Поле комментария – начинается с какого-либо разделителя, например с точки с запятой. Текст в поле комментария полностью игнорируется программой-ассемблер.

Поля ассемблерной строки – поля метки, кода операции (мнемоники), операнда.

Полный сумматор – устройство, предназначенное для сложения трех одноразрядных двоичных чисел.

Полусумматор – устройство, предназначенное для сложения двух одно-разрядных двоичных чисел.

Последовательный регистр сдвига в одном направлении –регистр, имеющий информационный и управляющий входы и один выход.

Постоянное запоминающее устройство – устройство, в котором инфор-мация в ячейки памяти записывается однократно и в процессе эксплуатации используется только режим чтения.

Постоянное запоминающее устройство, программируемое маской – устройство, в котором информация в ячейки памяти записывается с помощью фотошаблона в процессе изготовления микросхем.

Постоянное запоминающее устройство, программируемое пользова-телем – устройство, в котором информация в ячейки памяти записывается с помощью специальных устройств – программаторов.

Правило де Моргана – правило, следующее из принципа двойственности булевой алгебры.

Преобразователь «время-код» – устройство, в котором подсчитывается число импульсов за определенный интервал времени.

Преобразователь «код-напряжение» – устройство, в котором сумми-руются элементарные напряжения, образуемые с помощью деления заданного эталонного напряжения.

Прецизионный делитель – устройство, состоящее из цепочки резис-торов.

Программатор – устройство, выдающее для записи информации в микросхему напряжение, которое прожигает плавкие перемычки в элементах памяти.

Программируемая логическая матрица – универсальный элемент, кото-рый может быть запрограммирован на выполнение логических функций различной степени сложности.

Программирующий импульс перепрограммируемого постоянного запо-минающего устройства – импульс, подаваемый в режиме записи информации с некоторой задержкой относительно момента подачи кодовой комбинации адреса и записываемого числа на программирующий вход.

Программируемый параллельный интерфейс – унифицированная интерфейсная БИС общего назначения, обладающая универсальностью применения, достигаемой за счет программирования.

Программируемый связной адаптер – специализированная интерфейс-ная БИС, используемая для построения интерфейса МП-систем с устройст-вами последовательной передачи данных.

Программируемый интервальный таймер – устройство для получения управляемых временных задержек и времязадающих функций: генерирование сигналов изменяемой частоты; подсчет числа внешних событий; формирова-ние сигналов, задержанных во времени на программируемую величину, и т. д.

Псевдокоманды *–* это указания программе ассемблер о выполнении определенных действий в процессе ассемблирования, размещающие в памяти информацию и присваивающие численные значения символическим наимено-ваниям.

Распределитель – демультиплексор, т. е. коммутатор логических сигна-лов.

Регистр – устройство, способное воспринимать, хранить и выдавать слово информации.

Регистр-указатель стека – адресует последнюю занятую ячейку «верхушку» стека. Перед загрузкой в стек каждого байта производится декремент (уменьшение на 1) указателя стека.

Режимы адресации в МП – прямая адресация, непосредственная, регистровая, косвенная.

Режим записи новых слов – один из режимов оперативного запоми-нающего устройства.

Режим стирания информации в перепрограммируемом постоянном запоминающем устройстве – режим, при котором на микросхему подается воздействие, уничтожающее предыдущую информацию.

Резистивная цепь R-2R – многозвенная цепь для деления тока в ЦАП.

Сигнал переноса – сигнал, образующийся в одноразрядных сумматорах двоичных чисел.

Сигналы управления состоянием МП: входной сигнал сброса (ини-циализации) ***RESET***; входной сигнал готовности ***READY (RDY)***; выходной сигнал ожидания ***WAIT.***

Сигналы управления ША и ШД:выходной сигнал приема данных ***DBIN***; выходной сигнал записи (выдачи) ; входной сигнал запроса ши-ны (захвата) ***HOLD.***



Система логических элементов И, ИЛИ, НЕ – система, достаточная для построения любых комбинационных устройств.

Система команд МПК – команды пересылки данных и арифметико-логические команды, команды передачи управления, ввода-вывода и другие команды.

Системный контроллер – предназначен для фиксации слова состояния МП, которое часто обозначается ***PSW*** (PROCESSOR STATUS WORD); выработки системных управляющих сигналов, буферизации ШД и управления направлением передачи данных.

Склеивающиеся слагаемые – минтермы, которые записаны в виде «1» в соседних клетках карты Карно по вертикали и горизонтали.

Совершенная дизъюнктивная нормальная функция (СДНФ) – выраже-ние, содержащее конъюнкции всех переменных или их инверсий.

Совершенная конъюнктивная нормальная функция (СКНФ) – выраже-ние, содержащее дезъюнкции всех переменных или их инверсий.

Столбец неопределенности – столбец по карте Карно, определяющий неоднозначность выходного сигнала триггера при определенной совокупности входных сигналов.

Счетчики с произвольным модулем счёта – счетчики, модуль счёта которых не равен целой степени числа 2.

Счетчик программный – используется для хранения текущего адреса программной памяти. При каждом обращении к программной памяти производится увеличение на 1 (инкремент) его содержимого.

Схема десятичной коррекции – устройство для преобразования двоич-ного кода в двоично-десятичный (***BCD*** -код) при наличии соответствующей команды.

Таблица истинности – таблица, которая даёт краткую запись логических действий.

Типовые приемы алгебраической минимизации – применение всех возможных преобразований с целью получения тупиковой формы записи.

Тупиковая форма записи – логическая функция, не имеющая избыточных членов и не поддающаяся дальнейшей минимизации.

Универсальный логический элемент – элемент системы, обладающий безызбыточной функциональной полнотой.

Универсальный элемент «стрелка Пирса» - логический элемент ИЛИ-НЕ.

Универсальный элемент «штрих Шеффера» - логический элемент И-НЕ.

Устройство неравнозначности – устройство, реализующее логическую функцию, выходной сигнал которой равен «1» только при несовпадении информационных значений входных сигналов.

Устройство равнозначности – устройство, реализующее логическую функцию, выходной сигнал которой равен «1» только при совпадении информационных значений входных сигналов.

Факультативные условия – относятся к не полностью определенным логическим функциям.

Флаги – двоичные признаки, формируемые при выполнении многих команд микропроцессора: если какая-либо особенность имеет место, то в регистр флагов записывается 1, в противном случае – 0.

Форма представления числа – представление числа в ЭВМ с фиксированной и плавающей точками.

Функциональная полнота – система логических элементов, достаточная для построения любых комбинационных устройств.

Центральный процессорный элемент – микропроцессор.

Цифроаналоговый преобразователь – устройство, преобразующее код в аналоговое напряжение или ток.

Цифровой автомат – последовательностное устройство, состоящее из комбинационного устройства и памяти.

Шаг квантования – интервал между соседними уровнями квантования.

Шина системная – состоит из шины адреса ША, шины данных ШД и шины управления ШУ, по которым осуществляется взаимодействие микро-процессора с памятью и средствами ввода/вывода.

Шум квантования – ошибки, возникающие при замене значений аналоговой величины ближайшими уровнями квантования.

Язык ассемблер – это язык символического кодирования, позволяющий представить все элементы программы в символической буквенно-цифровой форме, которая отражает их содержательный смысл.

**3.4. Технические и программные средства обеспечения дисциплины**

**Описание компьютерной программы**

При выполнении лабораторных работ, а также практических занятий по дисциплине используется программа моделирования электронных устройств и систем Micro-Cap 9 (MC9) DEMO.

Запуск программы Micro-Cap 9 (MC9) осуществляется щелчком мыши по специальной пиктограмме. В системе MC9 используется многооконный интерфейс с ниспадающими и разворачивающимися меню. Главное меню состоит из следующих режимов: Fail, Edit, Component, Windows, Options, Analysis, Help.

Для выбора одного из режимов необходимо подвести указатель «мыши» к требуемому режиму и щелкнуть левой клавишей мыши. После этого на экране появится «спускающееся» меню команд. Для выполнения необходимой команды следует щелкнуть по ней левой клавишей мыши.

Чтобы провести анализ любого радиоэлектронного устройства по заданной принципиальной схеме, прежде всего следует отредактировать параметры компонентов, входящих в данную схему. Параметры компонентов, используемых в схеме, обычно задаются в процессе создания чертежа схемы, однако для их редактирования необходимо использовать диалоговое окно атрибутов компонентов.

Если создан чертеж схемы, т. е. файл описания схемы записан на диске, то для его открытия можно воспользоваться соответствующей стандартной пиктограммой либо использовать команду Open из режима File. При этом на экране появляется окно со списком файлов описания схем, из которого нужно выбрать необходимый чертеж схемы.

Для вызова диалогового окна атрибутов компонентов необходимо щелкнуть два раза левой клавишей мыши по соответствующему компоненту на схеме либо по его атрибуту. К атрибутам компонентов относятся позиционное обозначение PART, например R1, C2, и номинальное значение параметра VALUE, например 2.2к, 10pF.

**Работа в режиме анализа переходных характеристик Transient**

Для анализа переходных процессов следует выбрать пункт Transient Analysis в меню Analysis. При отсутствии ошибок в схеме открывается окно задания параметров моделирования Transient Analysis Limits

**Числовые параметры**

|  |  |
| --- | --- |
| Time Range (*T*max [, *T*min]) | Конечное и начальное время расчета переходных процессов |
| Maximum Time Step | Максимальный шаг интегрирования |
| Number of Points | Количество точек, выводимых в таблицы. |
| Temperature (*t*max [, *t*min [, *t*step ]]) | Диапазон изменения температуры |

В разделе «Вывод результатов моделирования» в окне Transient Analysis Limits имеется дополнительная графа User File – создание на диске файла, в который заносятся отсчеты функции, заданной в графе Y Expression.

**Опции**

**Run Options** – управление выдачей результатов расчетов. Имеет те же значения, что и для DC анализа.

**State Variables** – установка начальных условий:

|  |  |
| --- | --- |
| Zero | Установка нулевых начальных условий |
| Read | Чтение начальных условий с дискового файла перед каждым вариантом расчета |
| Leave | Установка в качестве начальных условий значений, полу-ченных при окончании расчета предыдущего варианта |

**Operation Point** – включение режима расчета по постоянному току перед началом каждого варианта расчета переходных процессов.

**Operation Point Only** – расчет только режима по постоянному току (расчет переходных процессов не производится).

**Auto Scale Ranges** – автоматическое масштабирование по осям X и Y.

После перехода в режим расчета переходных характеристик в главном меню появляется новый режим Transient, который имеет те же команды, что и режим DC, и дополнительную команду DSP (Digital Signal Processing – цифро-вая обработка сигналов). Эта команда открывает окно DSP, в котором задаются границы интервала времени для расчета спектральных плотностей и количество отсчетов Number of Points, в точности равное степени числа 2 от 64 до 8192.

После установки всех параметров и опций в окне задания параметров моделирования следует нажать кнопку Run или функциональную клавишу F2 для начала расчета переходных характеристик Transient.

**3.5. Методические указания к проведению практических занятий**

**Общие указания**

Студенты специальности 210302.65 выполняют практические занятия в соответствии с заданным вариантом задания. К выполнению задания студенты допускаются только после проверки преподавателем их подготовленности. Для проведения практических занятий на ПК комплектуется группа из 10-12 человек (по 2 студента на рабочее место).

Рабочее место представляет собой стол, на котором расположен персональный компьютер (ПК). При выполнении заданий должны соблюдаться правила техники безопасности при работе с персональным компьютером.

После окончания практических занятий студенты должны выключить все блоки ПК.

Отчет о выполненных работах оформляется в электронном виде индивидуально каждым студентом в соответствии с шифром. Экспериментальные и расчетные данные следует оформлять в виде таблиц, графиков и распечаток с принтера в соответствии с указаниями, приведенными в описаниях занятий; графики и распечатки должны быть аккуратно вклеены в соответствующие места отчета. Кривые на графиках могут быть вычерчены тушью, карандашом или фломастером. На графиках и распечатках внизу должны быть приведены принятые обозначения и ссылки на таблицы, согласно которым построены кривые. Каждый пункт отчета, помимо таблиц и графиков, должен содержать краткое объяснение полученных результатов с выводом о проделанной работе.

На обложке отчета следует указать название университета и кафедры; год; фамилию, имя, отчество студента; специальность; шифр; УКП и группу, а также название дисциплины, по которой выполнены практические задания.

Студент должен проявить умение работать с программами анализа различных режимов радиоэлектронных устройств, с библиотеками элементов, используемых при анализе; уметь объяснить ход кривых, полученных при расчетах, математическом и имитационном моделировании.

**Практическое занятие 1**

**по теме «Алгебра логики»**

**Теоретическая часть занятия**

При подготовке к данному занятию следует ознакомиться с темой 1.1 в учебном пособии [1].

Для математического описания и анализа работы цифровых устройств применяется алгебра логики, которая представляет собой исчисление высказываний. Любое высказывание является логической функцией от некоторого числа аргументов. При большом числе аргументов логические функции можно представить как совокупность более простых функций или логических операций от одного или двух аргументов.

К простым логическим операциям относятся операции дизъюнкции, конъюнкции, инверсии.

***Операция дизъюнкции.*** Это операция логического сложения, иначе операция **ИЛИ**. Аналитически операция дизъюнкции записывается следующим образом:

*Y=x1+x2=x1х2*. **(**1**)**

Выражение (1) дает краткую запись следующих действий:

10=1; 01=1; 11=1; 00=0.

***Операция конъюнкции.*** Эту операцию называют также операцией **И** (операцией логического умножения). Аналитически в случае двух переменных эта операция записывается в виде

*Y=x1x2=x1* *x2****.***  (2)

Смысл операции конъюнкции заключается в том, что ***Y=1***, если одновременно ***x1=*** 1 и ***x2=*** 1.

В случае ***n*** переменных: *Y=x1x2х3....хn.* (3)

***Операция инверсии.*** Это операция логического отрицания, иначе опе

\_

рация **НЕ.** Аналитически это записывается в виде *Y= x.*

Основные правила выполнения операций дизъюнкции и конъюнкции для одной переменной с константами и с инверсией самой переменной можно сформулировать следующим образом:

\_

*x* v *0=x; x* v *1=1; x* v *x=x; x* v *x=1; x* v *x* v *x… =x;*

*\_ =*

*x ∙* 0*=*0*; x ∙* 1*=x; x ∙ x=x; x ∙ x=*0*; x = x.* (4)

Схемная реализация элемента **ИЛИ** в программе MC9 DEMO, представлена на рис. 1.

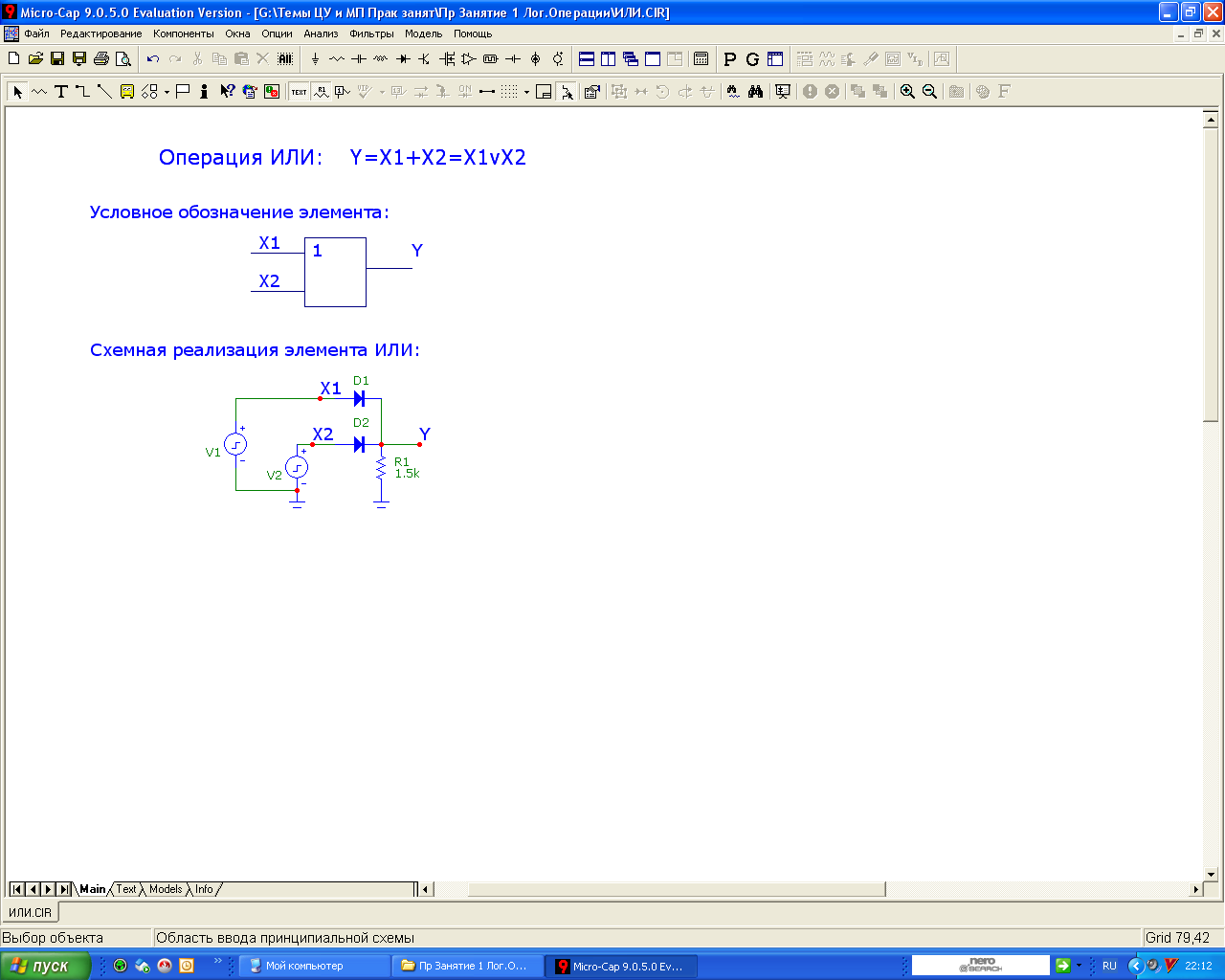


Рис. 1

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соответствующую пиктограмму;

2. В появившемся окне выбрать файл «ИЛИ» и открыть его (рис. 1);

3. Выбирать режим Transient Analysis (Alt+1) – расчет переходных процессов;

4. Зафиксировать осциллограммы процессов во времени и установить временные интервалы, в течение которых выполняется операция ИЛИ;

5. Составить таблицу истинности.

Схемная реализация элемента **И** в программе MC9 DEMO представлена на рис. 2.

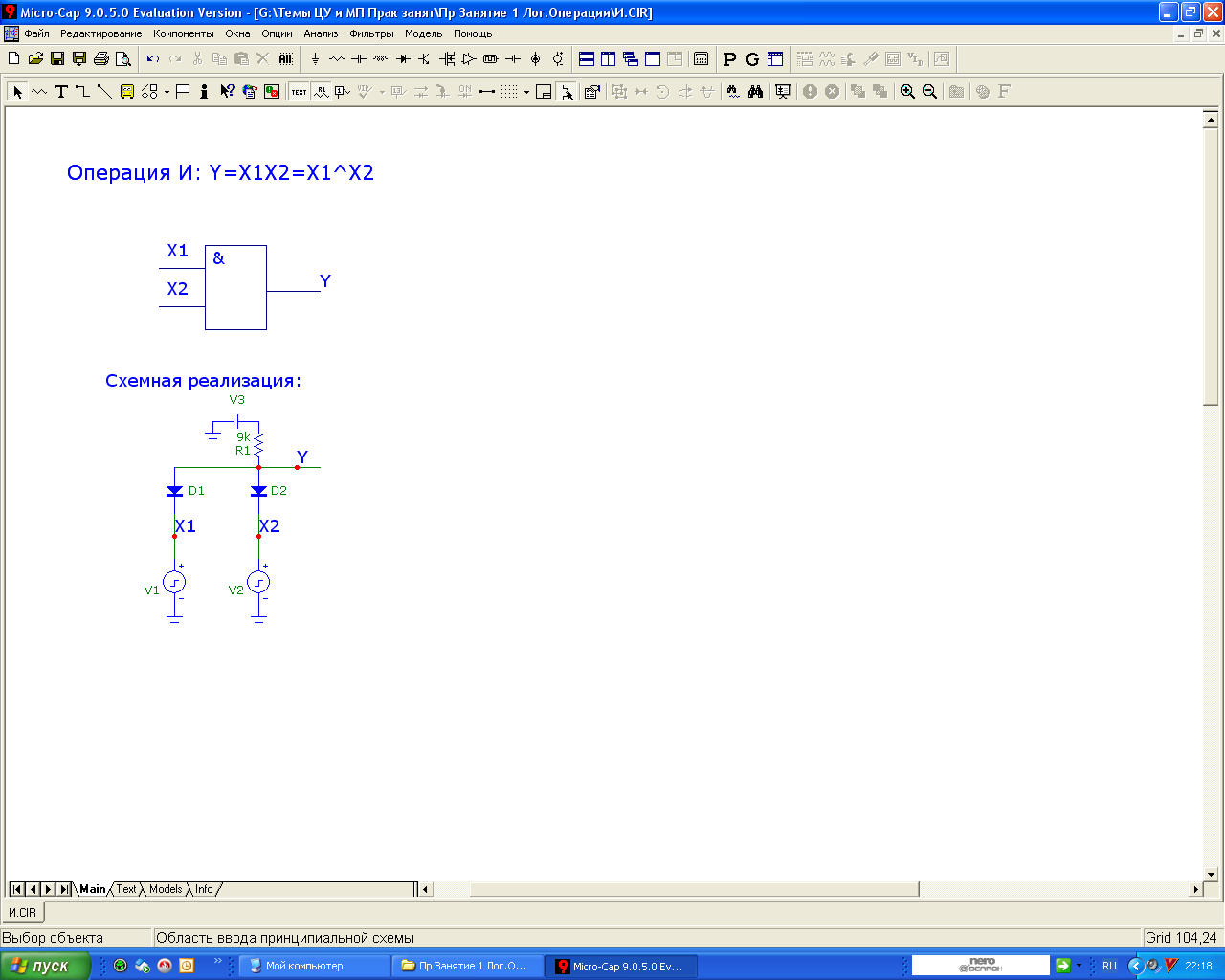


Рис. 2

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соот-ветствующую пиктограмму;

2. В появившемся окне выбрать файл «И» и открыть его (рис. 2);

3. Выбирать режим Transient Analysis (Alt+1) – расчет переходных процессов;

4. Зафиксировать осциллограммы процессов во времени, и установить временные интервалы, в течение которых выполняется операция И;

5. Составить таблицу истинности.

**Задание**

Используя схемы рис. 1, 2 и компоненты электрических схем табл. 1, самостоятельно разработать схемы для доказательства выражений (4) и про-вести моделирование.

**Содержание отчета**

1. Титульный лист с названием работы.
2. Основные теоретические положения.
3. Схемы логических выражений и таблицы истинности.
4. Схемы для доказательства выражений (4) и результаты моделирования.

### *Таблица 1.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ п/п** | **Название** | **Условное графическое обозначение** | | **Доступ к компоненту в МС** |
| **Россия  ГОСТ 2.743-72** | **США MIL-STD 806B Используется в МС** |
| **1** | **“И”** |  |  | **Component/Digital Primitives/Standard Gates/And Gates/And2** |
| **2** | **“ИЛИ”** |  |  | **Component/Digital Primitives/Standard Gates/Or Gates/Or2** |
| **3** | **“НЕ”** |  |  | **Component/Digital Primitives/Standard Gates/Inverters/Inverter** |
| **4** | **Цифровой переключатель** |  |  | **Component/Animation/Digital Switch** |
| **5** | **Элемент индикации** |  |  | **Component/Animation/LED** |

**Занятие 2**

**по теме «Алгебра логики»**

**Теоретическая часть занятия**

В занятии производится доказательство принципа двойственности в алгебре логики. Из сопоставления таблиц истинности для операций дизъюнкции, конъюнкции и отрицания можно отметить следующую закономерность. Операции Ии ИЛИможно поменять местами, если «1» поменять на «0», а логическое умножение – на логическое сложение:

если то  (1)

еслито  (2)

Равенства (1) и (2) отражают принцип двойственности в алгебре логики.

**Задание**

1. Изучить раздел 1 учебного пособия [1].

2. Для доказательства тождеств (1) и (2) , используя компоненты табл. 1, приведенные в занятии 1, построить в среде MC9 DEMO схемы, являющиеся аналогами логических выражений обеих частей тождества и проанализировать выходные значения у каждой схемы.

**Порядок выполнения**

Провести моделирование схем и зафиксировать результаты в таблице истинности. Если схемы составлена правильно и доказываемые тождества справедливы, то столбцы выходных значений, соответствующие левой и правой частям тождеств (1) и (2), будут одинаковыми.

**Содержание отчета**

1. Титульный лист с названием работы.

2. Основные теоретические положения.

3. Схемы логических выражений и таблицы истинности.

**Занятие 3**

**по теме «Построение логических схем»**

Задано логическое выражение функции двух переменных в виде



1. **Задание**
2. Создать схему цифрового устройства для исследования заданной логической функции двух переменных и провести ее анализ.
3. На рис. 1 представлена схема, которую следует реализовать и моделировать в среде MC9 DEMO.

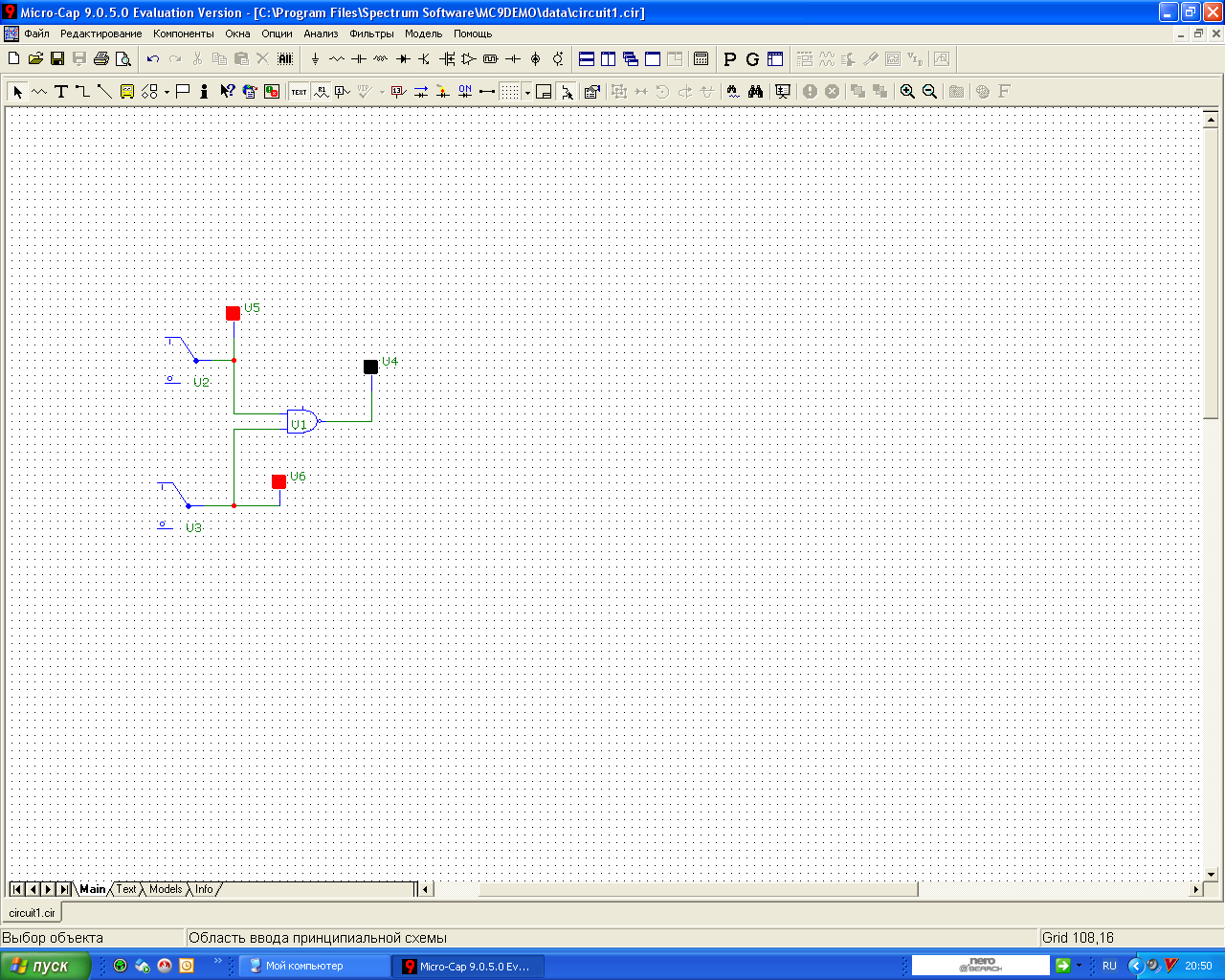


Рис. 1

**Порядок выполнения занятия**

1. Запустить МС 9, после чего активизируется диалоговое окно **New File**, в котором нужно выбрать **Schematic** и нажать **ОК**. Для удобства расположения компонентов рекомендуется включить режим отображения сетки кнопкой с пиктограммой **Grid .**



1. С помощью мыши выбрать пункт меню **Component/Digital Primitives/Standard Gates/Nand Gates/Nand2** (логический элемент, выполняющий логическую функцию “И-НЕ” с двумя переменными). В окне редактирования изображения схемы появится условное графическое изображения (УГИ) этого компонента, которое будет двигаться при перемещении мыши.
2. В открывшемся диалоговом окне **Attribute dialog box** задания параметров компонентов для параметра **TIMING MODEL** в правой части окна выбрать D0\_GATE.
3. Аналогично, выполнив команду **Component**/**Animation/Digital Switch,** разместить новый компонент– цифровой переключатель. Для установки вто-рого переключателя достаточно щелкнуть мышью второй раз в нужном месте.
4. Последний компонент – светодиодный индикатор выбирается по команде **Component**/**Animation/LED**.
5. Соединить компоненты проводниками, для чего выбрать на панели инструментов проводник (**Wire),** нажав кнопку или комбинацию клавиш CTRL+W. Для создания проводников с двумя изломами данный инструмент применяют дважды.



1. Провести анализ схемы. Для этого в меню **Analysis** выбрать команду **Transient Analysis** (комбинация клавиш Alt+1). Если ошибки отсутствуют (в противном случае в режиме редактирования устранять ошибки в изображении схемы), то появляется диалоговое окно задания пределов изменения переменных (**Analysis Limits dialog box**). Согласиться с предлагаемыми значениями, так как элементом индикации в настоящее время является светодиод, и нажать **RUN**.
2. Для того чтобы видеть изображение схемы, которое закрыто окном с результатами анализа, изменить взаимное расположение окон, нажав одну из кнопок или выбрав соответствующую команду в меню **Windows**. Светодиодный индикатор, подключенный к выходу индицирует его состояние: красный цвет – логическая “1”, черный – “0”.



1. Изменить положение цифрового переключателя. Выполнить команду **Run** (нажатием на кнопку или клавишу F2). Зафиксировать входные и выходные логические состояния.



1. Составить таблицу истинности анализируемой схемы, для которой существуют 4 возможных двоичных наборов аргументов.

**Содержание отчета**

1. Титульный лист с названием работы.

2. Схема логической функции и таблица истинности.

3. Выводы.

**Занятие 4**

**по теме «Построение логических схем»**

**Теоретическая часть занятия**

Можно показать, что, уменьшая количествологических элементов, придем к системе из одного логического элемента, выполняющего все логические операции. Такие логические элементы называются универ-сальными и бывают двух типов.

***Универсальный логический элемент ИЛИ  НЕ.*** Этот логический элемент реализует логическую функцию отрицание дизъюнкции: 

Эта функция называется «стрелка Пирса» и обозначается: *Y=x1↓x2.*

Условное графическое обозначение элемента показано на рис. 1.

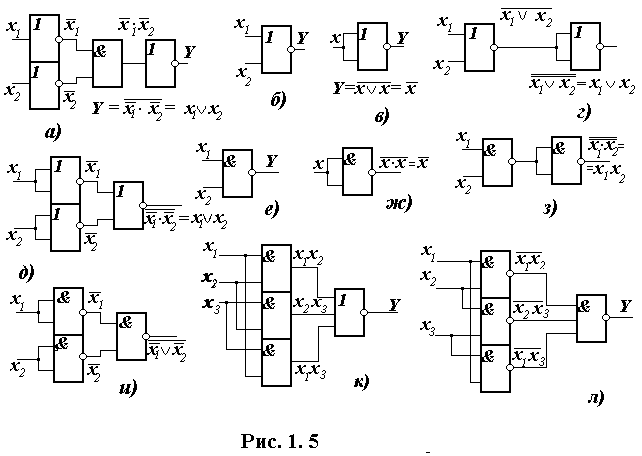
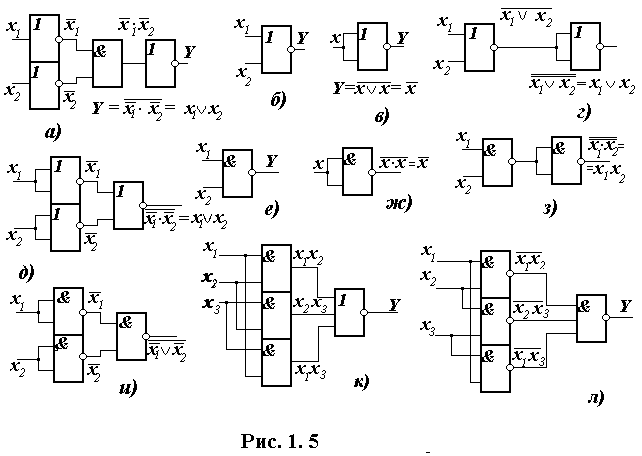
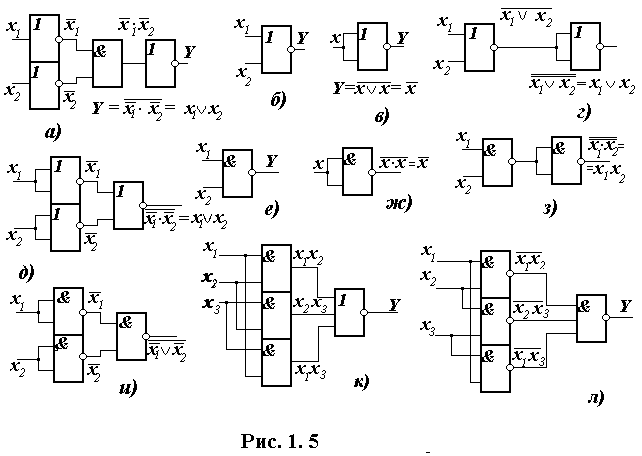
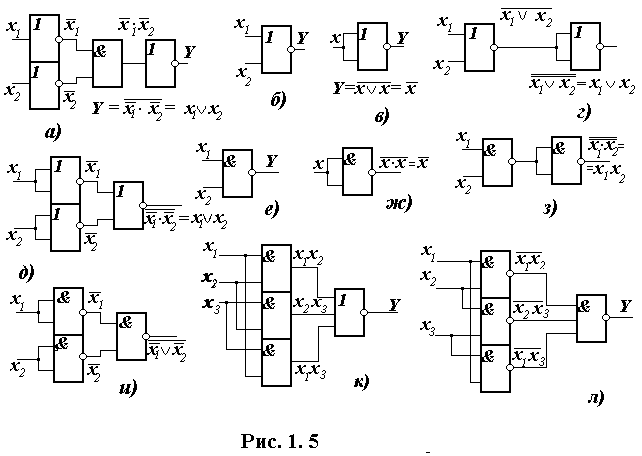


Рис. 1 Рис. 2

На рис. 2показано, как с помощью универсального элемента получить функцию НЕ, т. е. выполнить операцию инверсии.

Рис. 3 показывает реализацию функции ИЛИ, а рис. 4  функции И.



***Y= х1х2***

Рис. 3 Рис. 4

***Универсальный логический элемент И  НЕ.*** Этот логический элемент также образует функционально-полную систему логических элементов. Этот логический элемент реализует логическую функцию  («штрих Шеффера»). Его условное графическое изобра-жение показано на рис. 5*,* а на рис. 6 выполняемая им операция НЕ.

Две другие логические операции, выполняемые этим универсальным элементом, – И и ИЛИ показаны соответственно на рис. 7и на рис. 8. В общем случае все универсальные логические элементы могут иметь не два, а *n*входов.

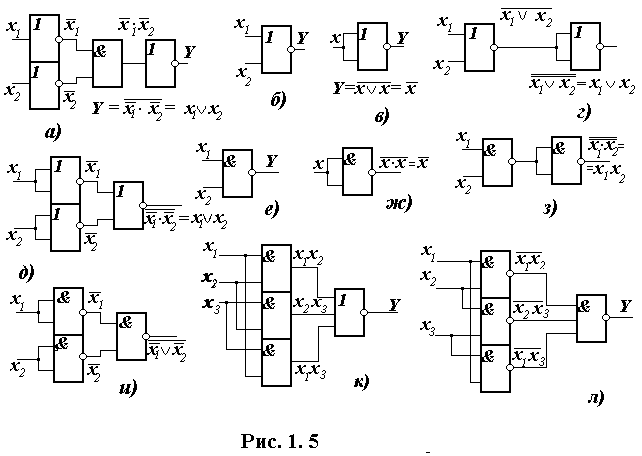
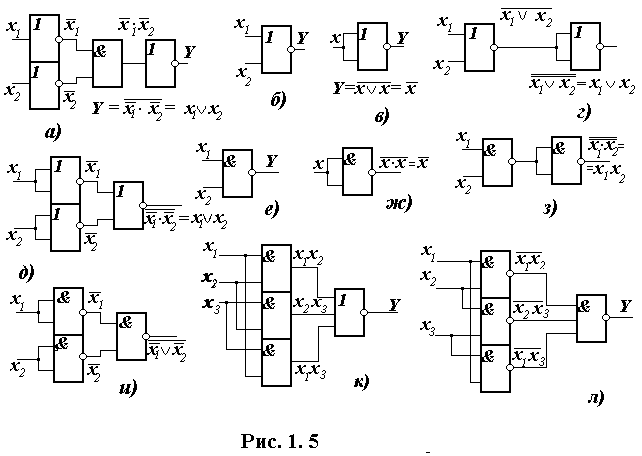


Рис. 5 Рис. 6

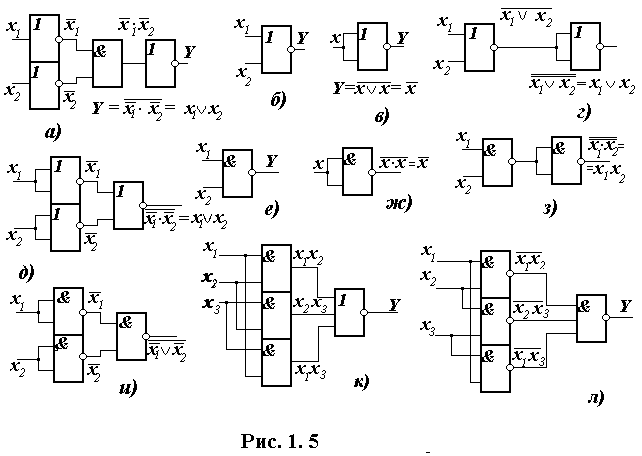
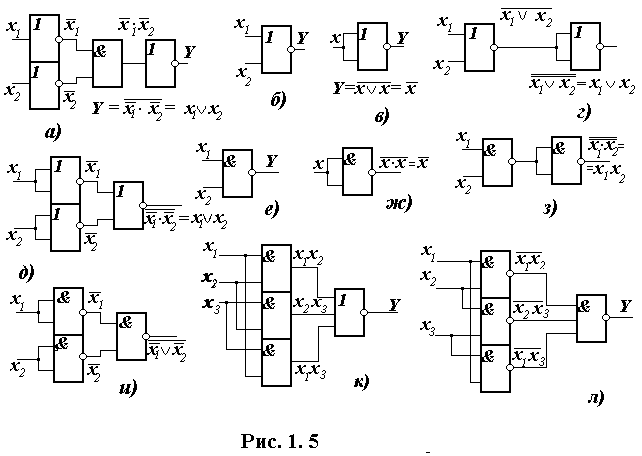


Рис. 7 Рис. 8

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соот-ветствующую пиктограмму;

2. В появившемся окне выбрать файл «ИЛИ-НЕ ИЛИ» и открыть его (рис. 9);

3. Выбрать режим Transient Analysis (Alt+1) – расчет переходных про-цессов;

4. Составить таблицу истинности.

Схемная реализация элемента ИЛИ, выполняемая универсальным элементом «ИЛИ-НЕ», синтезированная в программе MC9 DEMO, пред-ставлена на рис. 9.

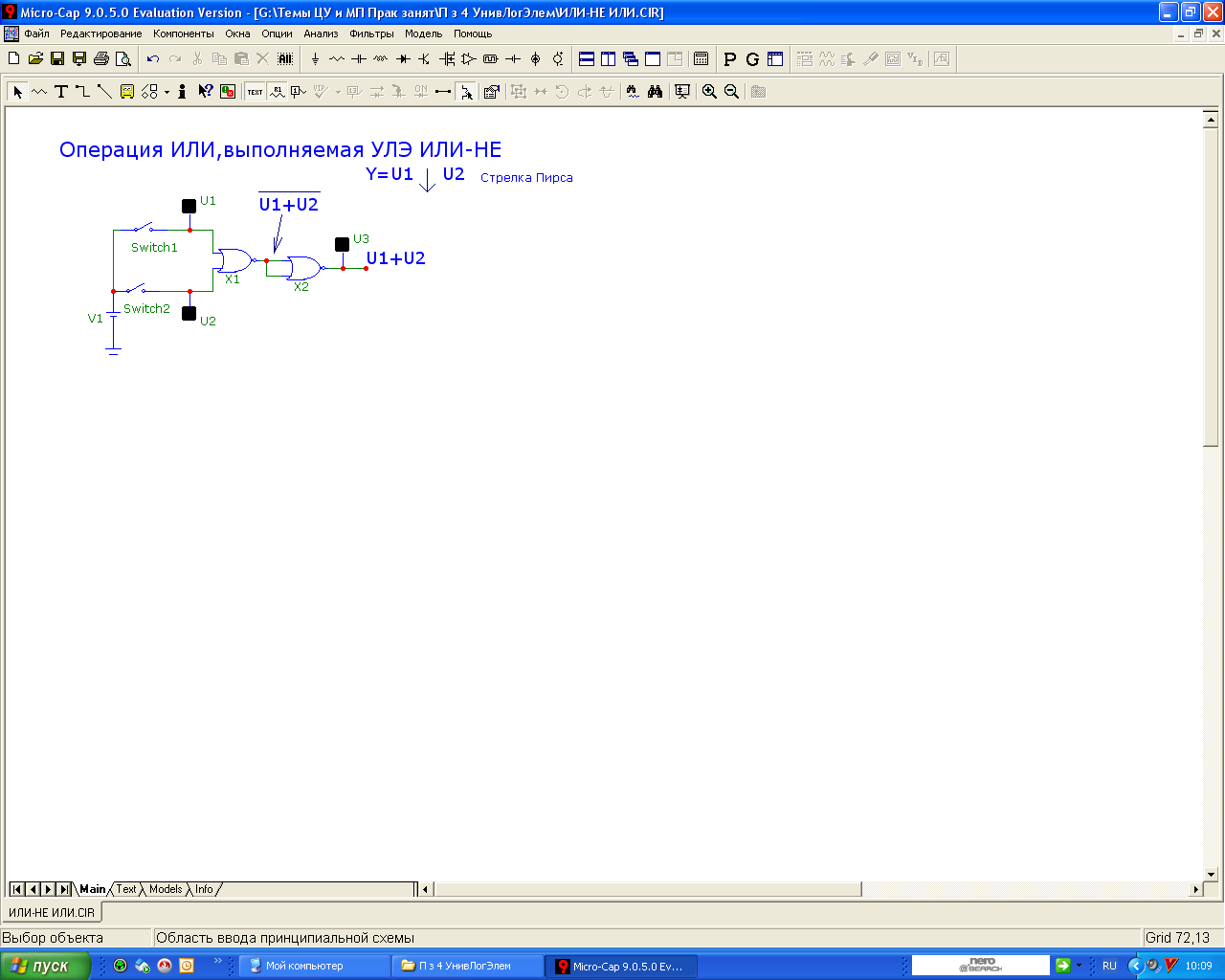


Рис. 9

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соот-ветствующую пиктограмму;

2. В появившемся окне выбрать файл «ИЛИ-НЕ И» и открыть его (рис. 10);

3. Выбрать режим Transient Analysis (Alt+1) – расчет переходных про-цессов;

4. Составить таблицу истинности.

Схемная реализация элемента И, выполняемая универсальным элементом «ИЛИ-НЕ», синтезированная в программе MC9 DEMO, представлена на рис. 10.

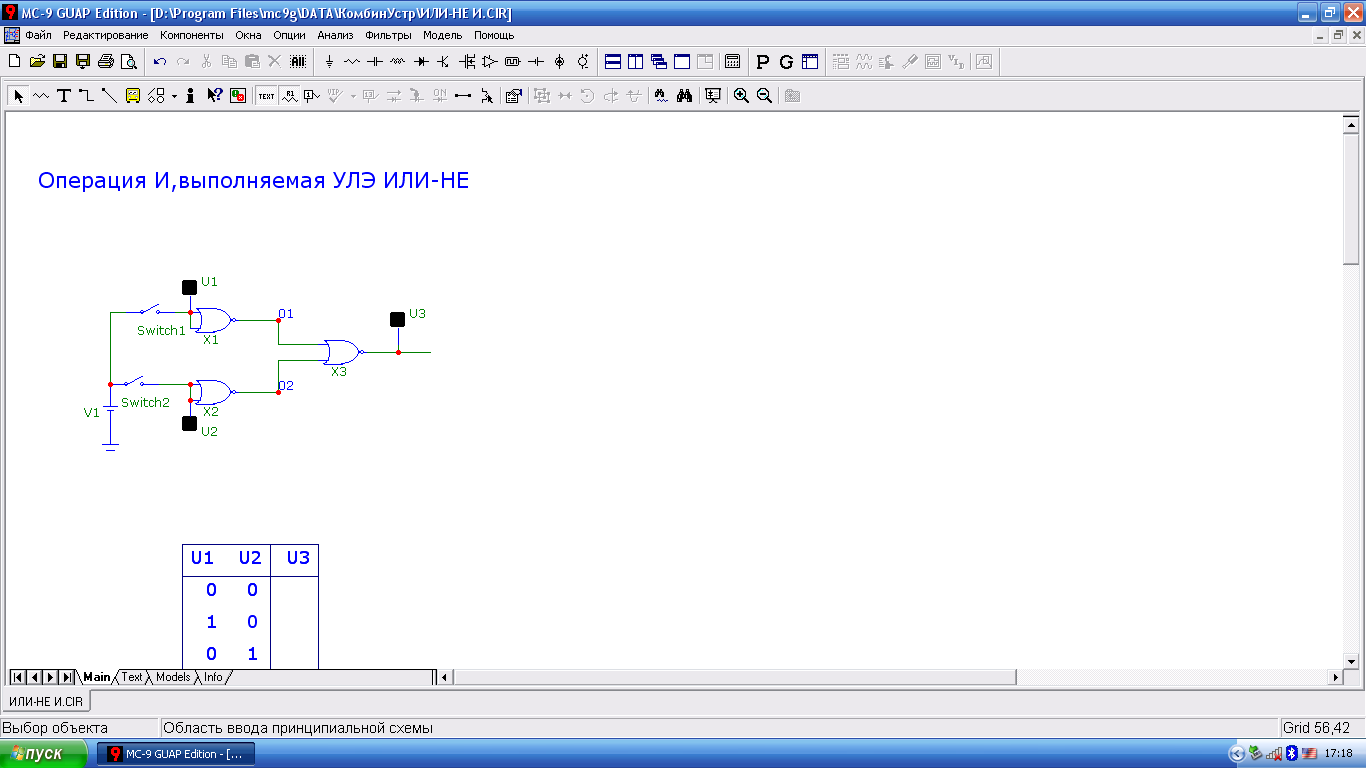


Рис. 10

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соот-ветствующую пиктограмму;

2. В появившемся окне выбрать файл «ИЛИ-НЕ НЕ» и открыть его (рис. 11);

3. Выбрать режим Transient Analysis (Alt+1) – расчет переходных процессов;

4. Составить таблицу истинности.

Схемная реализация элемента НЕ, выполняемая универсальным элемен-том «ИЛИ-НЕ», синтезированная в программе MC9 DEMO, представлена на рис. 11.

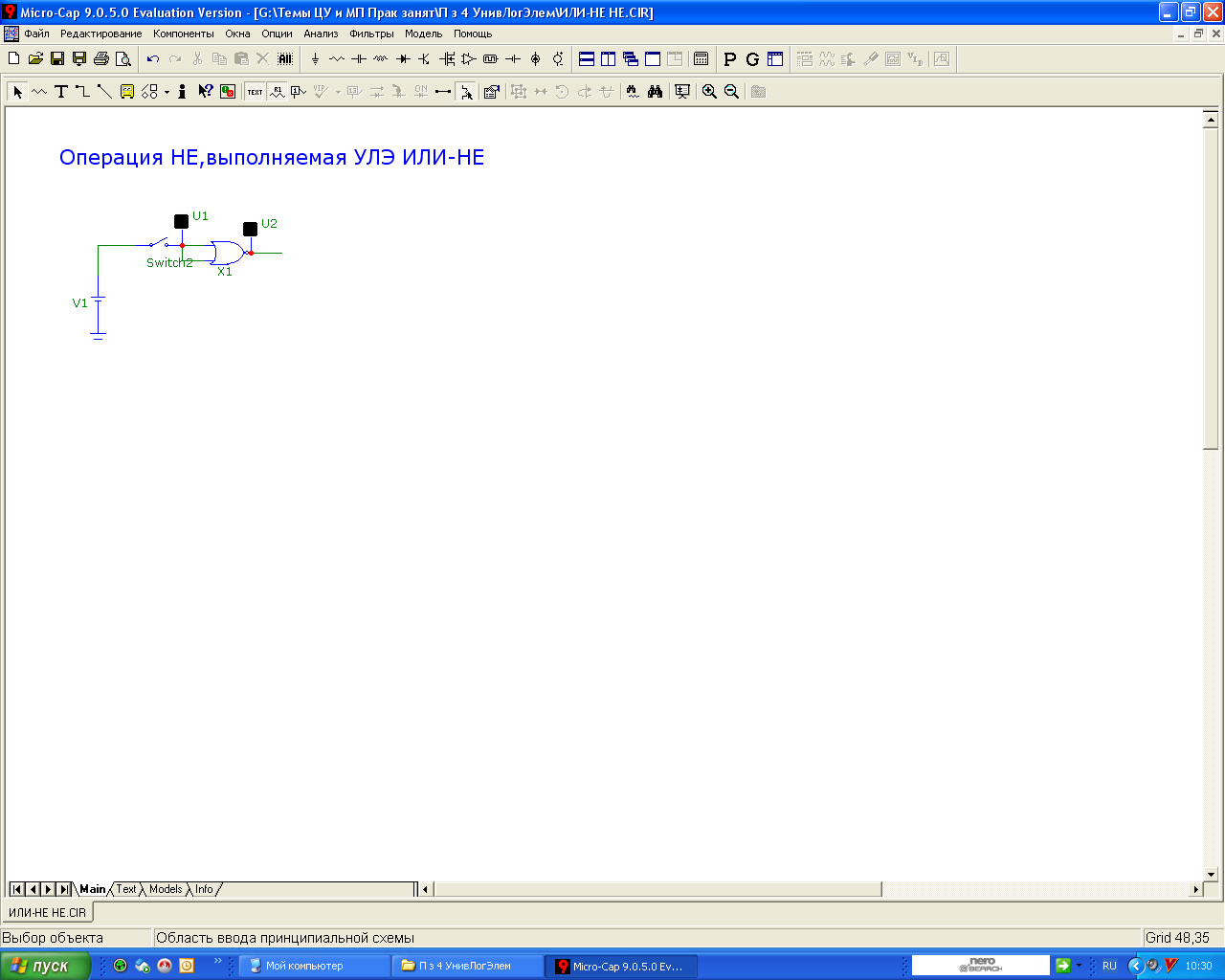


Рис. 11

Аналогично предыдущим пунктам, исследовать операции ИЛИ, И, НЕ, выполняемые универсальным логическим элементом «И-НЕ». Соответ-ствующие файлы имеют наименования: «И-НЕ ИЛИ», «И-НЕ И», «И-НЕ НЕ».

**Содержание отчета**

1. Титульный лист с названием работы.

2. Основные теоретические положения.

3. Схемы логических выражений и таблицы истинности.

4. Выводы.

**Занятие 5**

**по теме «Системы счисления»**

**Задание**

Число в 16-ричной системе счисления (табл. 1) представить в двоично-десятичной системе счисления (код 8421) и в десятичной системе счисления.

*Таблица 1*

|  |
| --- |
| Предпоследняя цифра шифра |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Последняя цифра шифра | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 3СH | 7DH | 2DH | 3CH | 3AH | 3BH | 41H | 57H | A8H | B7H |
| 1 | 8ВH | 5EH | 9EH | 9FH | A0H | A1H | A2H | A3H | A4H | A5H |
| 2 | A6H | A7H | A8H | A9H | AAH | ABH | ACH | ADH | AEH | AFH |
| 3 | B0H | B1H | B2H | B3H | B4H | B5H | B6H | B7H | B8H | B9H |
| 4 | BAH | BBH | BCH | BDH | BEH | BFH | 96H | 97H | 98H | 99H |
| 5 | 4AH | 5BH | 6CH | 7DH | 8EH | 9FH | 77H | 78H | 79H | 85H |
| 6 | E1H | E0H | DFH | DEH | DDH | DCH | DBH | DAH | D9H | D8H |
| 7 | EBH | EAH | E9H | E8H | E7H | E6H | E5H | E4H | E3H | E2H |
| 8 | F5H | FF4H | F3H | F2H | F1H | F0H | EFH | EEH | EDH | ECH |
| 9 | FFH | FEH | FDH | FCH | FBH | FAH | F9H | F8H | F7H | F6H |

**Указание к выполнению занятия 5**

Перед выполнением занятия 5следует изучить раздел 2 учебного пособия [1], в котором рассмотрены преобразования чисел из одной системы счисления в другую.

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Расчёты по занятию.

3. Выводы по занятию.

**Занятие 6**

**по теме «Формы представления чисел»**

**Задание**

Числа, представленные в табл. 1 в десятичной системе счисления, пере-вести в двоично-десятичную систему (код 8421), сложить алгебраически, предварительно представив отрицательные числа в дополнительном коде, а результат – в десятичной системе счисления.

*Таблица 1*

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Послед-няя цифра шифра | Числа в десятичной системе | | | | | | | | | |
| 0 | 33  -54 | -41  26 | 23  -28 | -60  89 | 63-28 | 59  48 | -67  28 | 69  -24 | -73  89 | -33  85 |
| 1 | -17  56 | -56  28 | 57  -79 | -61  88 | 64  -27 | 51  49 | -68  27 | 69  -25 | -75  85 | 85  34 |
| 2 | 24  -14 | 91  -74 | 81  45 | 92  -44 | -56  48 | -36  37 | 39  -49 | 56  -37 | 81  -41 | 34  -72 |
| 3 | 56  -91 | 49  36 | 58  -28 | 64  -32 | 91  -34 | 45  -26 | 23  -28 | 25  75 | 71  -24 | 99  -26 |
| 4 | -36  89 | 34  -65 | 24  -46 | 84  -28 | 71  -28 | 91  -27 | 98  -25 | 97  -23 | 95  -24 | 92  -25 |
| 5 | 49  -75 | 48  -28 | 47  -23 | 45  -22 | 43  -21 | 41  20 | 39  -18 | 37  -25 | 38  -85 | 43  -84 |
| 6 | 45  -23 | 47  -25 | 49  -99 | 51  -73 | 53  -48 | 43  -29 | -43  29 | -53  48 | -51  73 | -49  99 |
| 7 | -33  48 | -35  49 | -37  51 | -39  53 | 63  -39 | 39  -53 | 37  -51 | 37  -50 | 35  -47 | 39  -23 |
| 8 | 63  -23 | 65  -25 | 67  -24 | 68  -23 | 73  -28 | -73  28 | -68  23 | -67  24 | -65  25 | -69  23 |
| 9 | -27  55 | -28  56 | -29  57 | -30  58 | -31  59 | 33  -60 | 34  -61 | 35  -60 | 37  -62 | 39  -64 |

**Указание к выполнению занятия 6**

Перед выполнением занятия 6следует проработать раздел 2 учебного пособия [1], в котором рассмотрены формы представления чисел.

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Расчёты по занятию.

3. Выводы по занятию.

**Занятие 7**

**по разделу «Комбинационные устройства»**

**Теоретическая часть занятия**

В занятии производится минимизация комбинационных схем алгебраическим методом. Задача минимизации сводится к тому, чтобы после получения аналитической формы записи структурной формулы выполнить ее минимизацию, т. е. найти такую форму записи, которая потребует при реализации наименьшего числа элементов.

Типовыми приемами при этом можно считать:

1) прибавление одного или нескольких однотипных членов из числа имеющихся в первой стандартной форме, так как 

2) умножение отдельных членов функции на сумму , где *А* может быть как одной из переменных , так и функцией этих переменных; так как  то такое умножение не нарушает тождественности исходного и полученного выражений;

3) выделение слагаемых типа  путем применения закона дистрибутивности; после представления суммы в виде двух сомножителей, один из которых  выражение упростится;

4) использование законов поглощения и склеивания.

После проведения всех возможных преобразований получают функцию, не имеющую избыточных членов и не поддающуюся дальнейшей мини-мизации. Эту форму записи называют тупиковой.

**Задание**

Выполнить анализ комбинационного устройства, для чего:

а) определить функцию, реализуемую устройством в виде формулы;

б) представить функцию в виде таблицы;

в) упростить структуру устройства алгебраическим методом, если это возможно.

Варианты структурных схем в соответствии с последней цифрой шифра приведены ниже.

Варианты 0…4. Структурная схема устройства изображена на рис. 1, а элементарные блоки А1, А2, А3 для каждого из вариантов приведены в табл. 1.

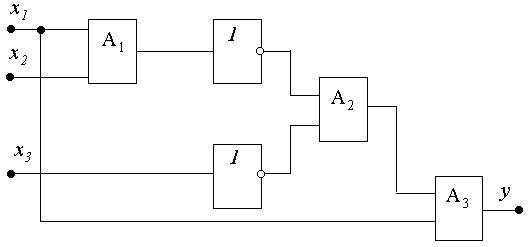


Рис. 1

*Таблица 1*

|  |  |  |  |
| --- | --- | --- | --- |
| № варианта | Блоки | | |
| А1 | А2 | А3 |
| 0 | или | или | или |
| 1 | или | или | и |
| 2 | или | и | или |
| 3 | и | или | и |
| 4 | и | и | или |

Варианты 5… 9. Структурная схема устройства изображена на рис. 2, а элементарные блоки А1, А2, А3 для каждого из вариантов приведены в табл. 2.

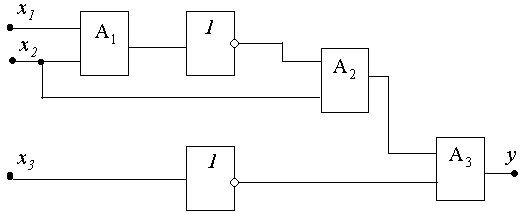


Рис. 2

*Таблица 2*

|  |  |  |  |
| --- | --- | --- | --- |
| № варианта | Блоки | | |
| А1 | А2 | А3 |
| 5 | или | или | или |
| 6 | или | или | и |
| 7 | и | или | или |
| 8 | и | и | или |
| 9 | и | и | и |

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Цель работы.

3. Аналитическая форма записи структурной формулы *Y*=*f*(*x1,x2,x3*).

4. Схемы комбинационных устройств и таблицы истинности.

1. Выводы по занятию.

**Занятие 8**

**по разделу «Комбинационные устройства»**

**Теоретическая часть занятия**

В занятии изучается графический метод минимизации комбинационных устройств. Графический метод минимизации логических функций использует карты Карно или диаграммы Вейча. Этот метод автоматизирует процедуру поиска склеивающихся слагаемых*.*

Карта Карно – это таблица, имеющая ячейки для всех возможных минтермов функции. Метод обычно применяется при небольшом числе минтермов (обычно не более 5 … 6). Для функции двух переменных карта Карно имеет вид рис. 1, *а* и содержит четыре клетки: каждая клетка предназначена для своего минтерма.

Если какой-то из этих минтермов в СДНФ присутствует, то в соответ-ствующей клетке ставится «1». Если какого-либо минтерма нет, то простав-

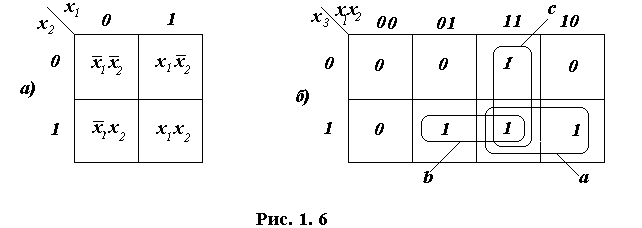


Рис. 1

ляется «0». Пользуясь этим правилом можно построить карту для схемы, зависящей от трех переменных.

Следует учитывать, что нумерация клеток в картах Карно по гори-зонтали и вертикали должна соответствовать коду Грея. Для некоторой комбинационной схемы карта Карно показана на рис. 1, *б*. Склеивания производятся между теми минтермами, которые записаны в виде «1» в соседних клетках карты по вертикали и горизонтали. На рис. 1, *б* имеются три таких контура: *a, b , c****.***

Контур «*a*» отражает склеивание минтермов:

 . (1)

Контур «*b*» отражает склеивание минтермов:

 . (2)

Контур «*с*» отражает склеивание минтермов:

. (3)

В результате выполненных операций склеивания из четырех слагаемых в данном примере остались три: *Y=x1x2* v *x2x3* v *x1x3*.

Таким образом, должна быть получена та же тупиковая форма, найденная алгебраическим методом в практическом занятии 7.

**Задание**

Выполнить минимизацию комбинационного устройства, для чего:

а) определить функцию, реализуемую устройством в виде формулы;

б) представить функцию в виде карты Карно;

в) упростить структуру устройства графическим методом минимизации Карно – Вейча, если это возможно.

Варианты структурных схем в соответствии с последней цифрой шифра (аналогично практическому занятию 7) приведены ниже.

Варианты 0…4. Структурная схема комбинационного устройства изображена на рис. 1, а элементарные блоки A1, A2, A3 для каждого из вариантов приведены в табл. 1.

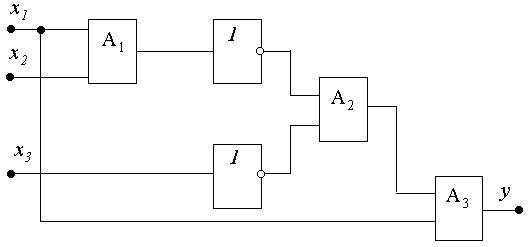


Рис. 1

*Таблица 1*

|  |  |  |  |
| --- | --- | --- | --- |
| № варианта | Блоки | | |
| A1 | A2 | A3 |
| 0 | или | или | или |
| 1 | или | или | и |
| 2 | или | и | или |
| 3 | и | или | и |
| 4 | и | и | или |

Варианты 5… 9. Структурная схема комбинационного устройства изображена на рис. 2, а элементарные блоки A1, A2, A3 для каждого из вариантов приведены в табл. 2.

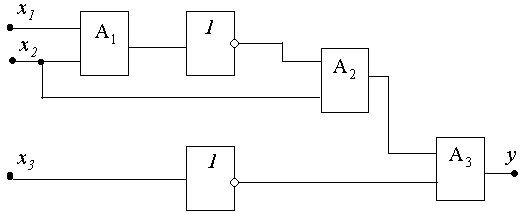


Рис. 2

*Таблица 2*

|  |  |  |  |
| --- | --- | --- | --- |
| № варианта | Блоки | | |
| A1 | A2 | A3 |
| 5 | или | или | или |
| 6 | или | или | и |
| 7 | и | или | или |
| 8 | и | и | или |
| 9 | и | и | и |

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Основные теоретические положения.

3. Склеивание по карте Карно.

4. Выводы по занятию.

**Занятие 9**

**по теме «Сумматоры и полусумматоры»**

**Теоретические сведения**

***Представление чисел в компьютере***

При записи числа в форме с фиксированной запятой разряды слева от запятой представляют целую часть числа, а справа – дробную часть числа (обычно только или целые, или дробные).

110101,1001001 - смешанная дробь,

110101 - целое число,

0,1001001 - дробь.

Разрядная сетка, содержащая n=m+s (m – число разрядов целой части, а s – дробной) двоичных разрядов, позволяет представить 2n различных неотрицательных чисел в диапазоне:

0 =< A =< 2m –1 – целые,

2-s =< A =< 1 –2-s – дробные,

2-s =< A =< 2m –2-s – смешанная дробь.

Для знака вводится дополнительный **знаковый** разряд, расположенный левее значащих:

“**0**” – положительные числа,

“**1**” – отрицательные.

Прямой код

В знаковый разряд кода записывается 0 (для положительных чисел) или 1 (для отрицательных чисел).

Отрицательное число, модуль которого равен 1011, представлено в прямом коде [А]пр = 1|1011.

Обратный код

Если число А < 0, то в знаковый разряд кода записывается 1, числовые разряды исходного числа инвертируются.

Число А= -(10000) в обратном коде [A]обр = 1|01111.

Дополнительный код

Если число А < 0, то в знаковый разряд кода записывается 1, числовые разряды исходного числа инвертируются, и к младшему числовому разряду добавляется 1.

Число А = -(10111) в дополнительном коде

[А]доп=1|01000+0|00001=1|01001.

В модифицированном дополнительном коде для представления знака числа отводятся два разряда (второй – перенос).

Запись положительных чисел в прямом обратном и дополнительном кодах одинакова: в знаковый разряд кода записывается 0, числовые разряды не изменяются. Положительное число, модуль которого равен 1011, представлено в прямом коде [А]пр = 0|1011.

Правила сложения

1. Вычитание заменяется сложением.

A+B+A+B

A-B=A+(-B)

-A+B=(-A)+B

-A-B=(-A)+(-B)

1. Слагаемые должны иметь одинаковое число разрядов.
2. Дописываем “0” слева и справа.
3. Преобразование в () – изменение знака.
4. Знаковые разряды участвуют в сложении, и знак результата фор-

мируется автоматически.

Результат представляется в том же коде, что и операнды.

***Пример***

А= -1210= - (01100)2; В = -1710= -(10001)2

[A]ОБР=1|10011; [B] ОБР =1|01110;

  1|10011

  1|01110

11|00001

  0|00001

  1|00010

[A+B]ОБР=1|00010= -(11101)2= -2910

[A]ДОП=1|10100; [B] ДОП =1|01111;

1|10100

1|01111

1|00011

[A+B] ДОП =1|00011= -(11101)2= -2910

Сумматор осуществляет арифметическое суммирование ***n***-разрядных кодов *A=*(*an-1,..,a0* ) и *B=*(*bn-1,..,b0*)*.* Правила сложения двух одноразрядных двоичных чисел:

0 + 0 = 0

0 + 1 = 1 + 0 = 1

1 + 1 = 0 и перенос 1 в старший разряд.

Устройство, реализующее эти правила, называется одноразрядным полусумматором и имеет два входа и два выхода.

Принципиальная схема полусумматора приведена на рис. 1.

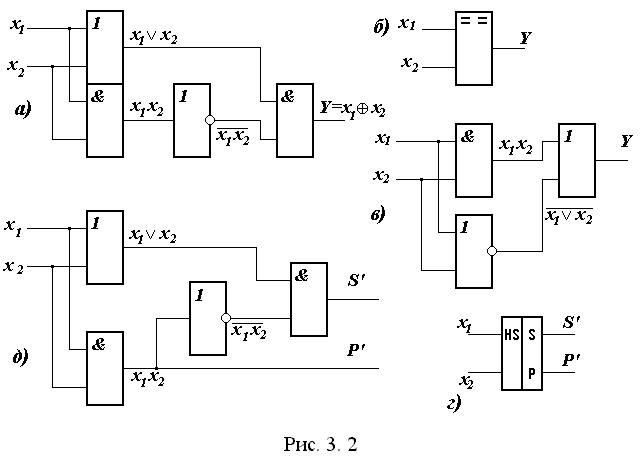


Рис. 1

Сложение трех одноразрядных чисел производится следующим образом:

0 + 0 + 0 = 0

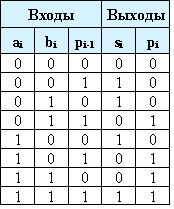
0 + 0 + 1 = 1

0 + 1 + 1 = 0 и перенос 1 в старший разряд

1 + 1 + 1 = 1 и перенос 1 в старший разряд.

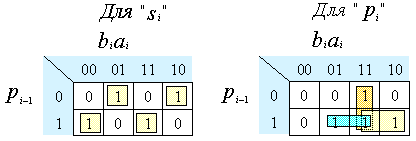
Устройство, реализующее эти правила, называется одноразрядным полным сумматором (ОПС) и имеет три входа и два выхода. Таблица истинности ОПС приведена в табл. 1.

*Таблица 1*



В табл. 1 *a*i, *b*i – одноименные двоичные разряды чисел *A* и *B*, *p*i-1 – перенос из предыдущего разряда, *s*i – частичная сумма по модулю два и *pi* – перенос в следующий разряд.

Составим карты Карно в соответствии с этой таблицей:



Выражение для *s*i имеет вид



В этом выражении используется запись логической функции исключающее “ИЛИ” в виде

.

Выражение в последней скобке необходимо преобразовать, используя сначала распределительный закон и правило свертки:



а затем – двойного отрицания и дважды соотношение двойственности:



С учетом последнего выражения получите выражение для *p*i.

Условное графическое изображение полного сумматора представлено на рис. 2, *а,* а принципиальная схема – на рис. 2, *б*.

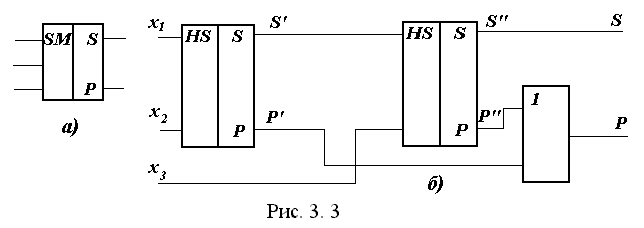


Рис. 2

Как видно из рис. 2, *б*, полный сумматор строится из двух полусумматоров и одного элемента, выполняющего операцию дизъюнкции двух переносов, возникающих в каждом полусумматоре. Поэтому полный сумматор также имеет два выхода: ***S*** – сумма и ***P*** – перенос.

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соот-ветствующую пиктограмму;

2. В появившемся окне выбрать файл «HS-полусумматор» и открыть его (рис. 3);

3. Выбрать режим Transient Analysis (Alt+1) – расчет переходных про-цессов;

4. Провести моделирование на ПК в соответствии с порядком выполнения занятия.

Схемная реализация полусумматора в программе MC9 DEMO пред-ставлена на рис. 3.

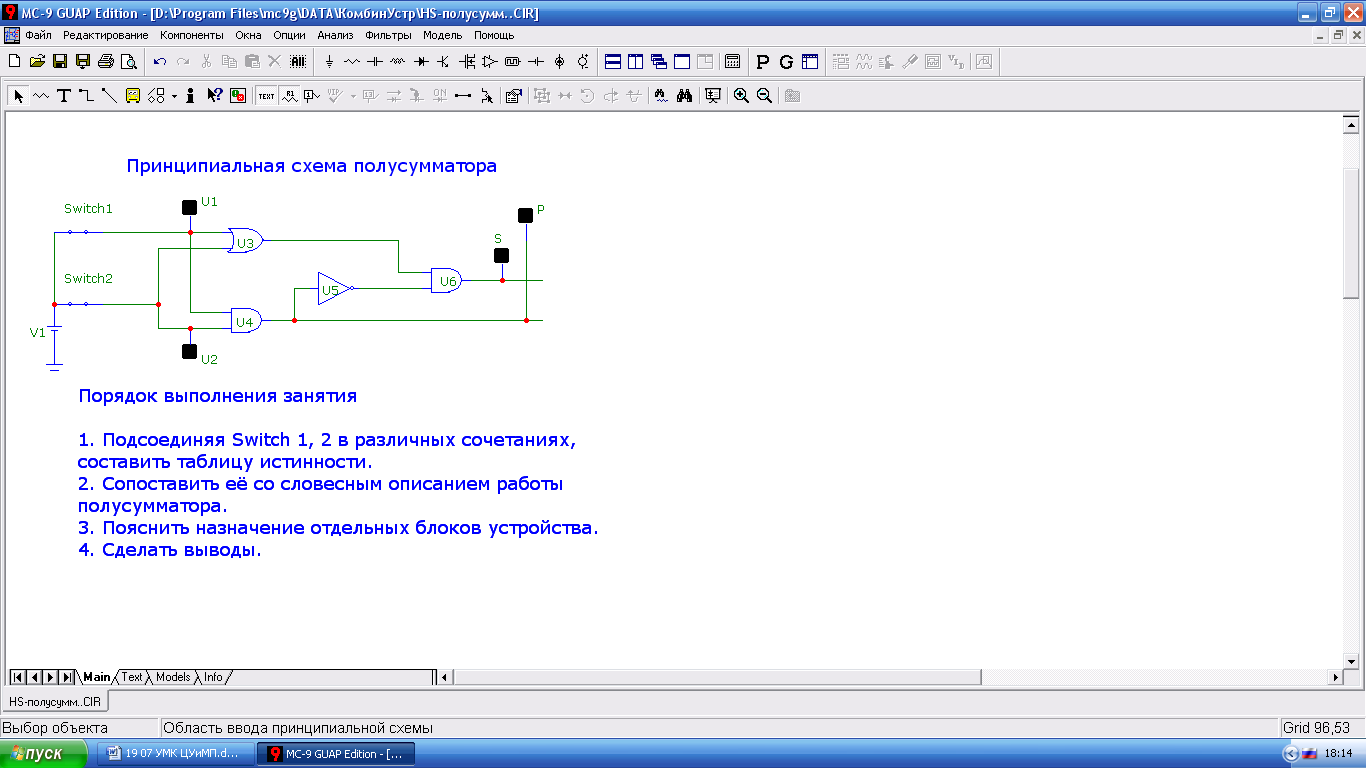


Рис. 3

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соот-ветствующую пиктограмму;

2. В появившемся окне выбрать файл «SM полн. сумматор» и открыть его (рис. 3);

3. Выбрать режим Transient Analysis (Alt+1) – расчет переходных процессов;

4. Провести моделирование на ПК в соответствии с порядком выполнения занятия.

Схемная реализация полного сумматора в программе MC9 DEMO пред-ставлена на рис. 4.

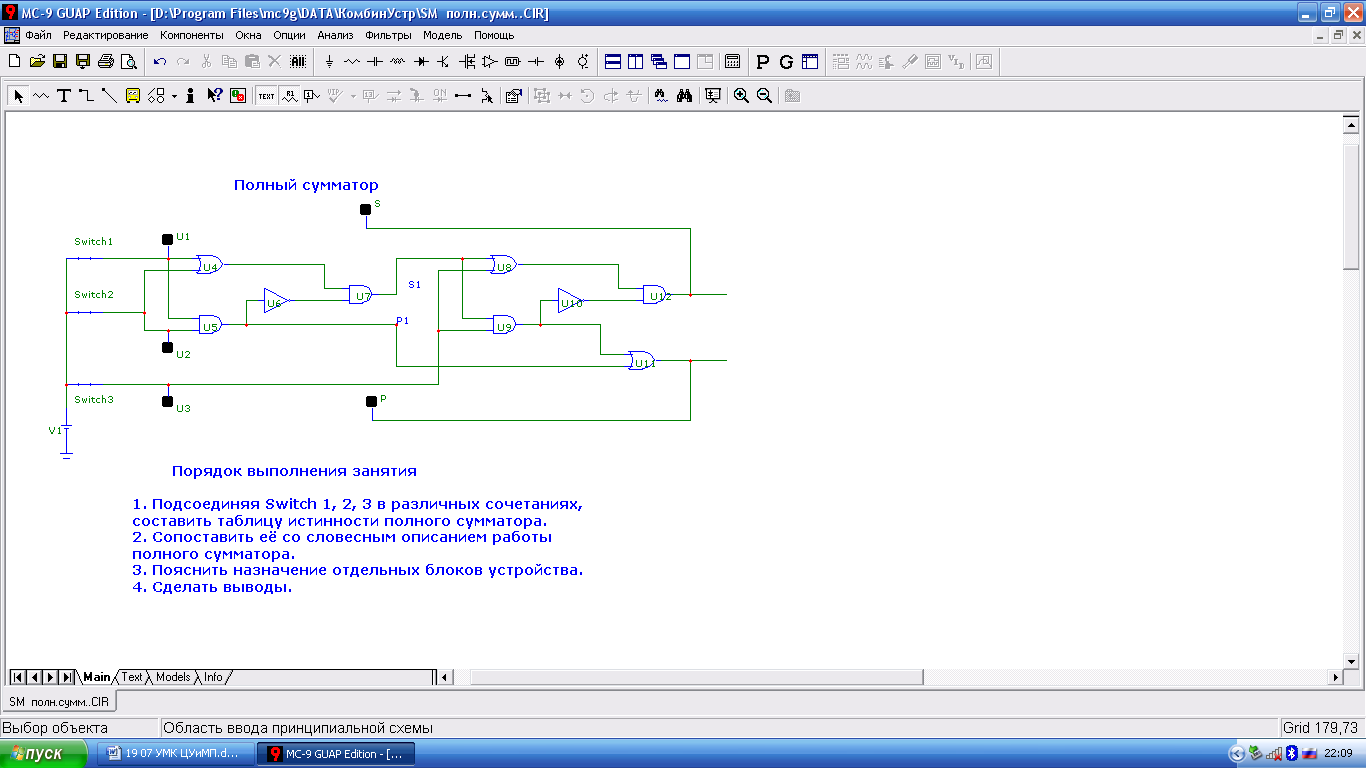


Рис. 4

**Задание**

1. Собрать схему одноразрядного полного сумматора, соответствую-щую полученным уравнениям, в среде МС9 (рис. 5).

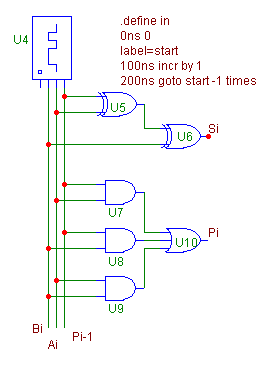


Рис. 5

2. Получить временные диаграммы, поясняющие работу одноразряд-ного полного сумматора (рис. 6).



Рис. 6

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Основные теоретические положения.

3. Аналитическая форма записи структурной формулы

*Y*=*f*(*x1, x2,…xn*).

4. Схемы одноразрядного полного сумматора, полусумматора и их таблицы истинности.

5. Выводы по занятию.

**Занятие 10**

**по теме «Устройства неравнозначности и равнозначности»**

**Теоретическая часть занятия**

***Устройство неравнозначности***

Это устройство с двумя входами *x1 , x2*и одним выходом *Y****,*** реализующим логическую функцию, называемую ИСКЛЮЧАЮЩЕЕ ИЛИ: *Y* ***=*** 1 только при несовпадении информационных значений входных сигналов.

Эта функция имеет специальное обозначение, как показано в выражении

,

и чаще называется «сумма по модулю 2».

Логическая схема, выполняющая эту функцию, представлена на рис. 1*.* Для этой функции существует ряд дополнительных тождеств, которые полезно использовать для упрощения логических выражений:







Если  то 

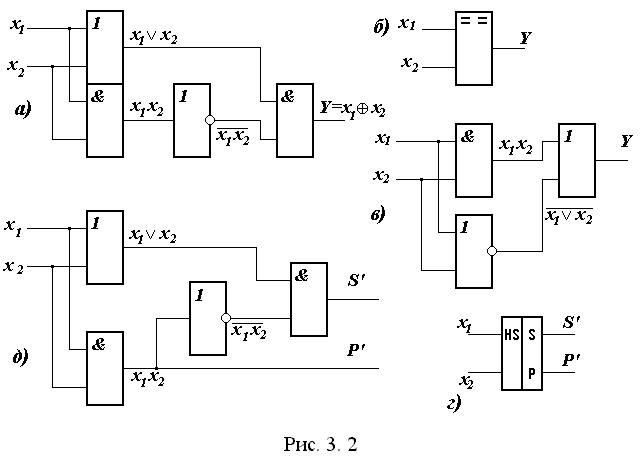


Рис. 1

***Устройство равнозначности***

Так называется устройство с двумя входами *x1* и *x2* и одним выходом *Y****,*** сигнал на выходе которого появляется только при совпадении информацион-ных значений входных сигналов.

Принципиальная схема устройства равнозначности приведена на рис. 2.

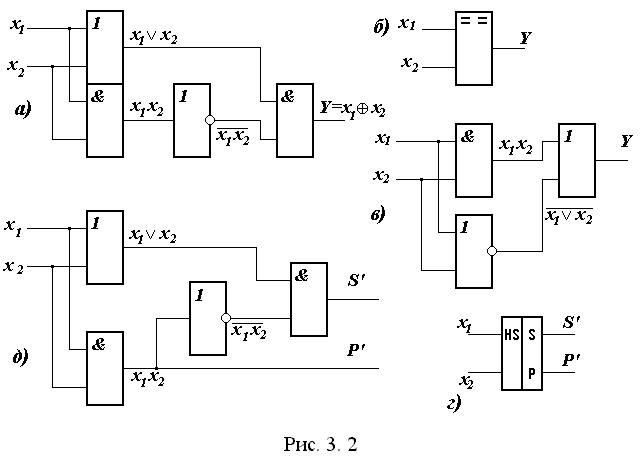


Рис. 2

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соот-ветствующую пиктограмму;

2. В появившемся окне выбрать файл «М2 неравнозначн.») и открыть его (рис. 3);

3. Выбрать режим Transient Analysis (Alt+1) – расчет переходных про-цессов;

4. Провести моделирование на ПК в соответствии с порядком выполнения занятия.

Схема устройства неравнозначности в программе MC9 DEMO представ-лена на рис. 3.

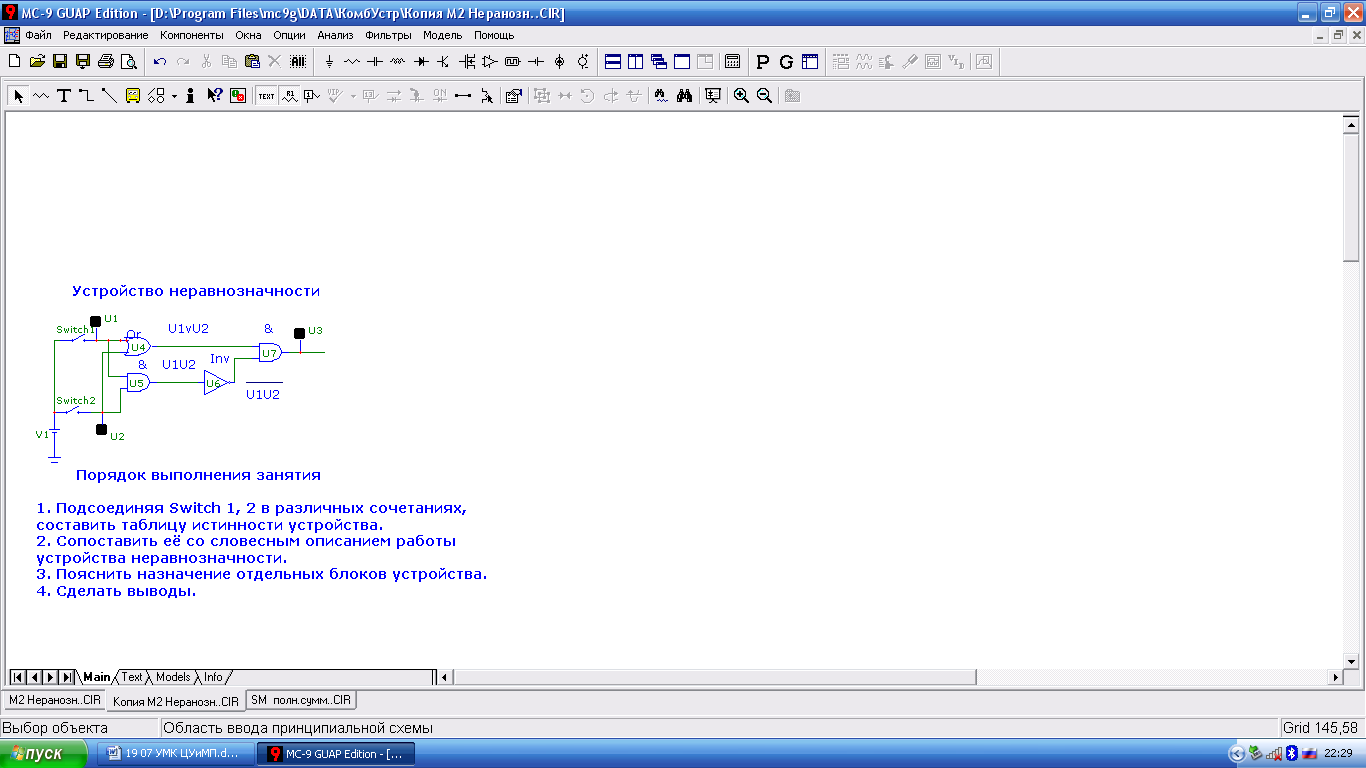


Рис. 3

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соот-ветствующую пиктограмму;

2. В появившемся окне выбрать файл «Равнозначн.») и открыть его (рис. 4);

3. Выбрать режим Transient Analysis (Alt+1);

4. Провести моделирование на ПК в соответствии с порядком выполнения занятия.

Схема устройства равнозначности в программе MC9 DEMO представ-лена на рис. 4.

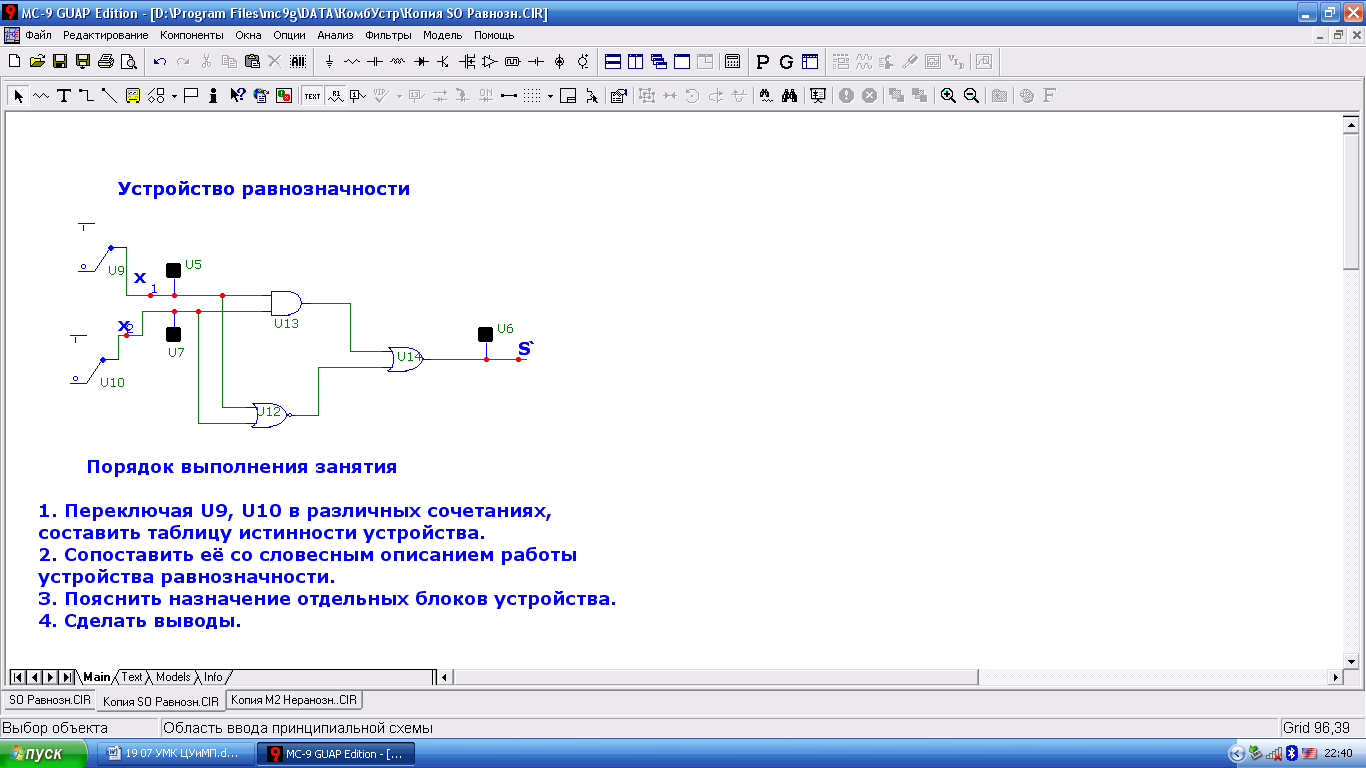


Рис. 4

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Аналитическая форма записи структурных формул.

3. Схемы устройств неравнозначности, равнозначности и таблицы ис-тинности.

4. Выводы по занятию.

**Занятие 11**

**по теме «Шифраторы, дешифраторы, кодопреобразователи»**

**Теоретическая часть занятия**

Дешифратор – это устройство, предназначенное для распознавания кодовых комбинаций. Дешифратор имеет число входов, равное значности принимаемого кода, и число выходов, равное возможному числу кодовых комбинаций. Дешифратор вырабатывает единичный сигнал на требуемом выходе только тогда, когда на вход поступает код числа, соответствующего этому выходу.

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соответ-ствующую пиктограмму;

2. В появившемся окне выбрать файл «Decoder») и открыть его (рис. 1);

3. Выбрать режим Transient Analysis (Alt+1);

4. Провести моделирование на ПК в соответствии с порядком выполнения занятия.

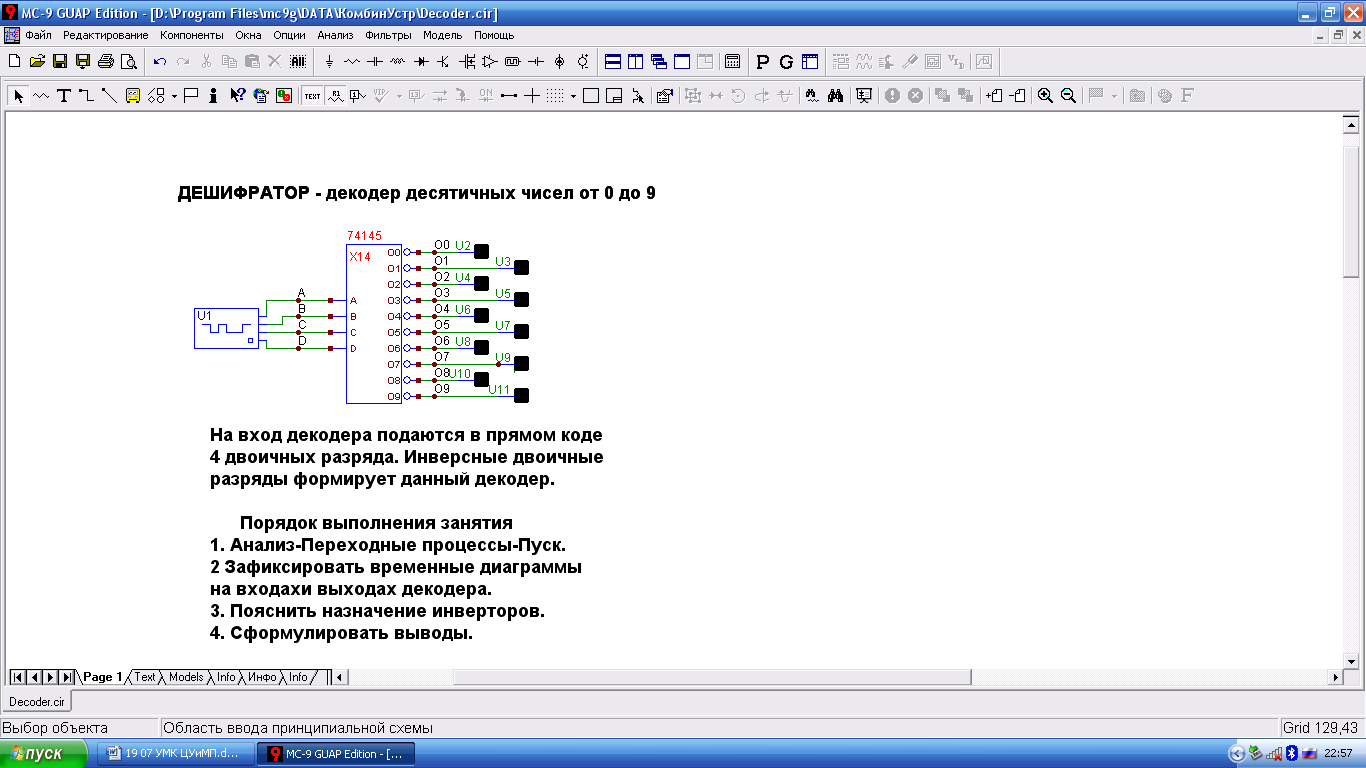


Рис. 1

**Содержание отчета**

1. Титульный лист с названием работы.

2. Осциллограммы на входах и выходах дешифратора.

3. Выводы.

**Занятие 12**

**по теме «Мультиплексоры, демультиплексоры, компараторы»**

**Теоретическая часть занятия**

***Мультиплексор*** – это комбинационная схема, имеющая ***m + 2m*** входов и один выход, где ***m*** – число адресных входов, а ***2m*** – число информационных входов мультиплексора. Основное назначение мультиплексора – это комму-тация ***2m*** сигналов со входа на один общий выход.

На рис. 1, *а* изображен четырехканальный мультиплексор, в котором использованы логические элементы ИЛИ – НЕ и И – ИЛИ (адресные входы ***x1*** и ***x2***). Промышленность выпускает мультиплексоры с различным числом адресных входов *m****=***2, 3и4. В случае, когда количество адресных входов *m****=***2***,*** мультиплексоры выпускаются сдвоенные, четырехканальные, со строби-рованием, условное графическое изображение которых приведено на рис. 1, *б* (микросхема 155КП2). Стробирующие входы *V1* и *V2*  используются для построения коммутаторов с *k*******2***m*** информационными входами (*k* ***=*** 2, 3, 4 ***…***) на основе 2m – канальных мультиплексоров.

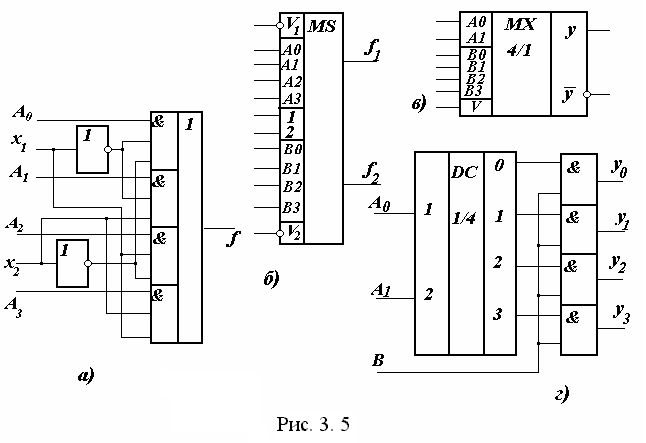


Рис. 1

***Демультиплексоры*** выполняют функцию, обратную функции мульти-плексора, т. е. коммутируют один входной сигнал на ***2m*** выходов. В качестве демультиплексоров могут быть использованы мультиплексоры, выполненные на основе аналоговых двунаправленных ключей. Например, в качестве демультиплексоров могут применяться микросхемы 564КП1 и 564КП2. В этом случае информационные входы *A****j*** являются выходами *f****i ,*** а выход *f* – входом *G****.***

Мультиплексоры и демультиплексоры – это коммутаторы логических сигналов. Демультиплексоры иногда называются распределителями. На передающей стороне используются мультиплексоры, а на приемной – демультиплексоры, т. е. совместное использование этих устройств позволяет осуществить временное разделение (уплотнение) каналов. Следует отметить, что в технической литературе встречаются обозначения мультиплексоров *MX* и *MS*. Кроме того, в структурных схемах иногда пишется и название устройства, например коммутатор с 4-х направлений на одно может быть обозначен, как на рис. 1, *в*.

Демультиплексор обычно обозначается тем же символом, что и дешифратор, т. е. *DC*. Иногда пишется и символ, обозначающий, на сколько выходов коммутируется входной сигнал, как показано на рис. 1, *г*, из которого видно, что при реализации демультиплексоров на выходах дешифраторов предусматриваются схемы И, одни из входов которых соединяются параллельно, образуя общий вход, на который подается входной информационный сигнал *B*. Логическое выражение, связывающее выходные переменные ***yi*** с входными *mi* и переменной *B*на информационном входе, можно представить выражением *yi=miB,* где *m****i =*** *A****1 …****A****p*** – полные правильные конъюнкции, т. е. по существу, минтермы на управляющих входах; *i* ***=*** 0***,*** 1 … 2*p* ***–*** 1 ;*p*– число управляющих переменных.

**Задание**

Собрать в среде МС 9 схему четырехканального мультиплексора по схеме на рис. 1, *а* и исследовать её работу в режиме «Переходные процессы».

**Порядок выполнения задания**

1. Запустить МС 9, после чего активизируется диалоговое окно **New Fi-le**, в котором нужно выбрать **Schematic** и нажать **ОК**. Для удобства рас-положения компонентов рекомендуется включить режим отображения сетки кнопкой с пиктограммой **Grid .**



2. С помощью мыши выбрать пункт меню **Component/Digital Primi-tives/Standard Gates/Nand Gates/Nand2** (логический элемент, выполняющий логическую функцию “И-НЕ” с двумя переменными). В открывшемся диалоговом окне **Attribute dialog box** задания параметров компонентов для параметра **TIMING MODEL** в правой части окна выбрать D0\_GATE.

3. Соединить компоненты проводниками.

4. Провести анализ схемы. Для этого в меню **Analysis** выбрать команду **Transient Analysis** (комбинация клавиш Alt+1).

5. Для того чтобы видеть изображение схемы, которое закрыто окном с результатами анализа, изменить взаимное расположение окон, нажав одну из кнопок.



6. Составить таблицы истинности анализируемой схемы, для которой существуют 4 возможных двоичных набора аргументов.

**Содержание отчета**

1. Титульный лист с названием работы.

2. Основные теоретические положения.

3. Схема логической функции и таблица истинности.

4. Временные диаграммы.

5. Выводы.

**Занятие 13**

**по теме «Асинхронные триггеры»**

**Теоретическая часть занятия**

***\_ Асинхронные триггеры*** – это простейшие КА, имеющие два выхода *Q* и *Q* и два входа  *S* и *R*. Входной сигнал *S* устанавливает триггер в положение *Q* ***=*** 1. Сигнал *R*переключает триггер в положение *Q* ***=*** 0. Случай *R* ***=*** *S* ***=*** 1 требует для каждой разновидности триггера отдельного рассмотрения, учитывающего особенности схемы.

Сигнал *R* ***=*** 1устанавливает на выходе *Q****=***0***.*** Поэтому при *S* ***=*** 0;*R* ***=*** 1;*Qn+1* ***=*** 0. Если при этом *Qn* ***=*** 1***,*** то триггер переключается, если *Qn* ***=*** 0***,*** то это состояние сохраняется. Совокупность входных сигналов *S = R* ***=*** 1не дает однозначного определения *Qn+1* , и на выходе получим неопределенность, которая обычно обозначается буквой Х.

***RS-триггер с инверсными входами***

Структурная формула этого триггера на элементах И – НЕ, представ-ленная в выбранном базисе, имеет вид, показанный на рис. 1.

Связь выходных сигналов устройства с входными вытекает из логии-ческой функции, выполняемой элементом «штрих Шеффера»:

 (1)

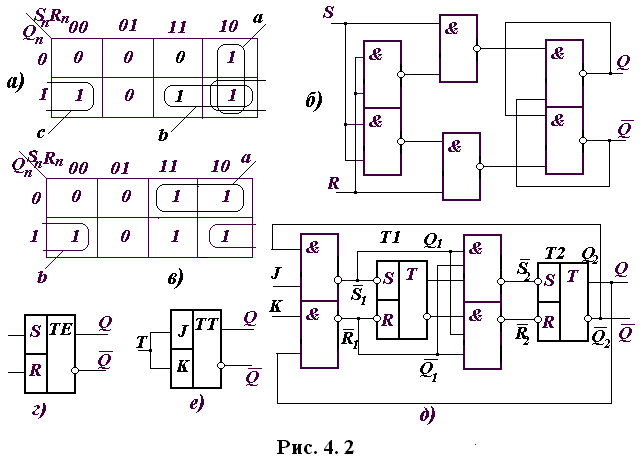


Рис. 1

Уравнения (1) называются уравнениями возбуждения. \_\_

Из всех возможных комбинаций устойчивых сочетаний *Q* и *Q* сочетания «00», и «11» следует исключить, так как не выполняется требование инверсности. Сочетание «00» неустойчиво, ему нет подобного, и оно исключается автоматически. Но остается еще сочетание «11», которое может быть устойчиво. Его необходимо исключить принудительно, запрещая сочетание на входах *S = R* ***=*** 1 . Это получается, если на входе выполняется равенство*SR* ***=*** 0.

***Е – триггер***

Полную структурную формулу этого триггера можно представить в виде  (2)

Выражение (1) можно реализовать на универсальных элементах И-НЕ.

Если составить уравнения возбуждения триггера в виде

,

то по обобщенной карте Карно можно определить, что для этого типа триггера запретные сочетания «00» и « 11» неустойчивы и автоматически исклю-чаются, а устойчивые состояния выходов всегда инверсны.

***JK-триггер***

Структурная формула *JK****-***триггера может быть получена в виде  (3)

Схема триггера, функционирующего в соответствии с (3), представлена на рис. 2. В эту схему входят два *RS****-***триггера с инверсными входами, выполненные также на элементах И – НЕ. Эти внутренние триггеры имеют собственные входы *R,S,* на которые подаются сигналы *R1, S1* для Т1 и *R2, S2* для Т2. Чтобы различать входные сигналы триггера в целом и входные сигналы «внутренних» триггеров, принято входы этого триггера обозначать буквами *J,K*. Вход *J* имеет то же значение, что и вход *S* , а вход *K* – то же значение, что и вход *R*. С учетом последнего замечания уравнение (2) можно представить в виде  (4)

На основании (4) можно построить таблицу переключений триггера. *JK-*триггер не имеет запрещенных сочетаний входных сигналов *J* и *K*. Это делает его похожим на *Е****-***триггер, но, в отличие от *Е****-***триггера, при *J=K****=***1 этот триггер изменяет свое состояние на инверсное, т. е. .

Это свойство позволяет *JK-*триггеру работать в счетном режиме. Для этого необходимо объединить входы *J, K,* и получим счетный триггер, или *Т****-***триггер. Этот тип триггера имеет более широкие возможности для использования, так как это – универсальный триггер.

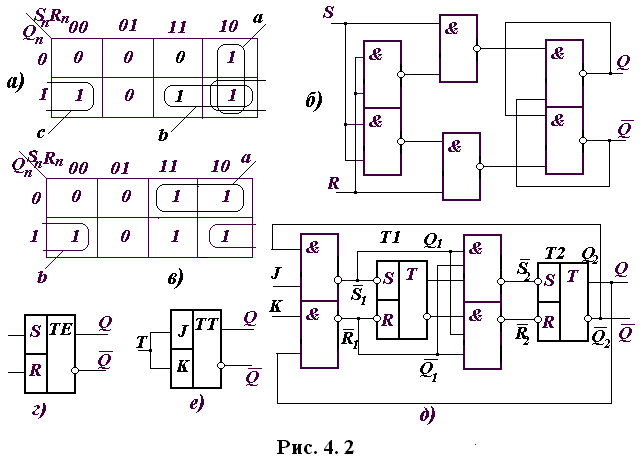


Рис. 2

**Задание**

Собрать в среде МС 9 схемы *RS*-триггера с инверсными входами и *JK-*триггера. Исследовать их работу в режиме «Переходные процессы», подключив входы триггеров к генераторам сигналов.

**Порядок выполнения задания**

1. Запустить МС 9, после чего активизируется диалоговое окно **New File**, в котором нужно выбрать **Schematic** и нажать **ОК.**

2. С помощью мыши выбрать пункт меню **Component/Digital Primi-tives/Standard Gates/Nand Gates/Nand2.**

3. В открывшемся диалоговом окне **Attribute dialog box** задания пара-метров компонентов для параметра **TIMING MODEL** в правой части окна выбрать D0\_GATE.

4. Соединить компоненты проводниками.

5. Провести анализ схемы. Для этого в меню **Analysis** выбрать команду **Transient Analysis** (комбинация клавиш Alt+1). Зафиксировать временные диаграммы входных и выходных сигналов триггеров.

6. Для того чтобы видеть изображение схемы, которое закрыто окном с результатами анализа, изменить взаимное расположение окон, нажав одну из кнопок.



7. Составить таблицу переключений анализируемой схемы.

**Содержание отчета**

1. Титульный лист с названием работы.

2. Основные теоретические положения.

3. Схемы *RS***-**триггера с инверсными входами, *JK-*триггера и таблицы переключений.

4. Временные диаграммы сигналов.

5. Выводы.

**Занятие 14**

**по теме «Синхронные триггеры»**

**Теоретическая часть занятия**

***Синхронный RS-триггер (RSC-триггер)***. Основа схемы – это асин-хронный *RS-*триггер с инверсными входами (элемент D3 на рис. 1). Управ-ляющие сигналы на каждый из входов триггера подаются через элементы И – НЕ (элементы D1, D2 на рис. 1).

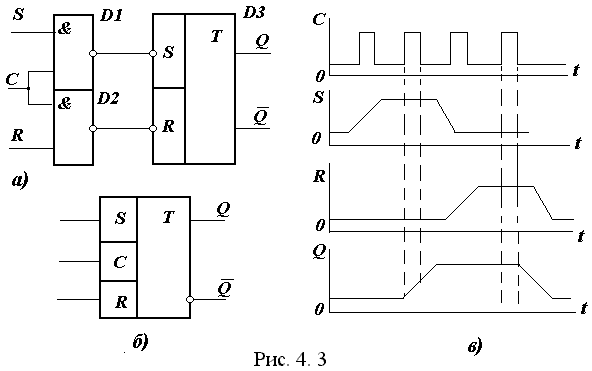
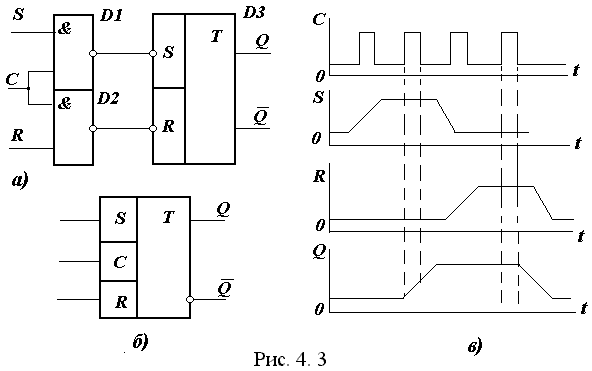


Рис. 1



Временные диаграммы работы триггера представлены на рис. 2.

Элемент D1 обеспечивает совпа-дение синхроимпульсов *С* и устано-вочных сигналов *S*, т. е. временное стробирование сигналов *S*, а элемент D2 стробирует по времени сигналы *R*.

Уравнение асинхронного *RS****-*** триггера с инверсными входами для

Рис. 2

синхронного *RSC-* триггера приобретает вид  (1)

\_

При этом нужно иметь в виду, что входным сигналом *S* в этом случае будет выходной сигнал элемента D1.

***D – триггер (триггер задержки).*** В синхронных триггерах в промежутке между синхроимпульсами выходные напряжения не изменяются. Это позволяет создать на основе синхронизируемых *RS-*триггеров триггеры задержки, или *D-*триггеры

Логическое уравнение этого триггера можно получить, если в (1) заменить сигнал *S* на сигнал *D* , а сигнал *R* – на инверсию сигнала *D*:

 (2)

Воспользовавшись правилом Де Моргана, (2) можно представить в более удобной форме, чтобы реализовать *D-*триггер на логических элементах И – НЕ :

 (3)

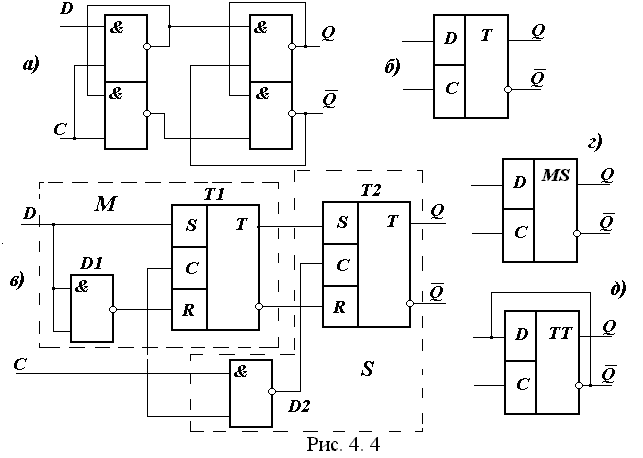


Рис. 3

На рис. 3изображена схема *D-*триггера, в основе работы которого лежит то, что он принимает информацию по входу *D*. Информация записывается в триггер при поступлении тактирующего импульса, а на выходе появляется в момент окончания тактирующего импульса. Эта информация может быть использована, когда придет следующий тактирующий импульс, т. е. в следу-ющем такте. Это и объясняет, почему *D-*триггер – элемент задержки на один такт.

**Задание**

Собрать в среде МС 9 схемы синхронного *RS*-триггера (*RSC-*триггера) и *D-*триггера. Исследовать их работу в режиме «Переходные процессы», подключив их входы к генераторам сигналов.

**Порядок выполнения задания**

1. Запустить МС 9 DEMO.

2. Собрать в среде МС 9 схемы синхронного *RS*-триггера (*RSC-*триггера) и *D-*триггера.

3. Провести анализ схемы. Для этого в меню **Analysis** выбрать команду **Transient Analysis** (комбинация клавиш Alt+1). Зафиксировать временные диаграммы входных и выходных сигналов триггеров.

4. Составить таблицы переключений анализируемых схем.

5. Представить результаты анализа в виде таблицы истинности на схеме, переключив МС в режим ввода текста, нажав CTRL+Tили кнопку **TEXT** панели инструментов



**Содержание отчета**

1. Титульный лист с названием работы.

2. Основные теоретические положения.

3. Схемысинхронного *RS*-триггера (*RSC*-триггера), *D*-триггера и таб-лицы истинности.

4. Временные диаграммы сигналов.

5. Выводы.

**Занятие 15**

**по теме «Регистры и счетчики»**

**Теоретическая часть занятия**

Счет импульсов – это одна из наиболее распространенных операций, выполняемых в цифровых устройствах. Счетчики можно классифицировать по разным признакам. Наиболее общая классификация – это три группы счетчиков: суммирующие, вычитающие и реверсивные, т. е. суммирующие или вычитающие в зависимости от управляющих сигналов.

Можно отметить два основных способа построения счетчиков – на основе триггеров и на основе регистров. При этом основными характеристиками будут модуль счета *m*, т. е. число, характеризующее количество устойчивых состояний счетчика, и быстродействие, которое характеризуется максимальной частотой поступления счетных импульсов.

***Двоичный счетчик с последовательным переносом.*** Этот счетчик относится к счетчикам суммирующего действия. Счетчик выполнен на синхронных *JK****-*** триггерах с асинхронным сбросом (рис. 1). Эту операцию выполняют входы *&* и *R0* . Особенностью счетчика является то, что если счет идет до 16 импульсов, то выход *Q1*следует соединить со входом *С2* , а считаемые импульсы подавать на вход *С1*. Переменный коэффициент счета – 2, 4, 8, 16 – обеспечивается на выходах *Q1 , Q2 , Q3 , Q4 .*

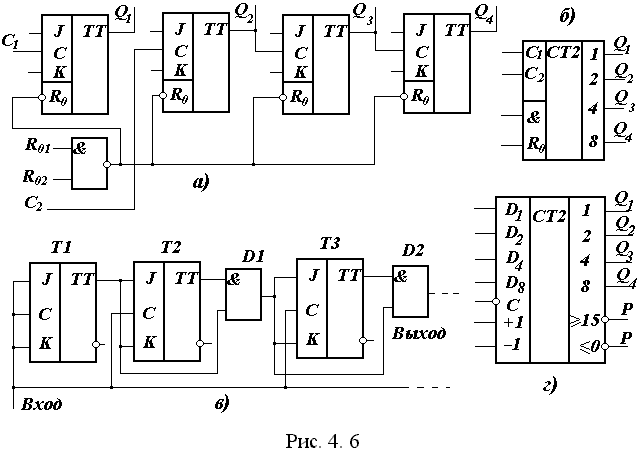
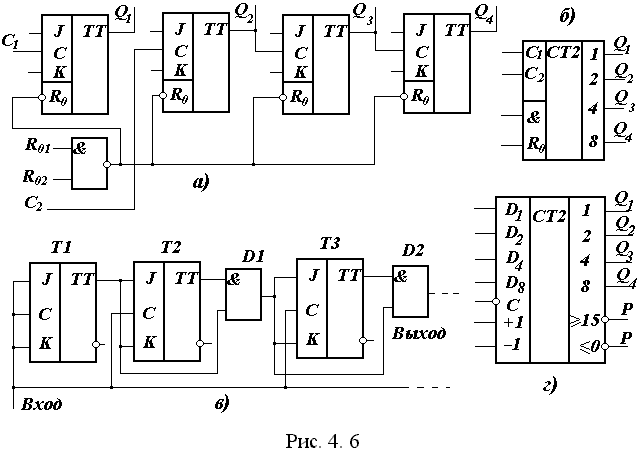


Рис. 1

Быстродействие счетчиков с последовательным переносом харак-теризуется величиной задержки, зависящей от количества последовательно включенных триггеров: 

***Двоичные счетчики с параллельным переносом.*** Параллельный пере-нос применяется для повышения быстродействия двоичных счетчиков. Здесь используется известная закономерность двоичных чисел: если к разряду с нулем прибавляется единица, то она будет в результате, а если к разряду с единицей прибавить единицу, то в результате получим нуль, а единица перемещается в следующий разряд. Логическая схема счетчика с параллель-ным переносом представлена на рис. 2.



↑***Вход*** Рис. 2

Наличие элементов D1, D2 в этом счетчике объясняется тем, что триггер Т1 срабатывает от каждого счетного импульса на входе. При этом *Q1 = J = K* ***=***1для триггера Т2 после первого импульса. Следовательно, Т2 сработает от второго счетного импульса *Q****2*** = 1***;*** *Q1 =* 0***.*** От третьего счетного импульса Q1 ***=*** *J = K=*1для Т3, который сработает от четвертого импульса и т. д. Поэтому при одном и том же числе счетных разрядов предельная частота счетчика с параллельным («сквозным») переносом будет выше, чем в счетчике после-довательного действия, и будет равна



где *tП* ***–*** задержка в срабатывании триггера;

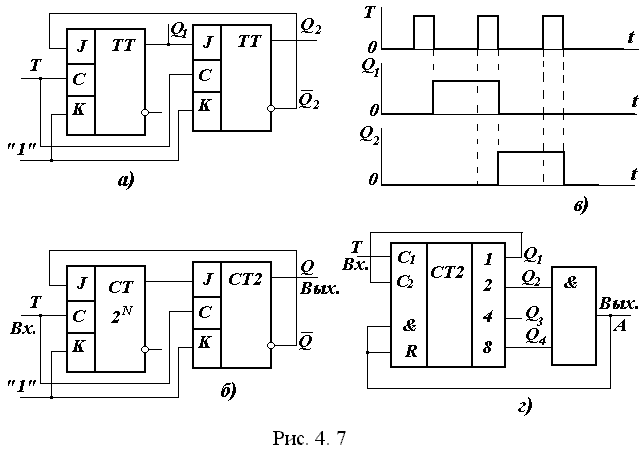
*tЗП*– задержка в элементах D1 , D2;

*tП******* *tЗП*.

***Реверсивный счетчик.*** Этот тип счетчика выполняет как операцию суммирования, так и вычитания, а также предварительную запись инфор-мации.

***Счетчики с произвольным модулем счета.*** К этой группе относятся счетчики, модуль счета которых не равен целой степени числа 2. Чаще всего такие счетчики строятся на регистрах сдвига с обратными связями.

Рассмотрим построение счетчика для случая коэффициента пересчета *N****=***3. Для его построения потребуется взять за основу двоичный счетчик с числом разрядов *m* ***=* [***log 2 N***]**, где выражение в квадратных скобках – ближайшее целое число, большее величины *m*. В рассматриваемом случае log**2**3 = 1,6 ***,*** т. е. потребуются два триггера.



***б)***

Рис. 3

Если в исходном состоянии выходы триггеров (рис. 3, *а* и *б*) *Q1=Q2=*0, то первый входной импульс установит счетчик в положение *Q1 =* 1*; Q2 =* 0*;*второй – в состояние*Q1 =* 0*; Q2 =* 1*,*а третий – снова в исходное состояние *Q1=Q2 =*0*.* На вход *K*постоянно подается логическая «1».

Построение счетчика *N =* 3 выполнено как бы путем увеличения на 1 коэффициента пересчета *N =* 2*.*По этой методике можно построить общий алгоритм получения недвоичного счетчика по принципу 2***N* +** 1 с помощью дополнительного счетчика СТ2.

***Декадный счетчик.*** Счетчик с коэффициентом пересчета *N =* 10носит название декадного. Такой коэффициент пересчета применяется сравнительно часто. Его удобно организовать на стандартной микросхеме К155ИЕ5 с коэффициентом пересчета *N =* 16. Для получения требуемого модуля счета необходимо при обнаружении числа 1010 (10102) сбрасывать счетчик в нулевое положение и начинать счет сначала, как показано на рис. 4.

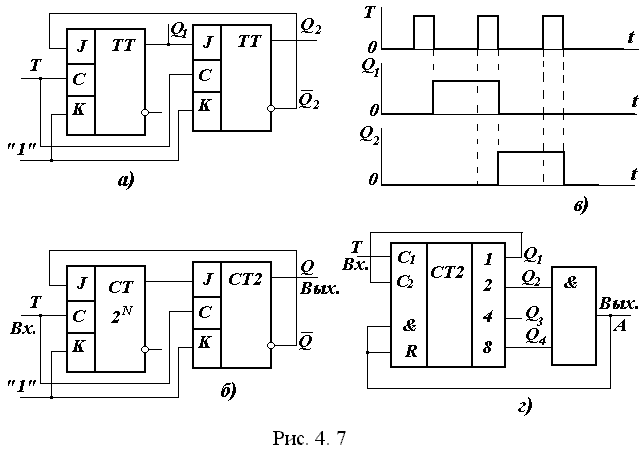


Рис. 4

Если построить для этого случая карту Карно, то искомое число 10102 будет получаться, когда , или с учетом факультативных условий *A = Q2 Q4* .

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соответ-ствующую пиктограмму;

2. В появившемся окне выбрать файл «OLS Counter 4р» и открыть его (рис. 5);

3. Выбрать режим Transient Analysis (Alt+1);

4. Провести моделирование на ПК в соответствии с порядком выполнения занятия.

Схемная реализация двоичного счетчика с последовательнымпереносом в программе MC9 DEMO представлена на рис. 5.

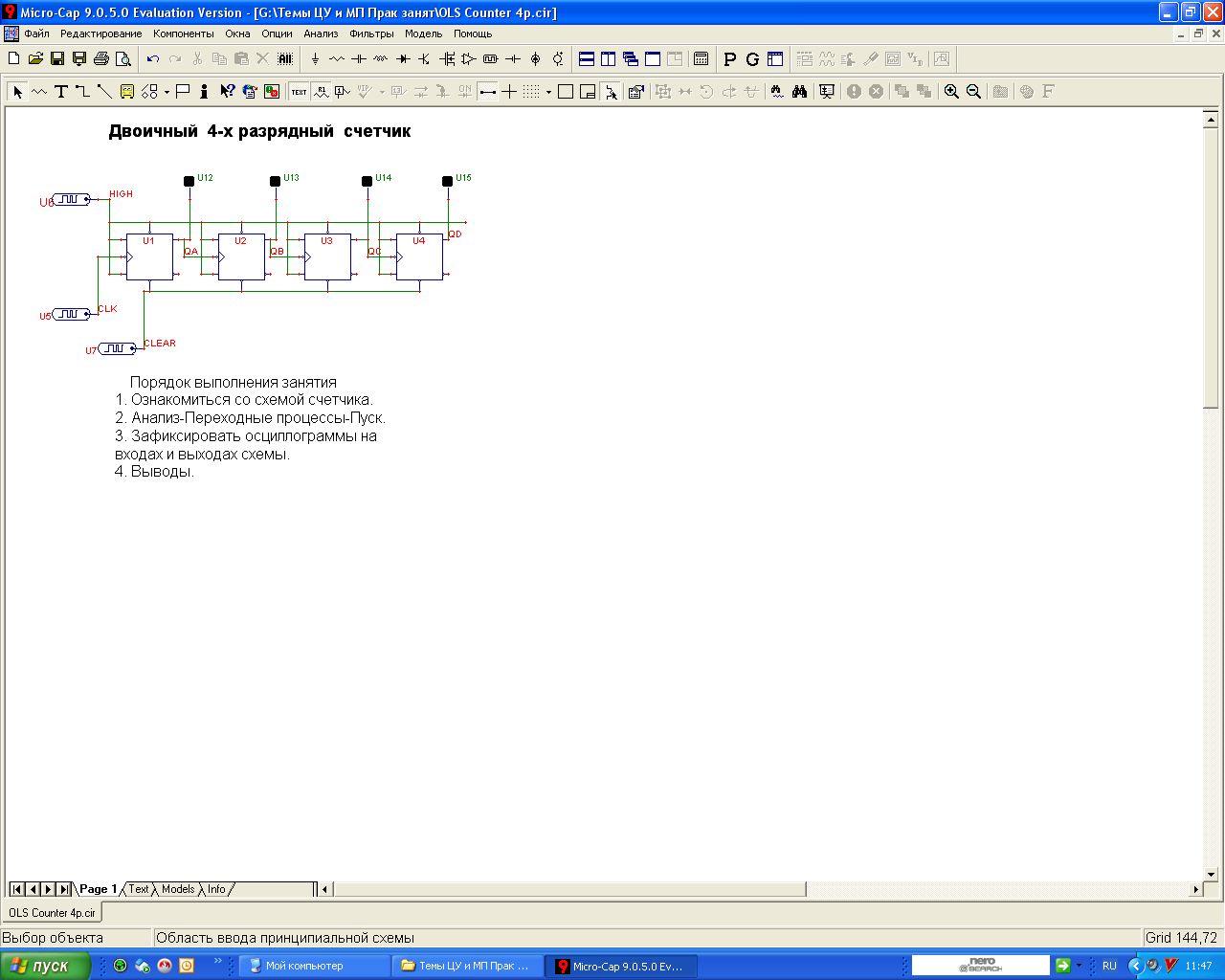


Рис. 5

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соответствующую пиктограмму;

2. В появившемся окне выбрать файл «Декад счет» и открыть его (рис. 6);

3. Выбрать режим Transient Analysis (Alt+1);

4. Провести моделирование на ПК в соответствии с порядком выполнения занятия.

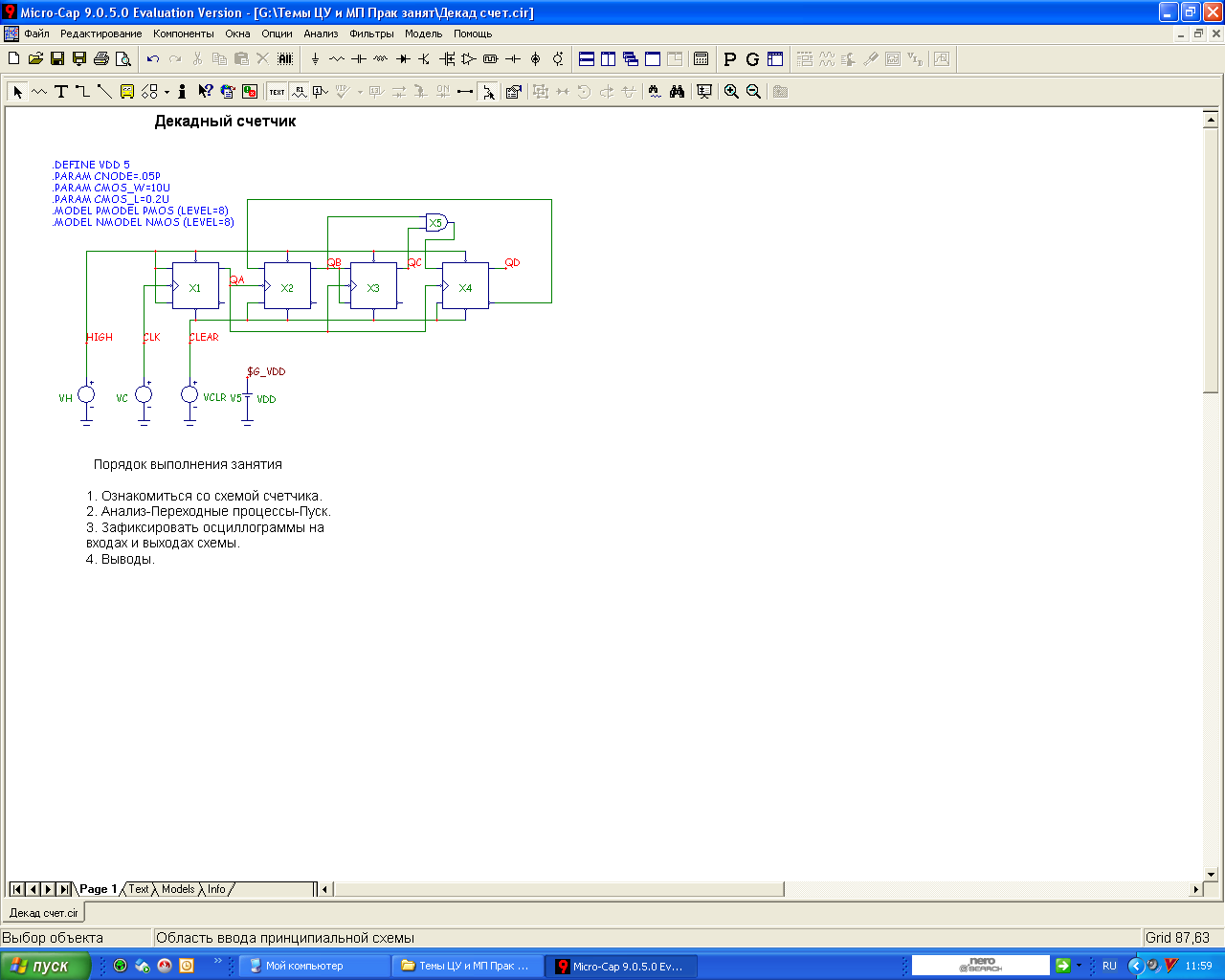


Рис. 6

**Содержание отчета**

1. Титульный лист с названием работы.

2. Основные теоретические положения.

3. Схемыдвоичного счетчика с последовательнымпереносом и декад-ного счетчика.

4. Временные диаграммы.

5. Выводы.

**Занятие 16**

**по теме «Аналого–цифровые преобразователи (АЦП)»**

**Теоретическая часть занятия**

Преобразования сигналов из аналоговой формы в цифровую выпол-няются в устройстве, называемом аналого-цифровым преобразователем (АЦП).

В АЦП обычно выделяются следующие процессы: дискретизация, кван-тование и кодирование.

В процессе дискретизации из непрерывного во времени сигнала выбираются отдельные его значения, соответствующие моментам времени, следующим через определенный временной интервал *Т* (на рис. 5.1 моменты *t0, t1* …). Интервал *Т* называется тактовым интервалом, а моменты *t0,t1,* … , в которые берутся отсчеты, - тактовыми моментами времени.

Дискретные значения сигнала обычно отсчитываются с таким малым тактовым интервалом *Т* , чтобы по ним можно было восстановить сигнал в аналоговой форме с необходимой точностью.

***Квантование и кодирование.*** При выполнении этих операций создается сетка уровней квантования сдвинутых друг относительно друга на величину Δ, называемую шагом квантования. Полученные в результате дискретизации значения исходного аналогового напряжения заменяются ближайшими к ним уровнями квантования.

Этот процесс носит название операции квантования,которая по существу состоит в округлении значений аналогового напряжения, выбранных в тактовые моменты времени. Округление приводит к ошибкам (шум квантования). При проектировании АЦП стремятся снизить шум квантования до предельно малой величины.

Другая важная операция, выполняемая при АЦП, – это кодирование*.* Смысл этой операции состоит в том, что при округлении значений напряжения при квантовании получаются числа, соответствующие уровням квантования. Эта последовательность чисел представляется двоичным кодом.

Таким образом,АЦП производит преобразование аналоговой величины, например мгновенное значение напряжения, в цифровой код. Это происходит при сравнении этого напряжения с эталонным, как показано на рис. 1, *а*.

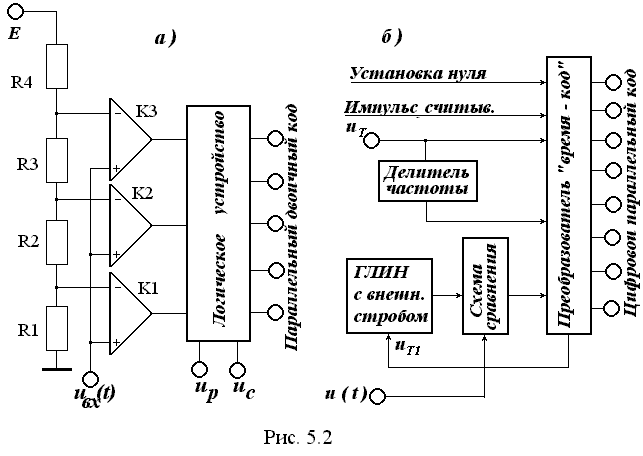


Рис. 1

На этом рисунке показан АЦП параллельного типа, в котором датчик уровней эталонных напряжений образуется при подаче напряжения на прецизионный делитель, состоящий из цепочки резисторов R1 … R4. Полученные эталонные напряжения поступают на компараторы К1 … К3. Если сигнал *uвх*(*t*) оказывается больше опорного, то происходит срабатывание компаратора: на его выходе получается логическая «1».

Логическое устройство в этой схеме записывает выходные сигналы компараторов в регистр (при наличии сигнала регистрации *u****p***), преобразует сигнал логической «1» в регистре в двоичный код с того компаратора, у которого был наибольший уровень опорного напряжения, и передает двоичный код на выход (при наличии сигнала списывания *u****c***).

АЦП параллельного типа отличаются достаточно высоким быстро-действием, так как опорные напряжения существуют у них постоянно, и на их формирование не расходуется дополнительное время. Но точность такого АЦП ограничена, так как ограничено число каналов. Требуется, чтобы разница между соседними уровнями квантования была существенно больше взаимной нестабильности порогов срабатывания соседних компараторов.

На рис. 1, *б* показан АЦП с промежуточным преобразованием напря-жения во временной интервал. Сам преобразователь «время - код» показан на рис. 2, а временная диаграмма, поясняющая его работу – на рис. 3.

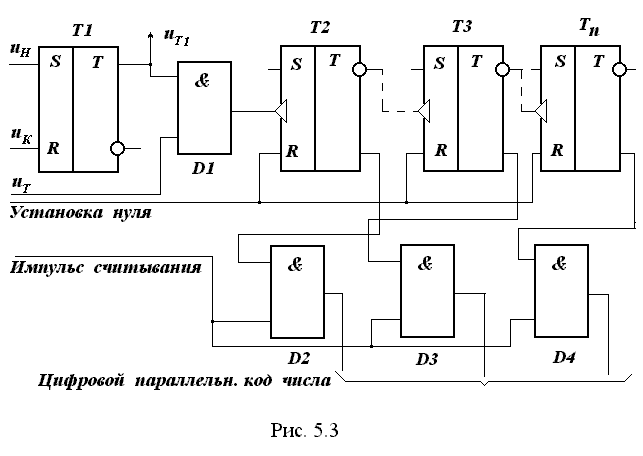


Рис. 2

Работу преобразователя «время - код» можно кратко описать следующим образом. Тактовые импульсы *u****Т*** поступают на делитель частоты, на выходе которого получаются импульсы начала *u****Н*** для запуска преобразователя «время - код». При этом на выходе триггера *Т1* формируется сигнал логической «1», который служит стробом для ГЛИН (сигнал *u****T1***). При формировании линейно нарастающего напряжения происходит его сравнение в компараторе (схема сравнения) с исследуемым напряжением, и в момент равенства этих напряжений вырабатывается сигнал конца (*u****к***), сбрасывающий триггер *Т*1 в ноль.

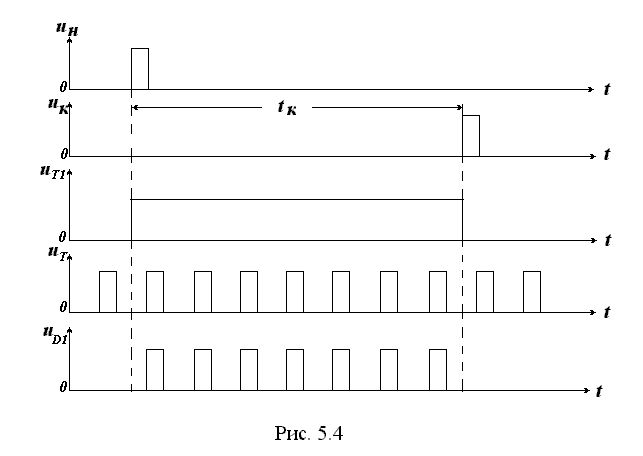


Рис. 3

На время действия строба работает счетчик импульсов на триггерах *Т2,**Т3 … Тn* ***,*** сигналы с которых поступают на схемы совпадения *D2, D3 …Dn .* При поступлении импульса считывания на выходе схем совпадения формируется параллельный код числа.

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соответ-ствующую пиктограмму.

2. В появившемся окне выбрать файл «АЦП 16 бит» и открыть его (рис. 4).

3. Выбрать режим Transient Analysis (Alt+1).

4. Провести моделирование на ПК в соответствии с порядком выполнения занятия.

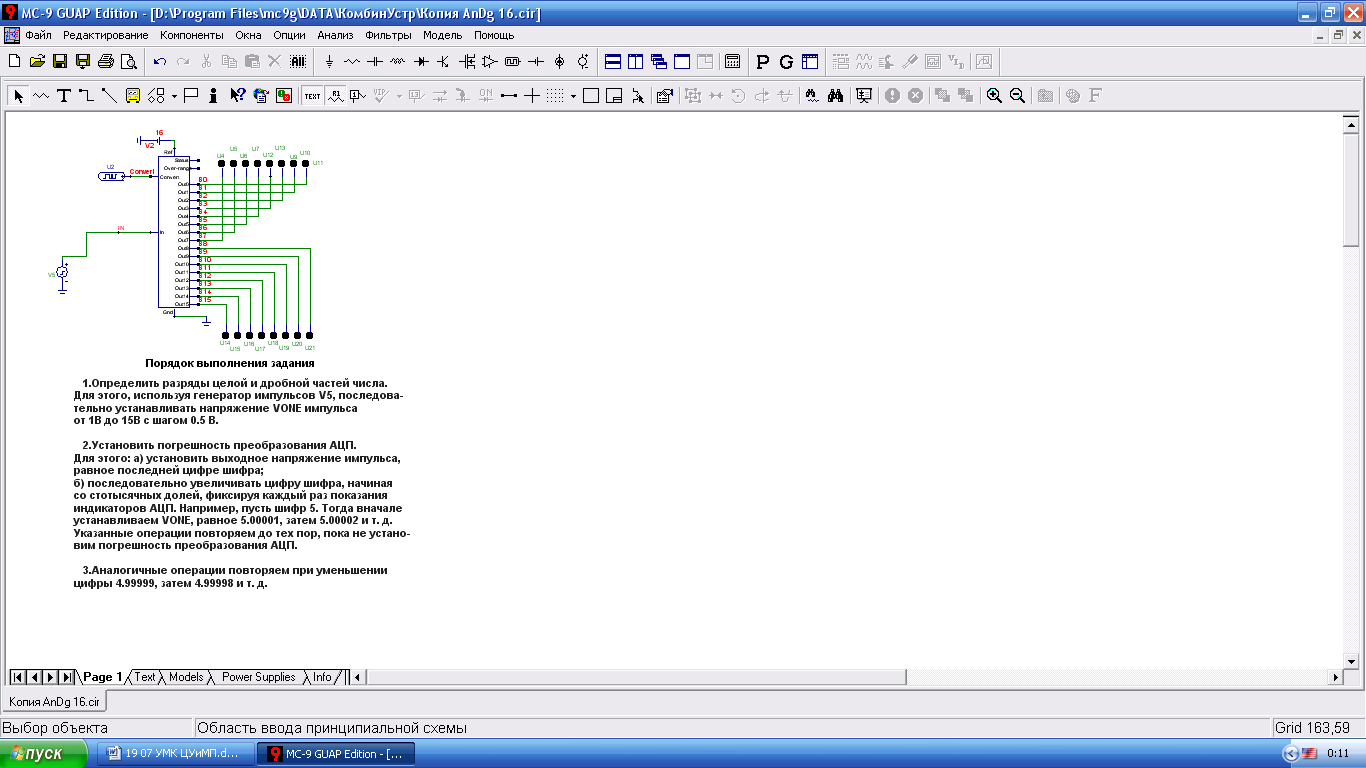


Рис. 4

**Порядок выполнения занятия**

* 1. Определить разряды целой и дробной частей числа.

Для этого используя генератор импульсов V5, последовательно устанавливать напряжение VONE импульс от 1 В до 15 В с шагом 0.5 В.

* 1. Установить погрешность преобразования АЦП.

Для этого: а) установить выходное напряжение импульсов, равное последней цифре шифра;

б) последовательно увеличивать цифру шифра, начиная с десяти-тысячных долей, фиксируя каждый раз показания индикаторов АЦП. Например, пусть последняя цифра шифра 5. Тогда вначале устанавливаем VONE, равное 5.0001, затем – 5,0002 и т. д. Указанные операции повторяем до тех пор, пока не будет установлена погрешность преобразования АЦП.

3. Аналогичные операции повторяем при уменьшении цифры 5 до значений 4.9999, затем 4.9998 и т. д.

**Содержание отчета**

1. Титульный лист с названием работы.

2. Основные теоретические положения.

3. Схема аналого-цифрового преобразователя.

4. Установленная погрешность преобразования АЦП.

5. Выводы.

**Занятие 17**

**по теме «Цифроаналоговые преобразователи (ЦАП)»**

**Теоретическая часть занятия**

Среди устройств ЦАП наиболее распространены устройства, преобразующие параллельный двоичный код в напряжение.

При этом преобразование сводится к суммированию элементарных напряжений. Элементарные напряжения *ui*образуются с помощью деления некоторого эталонного напряжения *Umax*резистивными делителями. Из операции суммирования исключаются те слагаемые *ui*, которые соответствуют нулевым значениям элементов *a****i*** , составляющих входное двоичное число: {*an-1 an-2 . . . a1 a0*}.

Принцип деления напряжения реализуется с помощью ключевых схем (рис. 1, *а* и *б*).

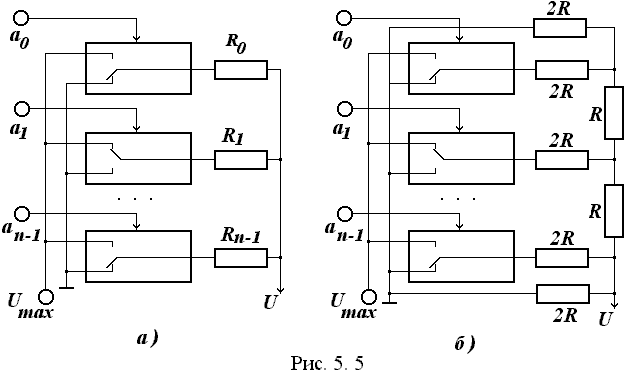


Рис. 1

Под ключом подразумевается достаточно быстродействующее электронное устройство. На рис. 1, *а* изображена схема с двоично- взвешенными резистивными цепями, т.е. схема с суммированием напряжений. На рис. 1, *б* изображен преобразователь «код - напряжение» с многозвенной резистивной цепью типа *R –* 2*R*(читается: «эр – два эр»). Эта схема чаще находит применение, когда производится не деление напряжений, а деление токов. В этой схеме резисторы имеют только два номинала, что делает ее удобной при интегральной технологии.

На рис. 2 приведена структурная схема ЦАП выпускаемой промыш-ленностью микросхемы К572ПА1.

Если на цифровом входе *i* ***–*** го разряда *ai*= 1, то усилитель откроет ключ, через который разрядный ток этого разряда резистивной матрицы поступит на *Выход* 1. В случае *ai*= 0 откроется другой ключ, через который ток поступит на выход2. При наличии сигналов на всех разрядах результирующий ток будет равен

 (1)

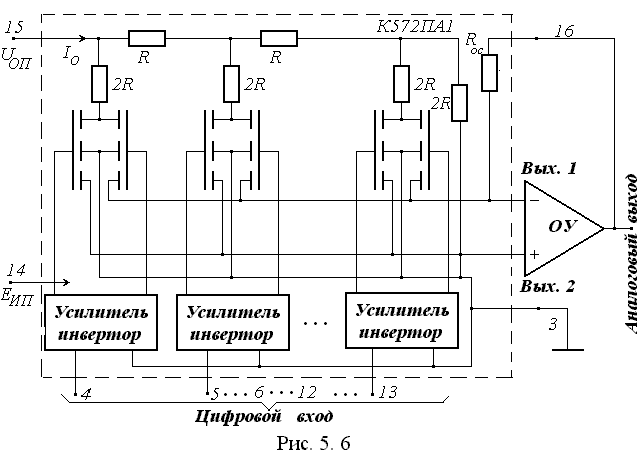


Рис. 2

Если необходимо иметь на выходе напряжение, что встречается чаще, то к микросхеме К572ПА1 подключается операционный усилитель. На аналоговом выходе ЦАП напряжение будет равно

 (2)

С учетом очевидных соотношений:

 (3)

выходное напряжение на аналоговом выходе, пропорциональное величине входного числа, можно представить в виде

 (4)

Следует также отметить, что использование цифровой формы представ-ления сигналов может обеспечить более высокую помехоустойчивость и стабильность параметров обработки, а также независимость от влияния изменений в окружающей среде.

**Задание**

1. Войти в меню File и выполнить команду Open либо щелкнуть соответ-ствующую пиктограмму.

2. В появившемся окне выбрать файл «АЦП 16 бит» и открыть его (рис. 3).

3. Выбрать режим Transient Analysis (Alt+1).

4. Провести моделирование на ПК в соответствии с порядком выполнения занятия.

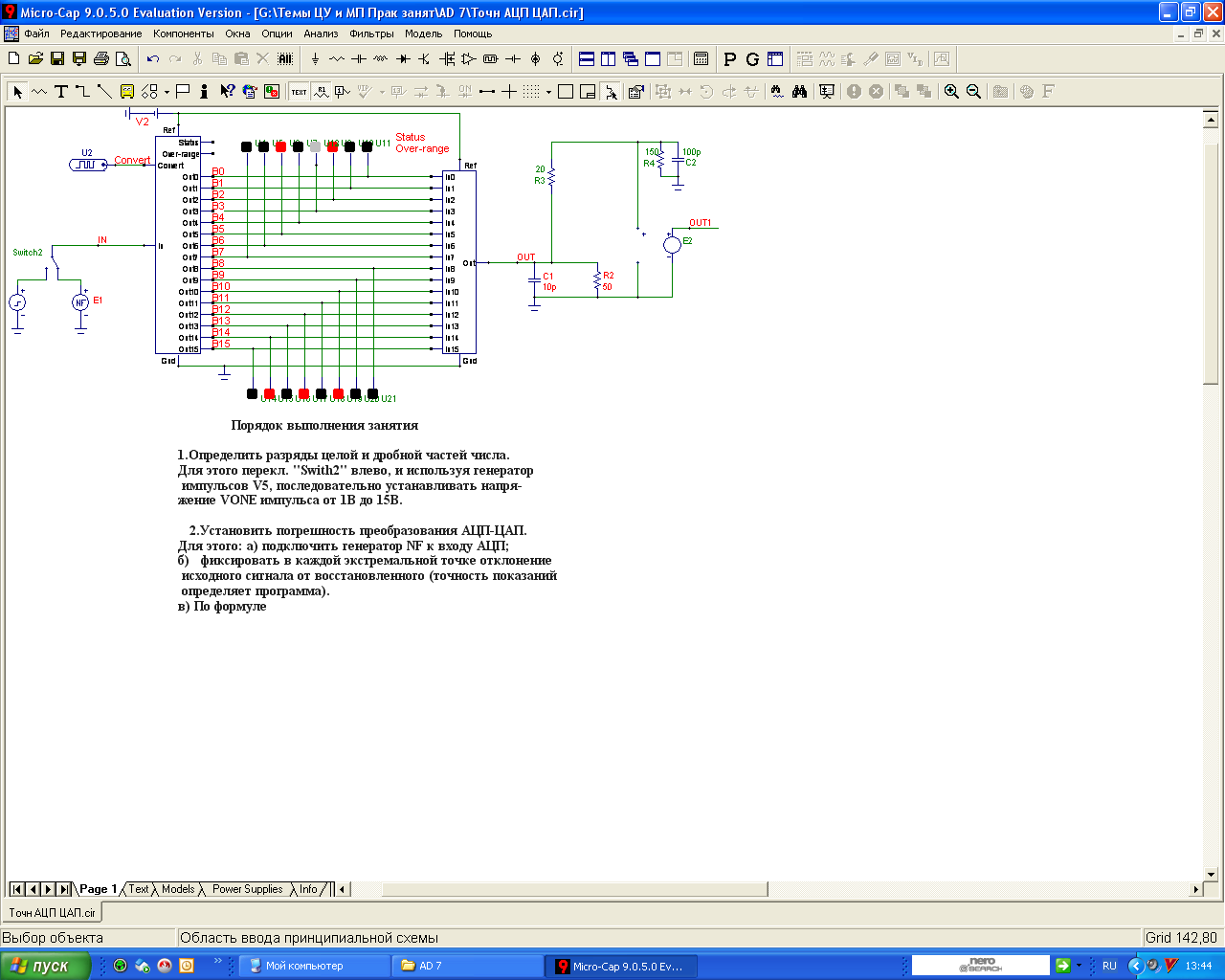


Рис. 3

**Содержание отчета**

1. Титульный лист с названием работы.

2. Основные теоретические положения.

3. Схема преобразователя АЦП-ЦАП.

4. Установленная погрешность преобразования

5. Выводы

**Занятие 18**

**по теме «Оперативные запоминающие устройства»**

Задание

1. Используя раздел 6 учебного пособия [1] и раздел 6.2 пособия [3], составить типичную структуру микросхемы оперативного запоминающего устройства (ОЗУ), пояснить назначение основных блоков и работу устройства.

2. Оценить емкость памяти ОЗУ, зависящую от числа строк и столбцов матрицы.

3. Привести временные диаграммы сигналов в режимах чтения и записи информации.

4. Составить схему наращивания разрядности ячеек памяти ОЗУ, пояснить работу схемы.

5. Составить схему наращивания числа и разрядности ячеек памяти ОЗУ, пояснить особенности работы схемы.

6. На микросхемах с организацией 1024 Х 1 создать блок памяти, имеющий организацию 4096 Х 8, т. е. блок памяти на 4096 8-разрядных ячеек.

а) Определить общее число необходимых микросхем;

б) Сформировать блок управления адресом, по которому будет производиться обращение к блоку памяти;

в) Составить упрощенную схему ОЗУ.

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Основные теоретические положения.

3. Расчет числа необходимых микросхем.

4. Схемы блока управления адресом и ОЗУ.

5. Выводы по занятию.

**Занятие 19**

**по теме «Архитектура микропроцессорной системы»**

**Задание**

1. Используя раздел 7 учебного пособия [1] и раздел 1.1 пособия [3], привести общую схему микропроцессорной системы на базе МПК КР 580, особенности её функционирования.

2. Привести структуру центрального процессора КР580ВМ80, особен-ности его функционирования.

3. Описать сигналы управления системы.

4. Описать организацию работы центрального процессора КР580ВМ80:

а) линии синхронизации;

б) машинные циклы;

в) временные диаграммы обобщенного машинного цикла работы МП.

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Основные теоретические положения.

3. Схемы структур микропроцессорной системы на базе МПК КР 580 и центрального процессора КР580ВМ80.

4. Временные диаграммы обобщенного машинного цикла работы МП.

5. Выводы по занятию.

**Занятие 20**

**по теме «Программная модель МП системы и способы адресации»**

**Задание**

1. Используя раздел 9 учебного пособия [1] и раздел 3 пособия [3], привести схему программной модели МП системы, пояснить её особенности.

2. Описать режимы адресации, привести схемы режимов адресации.

3. Отметить особенности прямой, непосредственной, регистровой и кос-венной адресации.

4. Привести примеры различных режимов адресации.

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Основные теоретические положения.

3. Режимы адресации, схемы режимов адресации.

4.Примеры режимов адресации: прямой, непосредственной, регистровой и косвенной адресации.

5. Выводы по занятию.

**Занятие 21**

**по теме «Команды пересылки данных**

**и арифметико-логические команды»**

**Основные теоретические сведения.**

Из арифметических операций микропроцессор (МП) выполняет только сложение и вычитание над байтами в двоичном и двоично-десятичном форматах.

Операнды представляются в формате с фиксированной запятой. В арифметических операциях адресуется только один операнд, второй операнд находится в аккумуляторе, а результат арифметической операции загружается в аккумулятор , и в соответствии с полученным результатом устанавливаются признаки S,Z,C,P в регистре признаков. При этом надо помнить, что при выполнении арифметических операций данные находятся в аккумуляторе, “разрушаются”, за исключением операций сравнения, при которых оба операнда не изменяются, а лишь происходит установка битов регистра признаков.

Арифметические операции выполняются с помощью команд сложения ADD и вычитания SUB над однобайтными числами. При сложении (вычитании) чисел формата более одного байта вначале складываются (вычитаются) их младшие байты с помощью команд сложения с переносом, вычитанием с заёмом ADD (SBB) . Затем та же операция проводится над старшими байтами, в которых автоматически учитывается перенос (заём) из младшего байта , если он имел место.

Также в системе команд МП имеются команды сложения двухбайтных слов DAD rp (HL+rp - HL), при этом суммируется содержимое регистра HL с содержимым регистровой пары rp (BC,DE,HL,SP), а результат заносится в регистр HL и устанавливается бит переноса.

На ПК установлена инструментальная программа, предназначенная в занятии для изучения и исследования программ, написанных для микро-процессорной системы на базе МП КР580.

Изучаемая программа набирается на языке ассемблера и после устранения синтаксических ошибок компилируется в машинный код микро-процессора КР580.

В процессе набора программ программа обеспечивает выполнение сле-дующих функций:

- ввод команд и данных с клавиатуры;

- отображение адресов ячеек памяти и их содержимого;

- отображение содержимого аккумулятора и других внутренних программно-доступных регистров (A, FL, B, C, D, E, H, L, SPH, SPL, PCH, PCL);

- отображение содержимого программного счетчика;

- запуск программы в режиме выполнения по машинным циклам и по тактам команды.

**Задание**

Произвести суммирование двух чисел.

Для суммирования выбираются две последние цифры шифра и их пе-рестановка. Например, число 47 и его перестановка 74.

При этом по последней цифре шифра команда суммирования двух чисел выбирается следующим образом.

Если цифра нечетная, то выбирается команда «ADD C», которой соответ-ствует машинный код «81»(см. «Система команд»).

Если цифра четная, то выбирается команда «ADD D», которой соответствует машинный код «82».

**Порядок выполнения суммирования**

1) На лицевой панели макета микропроцессора нажать кнопку «Система команд» и ознакомиться с её командами.

2) Ознакомиться с примером суммирования чисел 47 и 74***.***

А) Ввести в любую ячейку памяти, например в «0005» машинный код «80» команды суммирования «ADD B»(см. «Система команд»). Для этого:

а) «1Л»→ «ОЗУ»; б) клав.→ 0005 80; в) Ввод; г) проверить по индикатору, что: «Адрес Код Команда»↔«0005 80 ADD B».

Б) Ввести числа для суммирования. Для этого: а) «1Л»→ «Регистр»; б) «1Л»→ «А»; в) «1Л»→ 4; г) «1Л»→ 7; д) Ввод; е) на индикаторе в регистре А число 47; ж) «1Л»→ 7; з) «1Л»→ 4; и) Ввод; к) в регистре В число 74.

В) Установить программный счетчик РС. Для этого: а) «1Л»→ «ОЗУ»; б) «1Л»→ «PcL»; в) клав.→05; г) «1Л»→ «PcH»; д) клав.→ 00; е) на индикаторе – РС 0005.

Г) Проделать операции: а) «1Л»→ «ОЗУ»; б) клав.→ 0005 80; в) Ввод.

Д) Потактовый режим ○→ ∙

Е) «1Л»→ «Структурная схема» → «Выполнение». Зафиксировать циклограмму при выполнении команды.

**Задание**

Произвести вычитание двух чисел.

Для вычитания выбираются два числа: из числа 99 вычитаются две последние цифры шифра.

При этом по последней цифре шифра команда вычитания двух чисел выбирается следующим образом.

Если цифра нечетная, то выбирается команда «SUB C», которой соответствует машинный код «91»(см. «Система команд»).

Если цифра четная, то выбирается команда «SUB D», которой соответствует машинный код «92».

**Порядок выполнения вычитания**

Ознакомиться с примером с примером вычитания числа 57 из числа 99.

А) Ввести в любую ячейку памяти, например в «0010» машинный код «91» команды вычитания «SUB C»(см. «Система команд»). Для этого:

а) «1Л»→ «ОЗУ»; б) клав.→ 0010 91; в) Ввод; г) проверить по индикатору, что: «Адрес Код Команда»↔«0010 91 SUB B».

Б) Ввести числа для вычитания. Для этого: а) «1Л»→ «Регистр»; б) «1Л»→ «А»; в) «1Л»→ 9; г) «1Л»→ 9; д) Ввод; е) на индикаторе в регистре А число 99; ж) «1Л»→ 5; з) «1Л»→ 7; и) Ввод; к) в регистре В число 57.

В) Установить программный счетчик РС. Для этого: а) «1Л»→ «ОЗУ»; б) «1Л»→ «PcL»; в) клав.→10; г) «1Л»→ «PcH»; д) клав.→ 00; е) на индикаторе – РС 0010.

Г) Проделать операции: а) «1Л»→ «ОЗУ»; б) клав.→ 0010 90; в) Ввод.

Д) Потактовый режим ○→ ∙

Е) «1Л»→ «Структурная схема» → «Выполнение». Зафиксировать циклы и такты, а также состояние блоков схемы при выполнении команды.

Зафиксировать циклограмму при выполнении команды.

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Основные теоретические положения.

3. Программы суммирования и вычитания чисел шифра.

4. Циклограммы процессов и зафиксированные состояния блоков.

5. Выводы по занятию.

**Занятие 22**

**по теме «Команды пересылки данных**

**и арифметико-логические команды»**

**Основные теоретические сведения**

Умножение чисел в МП реализуется с помощью команд суммиро­вания и сдвига. Для получения  к сумме частичных произведений  добавля­ется , если очередной разряд 1, после чего осуществляется сдвиг ; указан­ная процедура повторяется для каждого разряда . При этом анализ  можно начинать со старших разрядов и соответственно сдвигать влево, либо наоборот.

Знак обычно формируют независимо от описанной процедуры умно-же­ния на основе анализа знаков сомножителей, а при осуществлении умно-жения используют прямые коды сомножителей без знака. В этом случае необходимо учитывать, что освобождающиеся разряды при сдвигах  должны заполняться нулями.

Деление двоичных чисел основывается на последовательном вычитании делителя из разрядов делимого и остатка от деления. Таким образом, деление реализуется с помощью операций сдвига, вычитания и анализа результата вы­читания. По аналогии с умножением здесь целесообразно осуществлять деле­ние чисел без знаков, а знак частного определять отдельно на основе анализа знаков операндов. Перед выполнением деления необходимо убедиться, что де­литель не равен нулю.

Рассмотренные варианты выполнения арифметических действий с помо­щью МП подразумевают представление чисел в форме с фиксированной запя­той (точкой). Обработку данных с плавающей запятой выполняют путем раз­дельной обработки порядков и мантисс чисел на основе рассмотренных алго­ритмов выполнения операций над числами, представленными в форме с фикси­рованной запятой. Следует отметить, что программы реализации арифметиче­ских операций над числами, представленными в форме с плавающей запятой, достаточно сложны и их выполнение занимает большое время, в результате чего такие программы редко применяются в МПК КР580.

Умножение двух двоичных чисел без знака может быть выполнено несколькими способами, например путём повторного сложения либо при-менения операции регистрового сдвига.

Повторное сложение обеспечивает простейшую, но самую медленную форму умножения.

Используя операцию сдвига, можно получить более быстрое умно-жение.

Рассмотрим пример умножения двух четырехбитных двоичных чисел с применением операции регистрового сдвига:

1101 множимое (23+22+0+20=13)

Х 13 х 11 =143

1011 множитель (23+0+21+20=11)

\_\_\_\_

1101 1-е частичное произведение

1101 2-е частичное произведение

0000 3-е частичное произведение

1101 4-е частичное произведение

----------

10001111 произведение (27+0+0+0+23+22+21+20=128+8+4+2+1=143).

Как видно из примера, в процессе выполнения операции умножения формируются частичные произведения, которые суммируются с соответ-ствующими сдвигами друг относительно друга.

При умножении множимого на соответствующий разряд множителя будут получены либо множимое, либо нуль.

В процессоре суммирование частичных произведений происходит по-следовательно: формируется одно из частичных произведений, к нему с соответствующим сдвигом прибавляется следующее частичное произведение и т. д., пока не просуммируются все частичные произведения.

Этот процесс суммирования можно начинать с младшего (как это пока-зано) либо со старшего частичного произведения.

Число разрядов для фиксации произведения равно сумме числа разрядов множимого и множителя.

Рассмотрим операцию умножения дробных чисел 0.1101=2-1+2-2+0+

+2-4=13/16 и 0.1011=2-1+0+2-3+2=4=11/16 с применением операции сдвига:

0.1101| 1-е частичное произведение

0.0110|1 сдвиг на один разряд вправо

+ |

0.1101| 2-е частичное произведение

------------

1.0011|1 сумма 1-го и 2-го частичных произведений

0.1001|11 сдвиг на один разряд вправо

+ |

0.0000| 3-е частичное произведение

----------|----

0.1001|11 cумма 1-го, 2-го и 3-го частичных произведений

0.0100|111 сдвиг на один разряд вправо

+ |

0.1101| 4-е частичное произведение

-----------|-----

1.0001|111 сумма частичных произведений

0.1000|1111 сдвиг вправо, произведение.

Число 0.10001111=2-1+0+0+0+2-5+2-6+2-7+2-8=143/256.

(13/16)\*(11/16)=143/256.

Если число разрядов произведения равно числу разрядов пере-множаемых чисел, то в таком приближенном представлении результата не фиксируются цифры разрядов, выдвигаемых за вертикальную линию, т. е. результат умножения равен 0.1000=2-1=1/2=128/256. Что означает потерю четырех младших разрядов.

Можно произвести округление по правилу: если старший из отбрасываемых разрядов содержит единицу, то к младшему из сохраняемых разрядов прибавляется единица. Тогда результат с округлением в примере:

0.1001=2-1+0+0+2-4=9/16=144/256.

Рассмотрим алгебраическое деление чисел, представленных в форме с фиксированной запятой. Знак частного может быть найден тем же приемом, что и знак произведения. Поэтому рассмотрим пример нахождения модуля част-ного положительных чисел *a*=0.10010 и *b*=0.10110.

При этом операцию вычитания числа заменим прибавлением числа – *b*, представленного в дополнительном коде: (-*b*)доп=1.01010.

Число *a*=0.10010=2-1+0+0+2-4=9/16;

Число *b*=0.10110=2-1+0+2-3+2-4=11/16.

Тогда частное *a/b*=9/11=0.81818…

Частное, полученное при делении двоичных чисел:

0.11010010…=2-1+2-2+0+2-4+0+0+2-7+0…=0.82031…

Округление до сотых долей дает одинаковый результат 0.82.

*a*  0.10010 0.10110

+ 0. 1 1 0 1 0 0 1 0

(-*b*)доп 1.01010 *с* <0

*с*  1.11100

сдвиг влево 1.11000

+

*b* 0.10110  *c* >0

*c*  0.01110

сдвиг влево 0.11100

+

(-*b*)доп 1.01010  *с* >0

*с*  0.00110

сдвиг влево 0.01100

+

(-*b*)доп 1.01010 *с* <0

*с*  1.10110

сдвиг влево 1.01100

+

*b*  0.10110 *c* >0

*c*  0.00010

сдвиг влево 0.00100

+

(-*b*)доп 1.01010 *с* <0

*с*  1.01110

и т.д.

Рассмотрим программу умножения (табл. 1), в которой умножение начинается со старших разрядов множителя, причём множимое и сумма частичных произведений в каждом такте умножения сдвигаются влево, а множимое остается неподвижным. Формат умножения 16 х 8, т.е. множимоё представлено двумя байтами, а множитель – одним байтом.

Входные параметры программы:

(С) – множитель;

(D,E) – множимое;

Выходные параметры:

(A, H, L) – произведение.

В процессе выполнения программы используются все регистры, сохра-няются (C) и (D, E).

Частичные произведения формируются, начиная со старшего частичного произведения. Накопление суммы частичных произведений происходит в паре регистров Н, L, т. е. к содержимому предварительно сброшенной в нуль пары H, L, вначале прибавляется восьмое частичное произведение, Затем после сдвига на один разряд влево к содержимому пары регистров H, L прибав-ляется седьмое частичное произведение и т.д.

Таким образом, этот процесс носит циклический характер: цикл, содержащий операции сдвига влево содержимого пары регистров H, L, форми-рования и прибавления к содержимому пары регистров H, L очередного частичного произведения, повторяется восемь раз.

Для счета числа повторений цикла организуется счетчик на регистре В. В этот регистр предварительно заносится число 8 и после каждого повторения вычитается единица из содержимого регистра В с последующей проверкой равно ли нулю его содержимое. При достижении нулевого значения произво-дится выход из цикла.

*Таблица* *1*.

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Машинный  код | Мнемокод | Комментарий |
| 0800 | AF | XRA A | Поразрядное исключающее ИЛИ над содержимым регистра и аккумулятора |
| 0801 | 67 | MOV H, A | Передача из регистра А в регистр H |
| 0802 | 6F | MOV L, A | Передача из регистра А в регистр L |
| 0803 | 81 | ADD C | Сложение содержимого реги-стра С и аккумулятора |
| 0804 | C8 | RZ | Возврат при нуле, если мно-житель =0 |
| 0805 | AF | XRA A | Поразрядное исключающее ИЛИ над содержимым реги-стра и аккумулятора |
| 0806 | B2 | ORA D | Поразрядное логическое сложение содержимого реги-стра D и аккумулятора |
| 0807 | B3 | ORA E | Поразрядное логическое сло-жение содержимого регистра Е и аккумулятора |
| 0808 | C8 | RZ | Возврат при нуле, если мно-жимое =0 |
| 0809 | 79 | MOV A, C | Передача из регистра C в регистр A |
| 080A | 06 | MVI B, 8 | Передача байта в регистр В – счетчик циклов |
| 080B | 08 |  |  |
| 080C | 29 | DAD H | Сдвиг текущей суммы части-чного произведения и множи-теля вправо |
| 080D | 17 | RAL | Циклический сдвиг содержи-мого аккумулятора влево че-рез перенос |
| 080E | D2 | JNC | Переход при отсутствии пе-реноса ( переход к адресу 0814) |
| 080F | 14 |  |  |
| 0810 | 08 |  |  |
| 0811 | 19 | DAD D | Сложение множимого с теку-щей суммой частичного произведения |
| 0812 | CE | ACI | Сложение байта с содер-жимым аккумулятора с учетом переноса |
| 0813 | 00 |  |  |
| 0814 | 05 | DCR B | Уменьшение содержимого регистра В на единицу |
| 0815 | C2 | JNC | Переход при отсутствии пе-реноса ( переход к адресу 080С) |
| 0816 | 0C |  |  |
| 0817 | 08 |  |  |
| 0818 | 76 | HLT | Останов |

**Задание**

Перемножить два числа.

Для умножения выбираются две последние цифры шифра и их пере-становка. Например, число 47 и его перестановка 74, которые вводятся соответственно в регистр С и пару регистров D, E.

***Примечание. Десятичные числа 47 и 74, введенные в указанные адреса, становятся шестнадцатеричными, т. е. число 47Н=4\*16+7=71; число 74Н=7\*16+4=116.***

***Поэтому их произведение будет равно71\*116=8236=202СН.***

**Порядок выполнения занятия**

1) На лицевой панели макета микропроцессора нажать кнопку «Система команд» и ознакомиться с командами микропроцессора.

2) Согласно табл. 1 в эмуляторе по указанным адресам набираются машинные коды и по третьему столбцу мнемокодов контролируется правиль-ность их набора.

3) После набора программы умножения табл.1 устанавливается счетчик команд: вначале младшие разряды PcL→ 00, затем старшие разряды PcH→ 08.

4) ОЗУ→0800 AF.

5) Покомандный режим: выполнение в режиме структурной схемы и фиксация состояния регистров в форме, которую разработать самостоятельно.

**Содержание отчета**

1. Титульный лист с названием занятия.

2.Основные теоретические положения.

3. Программы умножения чисел шифра.

4. Циклограммы процессов и зафиксированные состояния блоков.

5. Выводы по занятию.

**Занятие 23**

**по теме «Команды передачи управления, ввода-вывода и др.»**

**Теоретическая часть занятия**

Структурная схема процессорного модуля приведена на рис. 1.

УУ

ДШ

РК

УО

АЛУ

A

F

БР

РС

SP

H

L

B

C

D

E

W

Z

БА

ША

БД

ШД

УС

SYNC

RESET

Ф1

Ф2 WR1

DBIN

ЛС

Рг С

STSTB

ШУ

Рис. 1

В состав кристалла процессора ПР входят устройство синхронизации (УС), устройство управления (УУ), устройство обработки, блок регистров (БР) и буферы данных (БД) и адреса (БА).

В ходе выполнения программы процессор устанавливает адрес памяти на 16-разрядный ША По 8-разрядной ШД из памяти принимаются коды команд и данные, по этой же шине данные записываются в память. Коды команд поступают по внутренней магистрали в регистр команд РК и дешифрируются с помощью дешифратора ДШ. Для каждой команды устройство управления генерирует необходимую последовательность сигналов, управляющих обра-боткой и пересылкой информации в микропроцессоре.

Устройство обработки (УО) с помощью АЛУ выполняет задаваемые кодом команды арифметические и логические операции. При этом один из операндов находится в аккумуляторе, а второй подаётся через внутреннюю магистраль из блока регистров БР или с шины данных ШД через буфер БФ. Результат операции фиксируется в аккумуляторе А, а признаки результата – в регистре F. Отдельные биты регистра F расшифровываются следующим образом:

S – признак знака результата (при S=1результат отрицательный);

Z – признак нуля (при Z=1результат нулевой);

С – признак переноса (при С=1 имеется перенос из 7-го бита результата);

АС – признак вспомогательного переноса (из 3-го бита результата в 4-й);

Р – признак четного паритета (при Р=1 число единиц в байте результата четное).

16-разрядный счетчик команд РС находится в блоке регистров БР, и после выборки каждого байта команды его значение увеличивается на 1.

16-разрядный указатель стека SР используется для адресации находящейся в ОЗУ стековой области памяти. При записи информации в стек содержимое SP уменьшается на число переданных байтов, а при выборке информации – увеличивается.

8 – разрядные регистры H, L, D, B, E, C, W, Z используются для хранения промежуточных результатов. Заметим, что регистры W и Z так же как и регистр команд РК, программно не доступны. Выдачей информации из регистров БР на шину адреса и обменом с шиной данных управляет УУ.

В состав БР входит также не показанная на рис. 1 схема инкремента-декремента, позволяющая увеличивать или уменьшать на 1 содержимое регис-тров.

Устройство синхронизации УС осуществляет тактирование работы мик-ропроцессорного модуля и совместно с УУ – обработку и выдачу управля-ющих сигналов.

Функции управляющих сигналов рассмотрим на примере выполнения команды вывода OUT 2DH, которая выдаёт байт информации из аккумулятора А в буферный регистр внешнего устройства с адресом 2D.

Все числа далее указываются в 16-ричном коде, при записи на языке ассемблер этот код заканчивается буквой Н. Команда OUT 2DН занимает в памяти 2 байта. В первом байте записывается код команды D3, во втором – адрес внешнего устройства 2D. Будем считать, что первый байт расположен в ячейке с адресом 0800, а содержимое аккумулятора перед выводом равно 15.

Количество машинных циклов, в течение которых выполняется каждая команда, равно числу необходимых обращений к памяти и внешним устройс-твам.

Команда OUT выполняется в течение 3 циклов (рис. 2) – выборки кода операции М1, чтения адреса ВУ – М2 и вывода содержимого аккумулятора М3.

Каждый машинный цикл состоит из 3-5 тактов. Тактирование осуществляется подачей внешних импульсов частотой 2 МГц на входы Ф1 и Ф2 с генератора тактовых импульсов.

На рис. 2 приведена циклограмма выполнения команды.

В первом такте Т1 каждого машинного цикла на шине адреса ША выставляется адрес ячейки, в которой находится код команды, а на шину данных ШД выдаётся байт состояния, который фиксируется в регистре состояния РС с помощью сигнала SYNC. Байт состояния определяет тип выполняемого цикла, и его отдельные биты имеют следующий смысл:

D0=1 цикл прерывания;

D1=0 выдача данных (в память или внешнее устройство);

D2=1 адрес стека (на ША выдано содержимое SP);

D3=1 цикл останова (после выполнения команды HLT);

D4=1 вывод данных на внешние устройства;

D5=1 цикл выборки кода операции;

D6=1 ввод данных с внешнего устройства;

D7=1 чтение данных из памяти.

M1 M2 M3

T1 T2 T3 T4 T1 T2 T3 T1 T2 T3

Ф1

Ф2

ША

0800

0801

2D2D

SYNC

ШД

15

10

2D

82

D3

A2

MEMR1

WR1

I/O W1

Рис.2

Во втором такте Т2 проверяются сигналы готовности READY вну-треннего останова HLTA, захвата шин HOLD, а в последнем машинном цикле сигнал запроса прерывания INT. В конце второго такта процессор выдаёт сигнал приёма данных DBIN или выдачи информации. Они определяют время и направление передачи по шине данных. В цикле выборки кода операции выдаётся сигнал DBIN, который используется для формирования системного сигнала приёма MEMR1 на шине управления.

Если сигналы готовности READY, HOLD и HLTA не активны, то процессор переходит к такту Т3, в котором завершаются приём и выдача ин-формации по шине данных.

Такт Т4 в цикле выборки используется для внутренних операций процессора, в частности для дешифрования кода команды и межрегистровых передач. В некоторых командах такие передачи требуют пятого такта работы процессора.

Таким образом, в первом цикле выборки кода операции процессор выдаёт на ША из РС адрес 0800 и принимает в регистр команд РК код операции D3.

Во втором цикле М2 чтение из РС выдаётся адрес памяти 0801 и принимается адрес внешнего устройства 2D(он записывается в регистры W и Z). В третьем цикле вывода М3 на ША выдаётся адрес 2D2D из регистров W и Z и содержимое аккумулятора 15 выводится в буферный регистр внешнего устройства. Так как адреса внешних устройств, в отличие от адресов памяти, 8-разрядные, то при их выдаче по 16-разрядной ША младший и старший байты дублируются для недопущения несимметрии нагрузки.

**Задание**

Выполнить команду OUT, машинный код которой «2D»(см. «Система команд»).

**Порядок выполнения занятия**

1) На лицевой панели макета нажать кнопку «Система команд» и ознакомиться с командами микропроцессора.

2) Для ввода команды OUT проделать следующие операции.

А) «1Л» осуществить→ «Сброс ОЗУ» и «Сброс Рег».

Ввести в регистр А число«15». Для этого: а) «1Л» → «ОЗУ» ; б) «1Л»→ «Регистр» ;в) клавиатура «1Л»→ «А»; г) клавиатура «1Л»→1; «1Л»→5; д) «1Л»→ «Ввод»; е) убедиться, что число 15 введено в регистр А на индикаторе «Регистр».

Б) Ввести в память команду «OUT». Для этого:

а) «1Л»→ «ОЗУ»; б) «1Л»→ «Регистр»; в) клавиатура «1Л»→0800 D3;

г) «1Л»→Ввод; д) убедиться по индикатору, что по адресу – 0800, код – D3,

команда – OUT N.

В) Ввести в память адрес внешнего устройства «D2». Для этого:

а) по следующему адресу – 0801 «1Л»→D2; б) Ввод; в) на индикаторе – 0801 D2.

Г) Установить программный счетчик РС. Для этого: а) «1Л»→ «ОЗУ»; б) «1Л»→«PcL»; в) клавиатура→«00»; г) «Ввод»; д) «1Л»→«PcH»; е) клавиатура→«08»; ж) «Ввод»; з) убедиться, что в программном счетчике установлено – «РС: 0800».

Д) Проделать операции: а) «1Л»→ «ОЗУ»; б) клав.→ 0004 21; в) Ввод.

Е) Потактовый режим ○→ ∙

Ж) «1Л»→«Структурная схема»→«Выполнение». Зафиксировать цикло-грамму при выполнении команды, и сравнить её с рис. 2.

**Задание**

В соответствии с последней цифрой шифра выбрать команду «LXI r p,data16» - «загрузить непосредственный 16-разрядный операнд».

Если последняя цифра шифра четная, то выбирается команда «LXI B,d16», которой соответствует машинный код «01».

Если последняя цифра шифра нечетная, то выбирается команда «LXI D,d16», которой соответствует машинный код «11».

Последующие действия аналогичны рассмотренному ниже примеру выбора команды «LXI H,d16», которой соответствует машинный код «21», но с учетом необходимых поправок вводимых цифр и имен регистров.

**Порядок выполнения занятия**

Ознакомиться с примером ввода команды «LXI H,d16»:

А) «1Л» осуществить→ «Сброс ОЗУ» и «Сброс Рег».

Ввести в регистр Н последние две цифры шифра, например 47. Для этого: а) «1Л» →«ОЗУ»; б) «1Л»→«Регистр»; в) клавиатура «1Л»→«Н»; г) клавиатура «1Л»→4; «1Л»→7; д) «1Л»→«Ввод»; е) убедиться, что число 47 введено в регистр Н на индикаторе «Регистр».

Ввести в регистр L число 74 (перестановка числа 47), повторив анало-гично пп а)…е).

Б) Ввести команду «LXI H,d16»: а) «1Л»→«ОЗУ»; б) выбрать любую ячейку памяти, например с номером «0004», «1Л»→0004 21; в) «Ввод»; г) убедиться по индикатору, что по адресу – 0004, код – 21, команда – LXI H,d16.

В) Ввести второй байт команды в ячейку памяти с номером «0005»,например число 11; третий байт команды ввести в ячейку памяти «0006», например число 33. По индикатору убедиться о введении указанных чисел.

Г) Установить программный счетчик РС. Для этого: а) «1Л»→«ОЗУ»; б) «1Л»→«PcL»; в) клавиатура→«04»; г) «Ввод»; д) «1Л»→«PcH»; е) клавиатура→«00»; ж) «Ввод»; з) убедиться, что в программном счетчике уста-новлено – «РС: 0004».

Д) Проделать операции: а) «1Л»→ «ОЗУ»; б) клав.→ 0004 21; в) Ввод.

Е) Потактовый режим ○→ ∙

Ж) «1Л»→ «Структурная схема» → «Выполнение».

Согласно изучаемой команде «LXI H,d16» вначале в младший регистр L пары НL вместо числа «47» будет записан второй байт команды – число «11». Затем в старший регистр Н пары НL вместо числа «74» будет записано число «33», т. е. третий байт команды.

Зафиксировать циклограмму при выполнении команды.

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Основные теоретические положения.

3. Программы команды вывода «OUT», команды «LXI r p,data16» - «загрузить непосредственный 16-разрядный операнд».

4. Циклограммы процессов и зафиксированные состояния блоков.

5. Выводы по занятию.

**Занятие 24**

**по теме «Программирование в МП системе»**

**Теоретическая часть занятия**

В занятии изучается программирование на машинном языке и наязыке ассемблер реализации алгоритма работы устройства, заданного уравнением: V(t)=kx2(t)+y(t-N), где k — заданная константа; x(t) — сигнал на одном входе (порт X — 1 байт); y(t-N) — сигнал, поступающий по второму входу и отстающий от x(t) на N=10 тактов (порт Y — 1 байт); V(t) — выходной сигнал, выводится через порт V (2 байта) [5]. Для хранения коэффициента k можно отвести ячейку ОЗУ или формировать этот коэффициент в самой программе.

Для хранения N слов, считанных в моменты времени, следующие за ti после поступления y(ti -N), целесообразно организовать очередь. Внутри ОЗУ для этого выделяется некоторая область, выполняющая функции стека.

Если в проектируемом устройстве содержится не более восьми портов для подключения внешних устройств ввода-вывода, то каждому порту выделяется индивидуальная адресная линия, что соответствует использованию кода «1 из 8-ми» для кодирования портов ввода/вывода:

PORT 1 — 00000001 (01H) PORT 5 — 00010000 (10H=16)

PORT 2 — 00000010 (02H) PORT 6 — 00100000 (20H=32)

PORT 3 — 00000100 (04H) PORT 7 — 01000000 (40H=64)

PORT 4 — 00001000 (08H) PORT 8 — 10000000 (80H=128)

***Распределение памяти***

Весь объем адресуемой памяти (64К) с адресами 0000H…FFFFH следует распределить следующим образом. Для основной программы используются адреса, начиная с 0000H до FFFFH. Для инициализации портов ввода-вывода задаются их адреса: порт ввода переменной x(t) — 01H; порт ввода y(t-N) — 02H; порт вывода V(t) — 04H (применяется код «1 из 8-ми»).

***Структурная схема алгоритма***

В соответствии с заданным алгоритмом и словесным описанием составляется блок-схема алгоритма, представленная на рис. 1.

В приведенном алгоритме решения задачи константа k формируется программным путем с помощью сдвигов влево, вправо и суммирования ре-зультатов. Константа N, а также остальные константы записываются в 16-ричном коде.

Программы на ассемблере и в машинных кодах, приведенные в [5], пред-ставлены в табл. 1.

7

Начало

7

1

Вычисление kx2(t)

Инициализа-ция портов в/вывода вывода

Организация стека

Организация и обнуление очереди

Ввод

y(t-N)

Ввод x(t)

Вычисление kx(t)

7

2

3

4

5

6

y(t-N) в дополнит. коде

y(t-N)0

НЕТ

Вычисление V(t)

ДА

Вывод V(t)

Конец

8

9

10

11

Рис. 1

При программировании на машинном языке все элементы программы (коды операций, адреса, данные) представляются в 16-ричной форме, но при этом внутри МП системы информация циркулирует в двоичной форме.

*Таблица 1*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ад-рес | Машин-ный код | Метка | Мне-мони-ка | Операнд | Комментарий |
| Блок начальных директив | | | | | |
|  |  |  | ORG | 0000H | ;Директива начального адре-;са |
|  |  | PORTX | EQU | 01H | ;Инициализация портов вво-;да переменных x(t), y(t), ;y(t-N) и вывода V(t) |
|  |  | PORTY | EQU | 02H |
|  |  | PORTV | EQU | 04H |
| Организация стека и очереди | | | | | |
| 0000 | 31 00 84 | M0: | LXI | SP,8400H | ;Организация стека в ОЗУ |
| 0003 | 21 0A 80 |  | LXI | H,800AH | ;Загрузка адреса указателя ;начала очереди в ОЗУ ;(N=0АН) |
| 0006 | AF | M1: | XRA | A | ;Обнуление аккумулятора |
| 0007 | 77 |  | MOV | M,A | ;Обнуление ячейки очереди ;(адр. HL) |
| 0008 | 3E 01 |  | MVI | A,01H | ;Номер ячейки конца оче-;реди: 8001Н; счет ячеек в ;регистре L |
| 000A | 2D |  | DCR | L | ;Адрес следующей ячейки |
| 000B | BD |  | CMP | L | ;(A)-(L)=0? |
| 000C | C2 06 00 |  | JNZ | M1 | ;Переход на метку М1, если ;это не последняя ячейка ;очереди |
| 000F | 21 0A 80 |  | LXI | H,800AH | ;Повторная загрузка указа-;теля начала очереди |
| Ввод переменной *y(t-N)* | | | | | |
| 0012 | AF | LINE: | XRA | A | ;Подготовка к вводу пере-;менной |
| 0013 | DB 02 |  | IN | PORTY | ;Ввод y(t-N); параметр в ак-;кумуляторе |
| 0015 | 77 |  | MOV | M,A | ;Данные — в начале очереди |
| 0016 | F5 |  | PUSH | PSW | ;y(t-N) — в стеке для хра-;нения |
| 0017 | 00 |  | NOP |  | ;Пустые операции для уве-;личения длительности так-;та ввода |
| 0018 | 00 |  | NOP |  |
| 0019 | 00 |  | NOP |  |
| 001A | 00 |  | NOP |  |
| Ад-рес | Машин-ный код | Метка | Мне-мони-ка | Операнд | Комментарий |
| 001B | AF |  | XRA | A | ;Подготовка к вводу в акку-;мулятор содержимого по-;следней ячейки очереди |
| 001C | 3E 01 |  | MVI | A,01H | ;Указатель последней ячей-;ки очереди — в рег. А |
| 001E | 2D |  | DCR | L | ;Переход на следующую ;ячейку |
| 001F | BD |  | CMP | L | ;Последняя ли ячейка:  ;(A)-(L)=0? |
| 0020 | C2 12 00 |  | JNZ | LINE | ;Если результат не нулевой, ;то переход на LINE |
| 0023 | AF |  | XRA | A | ;Подготовка к вводу пере-;менной x(t) |
| Ввод и обработка *x(t)* | | | | | |
| 0024 | DB 01 |  | IN | PORTX | ;x(t) — в аккумуляторе |
| 0026 | 00 |  | NOP |  | ;Пустые операции для уве-;личения длительности так-;та ввода |
| 0027 | 00 |  | NOP |  |
| 0028 | 00 |  | NOP |  |
| 0029 | 00 |  | NOP |  |
| 002A | 21 00 00 |  | LXI | H,0000H | ;Обнуление регистровой па-;ры H, L |
| 002D | 16 00 |  | MVI | D,00H | ;Обнуление регистров обще-;го назначения D и E |
| 002F | 5A |  | MOV | E,D |
| 0030 | 5F |  | MOV | E,A | ;Копирование x(t) в рег. Е |
| 0031 | 07 |  | RLC |  | ;2x(t) — в аккумуляторе |
| 0032 | A7 |  | ANA | A | ;Сброс флага переноса: ;CY=0 |
| 0033 | 57 |  | MOV | D,A | ;Временное запоминание ;2x(t) в регистре D |
| 0034 | AF |  | XRA | A | ;Подготовка к записи в акку-;мулятор x(t) |
| 0035 | 7B |  | MOV | A,E | ;x(t) — в аккумуляторе |
| 0036 | 0F |  | RRC |  | ;Получение 0,5x(t) в аккуму-;ляторе |
| 0037 | A7 |  | ANA | A | ;Сброс флага переноса: ;CY=0 |
| 0038 | 82 |  | ADD | D | ;(A)+(D)=2,5x(t) в аккумуля-;торе |
| Ад-рес | Машин-ный код | Метка | Мне-мони-ка | Операнд | Комментарий |
| 0039 | CD 6C 00 |  | CALL | MULT | ;Вызов подпрограммы умно-;жения однобайтных слов; ;множимое: 2,5x(t) — в рег. ;А; множитель: x(t) — в рег. ;Е; произведение: 2,5x2(t) — ;в H, L |
| 003C | AF |  | XRA | A | ;Подготовка к обнулению ;РОНов |
| 003D | 57 |  | MOV | D,A | ;Обнуление регистра D |
| 003E | 5A |  | MOV | E,D | ;Обнуление регистра E |
| 003F | 42 |  | MOV | B,D | ;Обнуление регистров B, C |
| 0040 | 48 |  | MOV | C,B |
| 0041 | 54 |  | MOV | D,H | ;Запоминание 2,5x2(t) в ре-;гистровой паре D, E |
| 0042 | 5D |  | MOV | E,L |
| Обработка логического блока | | | | | |
| 0043 | F1 |  | POP | PSW | ;y(t-N) — в аккумуляторе |
| 0044 | BF |  | CMP | A | ;Знак результата — в ре-;гистре F |
| 0045 | FA 4B 00 |  | JM | M2 | ;При S=1 переход на метку ;М2 |
| 0048 | F2 57 00 |  | JP | M3 | ;При S=0 переход на метку ;М3 |
| 004B | 2F | M2: | CMA |  | ;INV(y(t-N)) |
| 004C | 3C |  | INR | A | ;INV(y(t-N))+1 — дополни-;тельный код |
| 004D | 47 |  | MOV | B,A | ;Дополнительный код в ре-;гистровой паре B, C |
| 004E | 09 |  | DAD | B | ;Сложение двухбайтовых ;слов. Результат — в H, L |
| Вывод результата | | | | | |
| 004F | AF |  | XRA | A | ;Подготовка к выводу млад-;шего байта |
| 0050 | 7D |  | MOV | A,L |
| 0051 | D3 04 |  | OUT | PORTV | ;Вывод младшего байта |
| 0053 | AF |  | XRA | A | ;Подготовка к выводу стар-;шего байта |
| 0054 | 7C |  | MOV | A,H |
| 0055 | D3 04 |  | OUT | PORTV | ;Вывод старшего байта |
| 0057 | AF | M3: | XRA | A | ;Процедура вывода млад-;шего байта при S=0 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ад-рес | Машин-ный код | Метка | Мне-мони-ка | Операнд | Комментарий |
| 0058 | 67 |  | MOV | H,A | ;Обнуление регистровой па-;ры H, L |
| 0059 | 6F |  | MOV | L,A |
| 005A | EB |  | XCHG |  | ;2,5x2(t) из регистровой па-;ры (D, E) поменять местами ;с содержимым (H, L). Об-;нуление регистровой пары ;D, E |
| 005B | 3A 0A 80 |  | LDA | 800AH | ;y(t-N) — в аккумуляторе |
| 005E | 57 |  | MOV | D,A | ;y(t-N) — в регистровой ;паре D, E для суммирова-;ния |
| 005F | 19 |  | DAD | D | ;Сложение двухбайтовых ;слов. Результат — в H, L |
| 0060 | AF |  | XRA | A | ;Подготовка к выводу млад-;шего байта |
| 0061 | 7D |  | MOV | A,L |
| 0062 | D3 04 |  | OUT | PORTV | ;Вывод младшего байта |
| 0064 | AF |  | XRA | A | ;Подготовка к выводу стар-;шего байта |
| 0065 | 7C |  | MOV | A,H |
| 0066 | D3 04 |  | OUT | PORTV | ;Вывод старшего байта |
| 0068 | C3 00 00 |  | JMP | M0 | ;Безусловный переход к но-;вому циклу вычислений |
| 006B | 76 |  | HLT |  | ;Останов |
| 006C | 21 00 00 | MULT: | LXI | H,0000H | ;Начало подпрограммы ум-;ножения. Сброс регистра ;суммы частичных произве-;дений |
| 006F | 06 08 |  | MVI | B,08H | ;Начальная установка счет-;чика циклов |
| 0071 | 29 | M4: | DAD | H | ;Логический сдвиг H, L на ;разряд влево |
| 0072 | 07 |  | RLC |  | ;Очередной разряд множи-;теля с переносом |
| 0073 | D2 77 00 |  | JNC | M5 | ;Переход на метку М5, если ;след. разряд нуль |
| 0076 | 19 |  | DAD | D | ;Добавление множимого к ;частичной сумме |
| 0077 | 05 | M5: | DCR | B | ;Счет циклов |
| 0078 | C2 71 00 |  | JNZ | M4 | ;Переход, если вычисления ;не закончены |
|  |  |  |  |  |  |
| Ад-рес | Машинн-ый код | Метка | Мне-мони-ка | Операнд | Комментарий |
| 007B | C9 |  | RET |  | ;Возврат в основную про-;грамму |
|  |  |  | END |  | ;Директива достижения кон-;ца программы |

**Пояснения к примеру выполнения** программирования на машинном языке и наязыке ассемблер реализации алгоритма работы устройства, задан-ного уравнением V(t)=kx2(t)+y(t-N) [5].

Блок начальных директив не обязателен. Он обычно вводится для удобства программирования. Адреса портов — 01H, 02H, 04H, 08H и т. д. Таким образом, для каждого порта выделяется отдельная линия.

Стек и очередь требуются в том случае, когда одна из переменных зависит от *t-N*. Если все переменные зависят от *t*, стек и очередь не нужны.

При наличии переменной, зависящей от *t-N*, сначала вводятся все значения этой переменной при *t…t-N* в виде очереди. Затем данные из очереди пересылаются в стек. При этом последним элементом оказывается значение переменной при *t-N*, которое извлекается первым и используется в вычис-лениях.

Организация стека и очереди является подготовкой для ввода соответствующей переменной, зависящей от *t-N* (в примере — *y(t-N)).* В команде LXI H, 800AH младший байт операнда равен значению N (10=0AH). Значение константы N, как и других констант, должно быть представлено в 16-ричной системе.

Команды NOP используются для согласования быстродействующего микропроцессора с медленным устройством ввода. Так как микросхемы интер-фейса ввода-вывода и периферийные устройства не заданы, количество этих команд не принципиально.

Умножение переменной на константу выполняется с помощью операций циклического сдвига RLC (умножение на 2) и RRC (умножение на 0,5). После каждой операции требуется команда ANA — сброс флага переноса, так как команды циклического сдвига влияют на этот флаг. Затем полученные результаты суммируются соответствующим образом.

Для перемножения переменных и возведения в квадрат используется специальная подпрограмма умножения, которая приведена в конце программы. Если заданы 3 переменные, требуется добавить соответствующие блоки и операции.

Сложение числа в дополнительном коде эквивалентно вычитанию этого числа в прямом коде. Это позволяет заменить вычитание двухбайтных чисел

их сложением.

Рассмотрим фрагмент прикладной программы, содержание которой можно описать следующим образом (рис. 2).

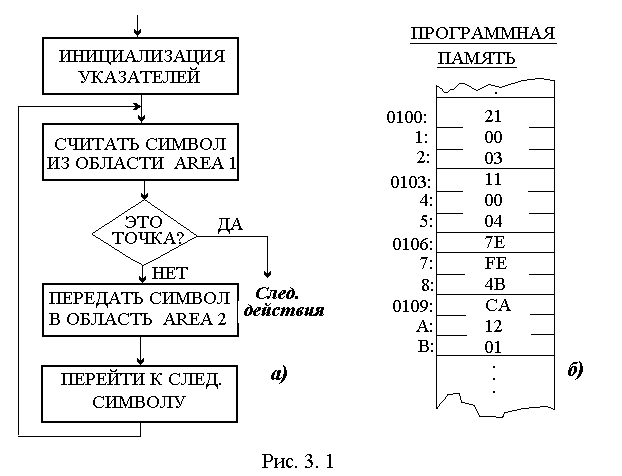


Рис. 2

В области памяти с начальным адресом 0300Н находится текст в виде последовательности символов. Каждый символ закодирован 8-ю битами и размещен в одной ячейке памяти. Часть текста до первого появления кода точки требуется переслать в другую область памяти с начальным адресом 0400 (здесь и в дальнейшем для краткости будем опускать признак 16-ричного кода – латинскую букву Н). В случае обнаружения кода точки 4В (0100 1011) передача прекращается и программа переходит к другим действиям. Фрагмент программы начинается с адреса 0100. Область-источник будем называть AREA 1, а область-получатель – AREA 2.

Формулировка задачи, по существу, является простой циклической про-граммой.

При программировании не следует использовать длинные команды с прямой адресацией. В рассматриваемом примере удобно ввести два 16-битных указателя памяти, адресующих текущие ячейки AREA 1, AREA 2. Через указатели с помощью косвенной адресации можно обращаться к памяти 1-байтными командами. В качестве указателя текущей ячейки для AREA 1 используем регистры H, L , а для AREA 2 – регистры D, E.

Программа начинается с инициализации указателей. В регистры H, L загружается начальный адрес 0300, а в регистры D, Е – адрес 0400. Используется 3-байтная команда непосредственной загрузки LXI r,p (r,p – регистровая пара). Загрузка в Н-пару имеет код операции 21, а в D-пару – 11. Следует отметить, что <B2> - младший байт адреса, а <B3> - старший. Стандартный формат бланка:

АДРЕС КОД ОПЕРАЦИИ <B2> <B3>

0100 21 00 03

0103 11 00 04

Размещение программы в памяти показано на рис. 2, *б.* Сравнить код символа с заданным можно только через аккумулятор с помощью пересы-лочной команды *MOV A, M*, имеющей код операции 7*Е*. По этой команде содержимое ячейки памяти, адрес которой содержится в регистровой паре *H, L*, пересылается в аккумулятор. Заключительные шаги программы по опре-делению первого появления кода точки приведены в табл. 2.

Если символ в аккумуляторе соответствует коду точки, то это фикси-руется установкой флага *Z*=1 в регистре флагов, и программа переходит к другим действиям. При наличии несоответствия *Z*=0 и необходимо переслать символ из аккумулятора в область *AREA* 2 и анализировать следующий символ текста. Выход из цикла происходит по команде условного перехода JZ с кодом операции СА, но адрес перехода неизвестен.

*Таблица 2*

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код операции | <B2> | <B3> |
| 0106 | 7E |  |  |
| 0107 | FE | 4B |  |
| 0109 | CA | ? | ? |
| 010C | 12 |  |  |
| 010D | 23 |  |  |
| 010E | 13 |  |  |
| 010F | C3 | 06 | 01 |
| 0109 | CA | 12 | 01 |

Зарезервируем две ячейки памяти по команде *СА* (адрес 0109). Поэтому следующий шаг программы начинается с адреса 010С. По этому адресу однобайтной командой *STAX* *В* в область *AREA* 2 передается символ из аккумулятора (код операции 12).

На этом обработка одного символа закончена, и необходимо проверить следующий. Для этого следует увеличить на единицу начальные адреса областей с помощью однобайтных команд инкремента *INX H* (код операции 23) и *INX D* (код операции 13).

Повторение цикла производится командой безусловного перехода *JMP* (код операции *С*3) к ячейке с адресом 0106, содержащей команду *MOV A, M* (код операции 7*Е*).

Последняя запись в табл. 2 позволяет определить адрес перехода в команде *JZ* – 0112. Этим завершается составление фрагмента программы на машинном языке.

**Задание**

Составить фрагменты прикладных программ в машинных кодах и на языке ассемблер (по выбору студентов, но не менее четырех).

.

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Фрагменты прикладных программ в машинных кодах и на языке ассем-блер.

3. Выводы по занятию.

**Занятие 25**

**по теме «Микроконтроллер КМ1816ВЕ48»**

**Теоретическая часть занятия**

Все команды микроконтроллера (ОЭВМ) КМ1816 имеют формат в 1 или 2 байта и выполняются за один или два машинных цикла (2,5 или 5 мкс соответственно при тактовой частоте 6 МГц). За два машинных цикла выполняются все команды с непосредственным операндом; команды ввода/вывода; команды передачи управления, вызова подпрограмм и некоторые команды пересылок. Все остальные команды – за один машинный цикл.

В командах КМ1816 используются прямая, непосредственная и косвен-ная адресации.

Группа команд арифметических и логических операций операции сло-жения, инкремента и декремента, сдвигов, конъюнкции, дизъюнкции, отри-цания (инверсии), суммы по модулю два и другие включает логические операции над непосредственным операндом, содержимым аккумулятора, регистров или ячеек памяти.

Все более сложные операции (умножение, деление, возведение в степень и проч.) выполняются по подпрограммам.

Из арифметических операций наиболее просто (с помощью однобайтной команды) выполняется операция сложения однобайтных чисел. Операция вычитания в списке команд отсутствует, и для ее выполнения (для одно-байтных чисел) эта операция заменяется сложением с числом в дополни-тельном или обратном коде.

Для выполнения более сложных действий – операций сложения и вычи-тания двухбайтных чисел, а также умножения целых однобайтных чисел без знака приходится использовать подпрограммы.

В подпрограммах обычно регистры *R0* и *R1* используются в качестве регистров косвенного адреса, регистр *R2 –* как расширитель аккумулятора (при операциях с двухбайтными словами), а регистр *R3 –* как счетчик программных циклов.

При операциях с двухбайтными словами регистр косвенного адреса указывает на старший байт числа, а аккумулятор применяется в качестве ис-точника одного из операндов и места фиксации результата. Сказанное пояс-няется примерами в учебных пособиях [1, 3].

**Задание**

1. Записать программу сложения двух двухбайтных чисел: последние цифры шифра студента в прямом порядке и эти цифры шифра в обратном порядке.

Например, эти цифры шифра в прямом порядке-3461, соответственно в обратном порядке-1643.

2. Записать программу логического сдвига полученного результата сложения двухбайтного числа влево.

3. Записать программу умножения однобайтных чисел. Первое число – две последних цифры шифра. Второе число - две последних цифры шифра в обратном порядке, например 61 и 16.

**Содержание отчета**

1. Титульный лист с названием занятия.

2. Основные теоретические положения.

3. Программы сложения двух двухбайтных чисел, логического сдвига полученного результата сложения, умножения однобайтных чисел.

4. Выводы по занятию.

**4. Блок контроля освоения дисциплины**

**4.1. Задания на контрольную работу**

**и методические указания к их выполнению**

По изучаемому материалу дисциплины «Цифровые устройства и микро-процессоры» учебным графиком предусмотрено выполнение одной кон-трольной работы, которая состоит из двух заданий. Номер варианта каждого задания выбирается из таблиц по двум последним цифрам шифра студента, чем обеспечивается многовариантность заданий.

4.1.1. Задания на контрольную работу

***З а д а н и е 1*.** Логическая функция 4-х булевых переменных *x1, x2, x3,* *x4* имеет истинное значение на тех наборах входных переменных, которые эквивалентны десятичным числам, указанным в табл. 1, т. е. заданы в числовом виде. Требуется:

1.1. Построить таблицу истинности для полного набора функций 4-х указанных переменных и определить из нее СДНФ, удовлетворяющую условиям задачи.

1.2. Минимизировать полученную функцию графическим методом Карно-Вейча или табличным – Квайна-МакКласки.

1.3. На основании полученной тупиковой формы переключательной функции построить логическую схему на микросхемах средней степени интеграции.

*Таблица 1*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Послед.  цифра шифра | Предпоследняя цифра шифра | | | | |
| 0,5 | 1,6 | 2,7 | 3,8 | 4,9 |
| 0 | 0, 3, 7, 10,  11, 13, 15 | 2, 5, 8, 12,  13, 14, 15 | 0, 2, 3, 6,  7, 12, 13 | 1, 4, 8, 10,  11, 12, 14, 15 | 0, 1, 2, 4,  5, 6, 11, 14 |
| 1 | 0, 1, 5, 7,  9, 10, 13, | 1, 2, 6, 8,  10, 14, 15 | 2, 3, 4, 6,  10, 11, 13 | 0, 7, 9, 10,  13, 14, 15 | 2, 3, 6, 7,  9, 10, 11, 12 |
| 2 | 2, 4, 5, 6,  7, 8, 15 | 3, 4, 6, 8,  11, 12, 14 | 4, 7, 8, 10,  12, 14, 15 | 1, 4, 5, 6,  11, 12, 14, | 0, 2, 3, 5,  8, 10, 13 |
| 3 | 2,3, 6, 7 10,  11, 12, 13 | 1, 3, 6, 9,  11, 12, 13, | 0, 1, 7, 8,  10, 12, 14 | 2, 3, 5, 10,  11, 12, 13 | 0, 1, 3, 6, 8,  9, 12, 13 |
| 4 | 0, 1, 10, 11,  12, 14, 15 | 0, 1, 2, 4,  5, 6, 11, 14 | 1,4,8,10,11,  12, 14, 15 | 0, 3, 6, 7,  8, 14, 15 | 1, 3, 6, 8,  9, 12, 13 |
| 5 | 3, 4, 5, 6,  7, 8, 9, 11 | 1, 2, 5, 7,  10, 13, 15 | 0, 3, 7, 8,  11,12,14,15 | 1, 3, 5, 6,  8, 9, 10, 13 | 0, 3, 4, 10, 12,13,14,15 |
| *Окончание таблицы 1* | | | | | |
| 6 | 2, 3, 5, 7,  9, 10, 11 | 3, 4, 5, 6,  7, 8, 9, 11 | 0, 1, 4, 5,  6, 11, 14 | 1, 4, 6, 11,  12, 14, 15 | 0, 2, 4, 6,  11, 13, 15 |
| 7 | 0, 2, 4, 6,  7, 9, 13 | 0, 1, 2, 5,  7, 9, 13, 14 | 2, 3, 6, 7,  8, 11, 12 | 0, 4, 5, 6,  7, 9, 14 | 2, 3, 4, 5,  12, 13, 14 |
| 8 | 3, 6, 10, 11,  12,13,14,15 | 1, 3, 4, 6,  7, 8, 11, 15 | 0, 2, 7, 8,  9, 10, 15 | 2, 5, 6, 8, 9,  10, 11, 12 | 4, 6, 8, 9,  11, 13, 15 |
| 9 | 1,5,8,10,11,  12, 14, 15 | 0, 2, 3, 7,  9,11,12,15 | 2, 3, 4, 5,  8, 10, 11 | 0, 2, 4, 8,  10, 13, 15 | 1, 2, 3, 4,  5, 9, 10, 13 |

***З а д а н и е 2*.** Выполнить следующие операции по преобразованию сис-тем счисления:

2.1. Число, содержащее целую и дробную части, заданное в форме с фиксированной запятой (точкой) в десятичной системе счисления (табл. 2), перевести в числа, представленные в двоичной системе счисления, в двоично-десятичной системе (код 8421) и в 16-ричной системе счисления. Выполнить не менее двух вариантов, которые выбираются произвольно.

2.2. Число в 16-ричной системе счисления (табл. 3) представить в двоично-десятичной системе счисления (код 8421) и в десятичной системе счисления.

2.3. Числа, представленные в табл. 4 в десятичной системе счисления, перевести в двоично-десятичную систему (код 8421), сложить алгебраически, предварительно представив отрицательные числа в дополнительном коде, а результат – в десятичной системе счисления.

2.4. Число из табл. 2 представить в форме с плавающейзапятой (точ-кой) с точностью до четвертого десятичного знака после запятой.

*Таблица 2*

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Послед-няя цифра шифра | Вариант 1 | | Вариант 2 | | Вариант 3 | Вариант 4 | | Вариант 5 | | | |
| 0 | 156.814 | | 136.112 | | 145.114 | 135.121 | | 138.315 | | | |
| 1 | 346.213 | | 245.314 | | 345.213 | 241.413 | | 124.101 | | | |
| 2 | 458.112 | | 247.211 | | 269.131 | 679.451 | | 691.314 | | | |
| 3 | 136.101 | | 126.141 | | 257.101 | 345.131 | | 245.913 | | | |
| 4 | 368.219 | | 678.121 | | 157.123 | 679.141 | | 129.141 | | | |
| *Окончание табл. 2* | | | | | | | | | | | |
| 5 | 167.891 | | 129.213 | | 367.112 | 789.131 | | 246.211 | | | |
| 6 | 234.135 | | 135.121 | | 378.121 | 246.135 | | 236.712 | | | |
| 7 | 456.811 | | 124.411 | | 459.145 | 126.812 | | 451.121 | | | |
| 8 | 236.711 | | 141.415 | | 145.610 | 367.913 | | 589.123 | | | |
| 9 | 467.159 | | 246.191 | | 691.214 | 357.149 | | 571.532 | | | |
|  | | |  | |  | | |  | |  |  |

*Таблица 3*

|  |  |
| --- | --- |
| Предпосл. цифра шифра | Последняя цифра шифра |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 3FH | 4EH | 2DH | 3CH | 3AH | 3BH | 41H | 57H | A8H | B7H |
| 1 | 6CH | 7DH | 9EH | 9FH | A0H | A1H | A2H | A3H | A4H | A5H |
| 2 | A5H | A7H | A8H | A9H | AAH | ABH | ACH | ADH | AEH | AFH |
| 3 | B1H | B1H | B2H | B3H | B4H | B5H | B6H | B7H | B8H | B9H |
| 4 | BAH | BBH | BCH | BDH | BEH | BFH | 96H | 97H | 98H | 99H |
| 5 | 4AH | 5BH | 6CH | 7DH | 8EH | 9FH | 77H | 78H | 79H | 85H |
| 6 | E1H | E0H | DFH | DEH | DDH | DCH | DBH | DAH | D9H | D8H |
| 7 | EBH | EAH | E9H | E8H | E7H | E6H | E5H | E4H | E3H | E2H |
| 8 | F5H | FF4H | F3H | F2H | F1H | F0H | EFH | EEH | EDH | ECH |
| 9 | FFH | FEH | FDH | FCH | FBH | FAH | F9H | F8H | F7H | F6H |
|  |  |  |  |  |  |  |  |  |  |  |

*Таблица 4*

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Предпослед-няя цифра шифра | Последняя цифра шифра | | | | | | | | | |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | -26  54 | -54  26 | 56  -28 | -60  89 | -31  28 | 59  48 | -67  28 | 69  -24 | -73  89 | -33  85 |
| 1 | -28  56 | -56  28 | 57  -79 | -61  88 | 64  -27 | 51  49 | -68  27 | 69  -25 | -75  85 | 85  34 |
| 2 | 24  -14 | 91  -74 | 81  45 | 92  -44 | -56  48 | -36  37 | 39  -49 | 56  -37 | 81  -41 | 34  -72 |
| 3 | 56  -91 | 49  36 | 58  -28 | 64  -32 | 91  -34 | 45  -26 | 23  -28 | 25  75 | 71  -24 | 99  -26 |
| 4 | -36  89 | 34  -65 | 24  -46 | 84  -28 | 71  -28 | 91  -27 | 98  -25 | 97  -23 | 95  -24 | 92  -25 |
| 5 | 49  -75 | 48  -28 | 47  -23 | 45  -22 | 43  -21 | 41  20 | 39  -18 | 37  -25 | 38  -85 | 43  -84 |
| 6 | 45  -23 | 47  -25 | 49  -99 | 51  -73 | 53  -48 | 43  -29 | -43  29 | -53  48 | -51  73 | -49  99 |
| 7 | -33  48 | -35  49 | -37  51 | -39  53 | 63  -39 | 39  -53 | 37  -51 | 37  -50 | 35  -47 | 39  -23 |
| 8 | 63  -23 | 65  -25 | 67  -24 | 68  -23 | 73  -28 | -73  28 | -68  23 | -67  24 | -65  25 | -69  23 |
| 9 | -27  55 | -28  56 | -29  57 | -30  58 | -31  59 | 33  -60 | 34  -61 | 35  -60 | 37  -62 | 39  -64 |

4.1.2. Методические указания к выполнению контрольной работы

При выполнении задания 1рекомендуется пользоваться учебным посо-бием [2], с. 16 … 20, где рассматриваются вопросы минимизации логической функции 4-х переменных, заданной числовым способом. Следует также обратить внимание на то, что в табл. 1 логическая функция, подлежащая минимизации, задана в дизъюнктивной форме.

Студенту предоставляется возможность самому решить, какой метод минимизации выбрать при решении этой задачи. Следует только помнить, что табличный метод Квайна-МакКласки обычно используется при минимизации логических функций, содержащих больше пяти переменных.

При построении логической схемы на основании полученной тупиковой формы следует использовать однотипные логические элементы средней степени интеграции, которые можно выбрать из справочника [10].

При выполнении Задания 2следует учитывать, что вопросы, отно-сящиеся к преобразованию чисел из одной системы счисления в другую, достаточно подробно освещены в [1] и [2].

При выполнении арифметических операций следует обратить внимание на суммирование чисел, представленных в двоично-десятичном коде 8421. В этом коде суммирование выполняется в два этапа. На первом этапе вычисляется предварительная сумма Sп. Здесь суммирование производится по обычным правилам двоичной арифметики, но при этом следует сделать несколько уточнений и дополнений:

если в i-й тетраде предварительной суммы Sпi образуется естественный перенос, то он учитывается в следующей (i + 1) – й тетраде;

если в i-й тетраде естественного переноса нет, но Sпi≥ 10, то считается, что есть искусственный перенос и его следует учитывать в (i + 1) – й тетраде; при Sпi< 10 переноса нет.

На втором этапе каждая тетрада Sпi корректируется по следующему правилу: если Sпi ≥ 10, т. е. в данной тетраде образовался естественный или искусственный перенос, то к ней прибавляется число 6, а получающийся при этом перенос не учитывается. Естественный перенос обозначим 1е, искус-ственный – 1и.

Сказанное поясним на примере.

Предположим, что требуется сложить два десятичных числа: 0.597 и 0.346. Слева показано сложение обычным способом – в столбик, а справа все цифры представлены в коде 8421:

1и 1и

0.597 0.0101 1001 0111

+ 0.346 + 0.0011 0100 0110

0.943 0.1001 1110 1101 Предварительная сумма

+

0000 0110 0110 Коррекция

0.1001 0100 0011 Сумма

Второй пример показывает, как образуется естественный перенос. Сложим еще два числа: 0.098 и 0.729:

1и 1е

0.098 0.0000 1001 1000

+ 0.729 + 0.0111 0010 1001

0.827 0.1000 1100 0001 Предварительная сумма

+ 0.0000 0110 0110 Коррекция

0.1000 0010 0111 Сумма

Представление чисел в форме с плавающей запятой (точкой) поясняется в [2], с. 28 …30. Отвечая на этот пункт задания 2, следует обратить внимание на точность представления чисел с плавающей точкой.

**4.2. Задание на курсовой проект**

**и методические указания к его выполнению**

# 4.2.1. Задание на курсовой проект

Объектом курсового проекта является микропроцессорное устройство обработки информации на базе однокристального МП комплекта серии КР580, реализующее заданную функцию обработки радиотехнических сигналов.

В процессе выполнения курсового проекта студент должен:

- составить алгоритм решения поставленной задачи;

- составить программы на языке ассемблер и в машинных кодах.

Исходные задания для проектирования приведены в табл. 1 и в табл. 2. Варианты задания выбираются по двум последним цифрам шифра студента.

*Таблица* 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Выход-ной сигнал | Послед-няя цифра шифра | Входные сигналы (предпоследняя цифра шифра) | | |
| 0, 1 | 2, 3 | 4, 5 |
| V(t) | 0, 1 | x2(t)+ky2(t)+y(t-N) | kx2(t)+Ny(t) | kx(t)z(t)+y(t-N) |
| W(t) | 2, 3 | x2(t)+ky(t)+z(t-N) | kx2(t)+Ny(t) | Nx2(t)+ky(t) |
| P(t) | 4, 5 | kx(t)y(t)+z(t-N) | kz(t)y(t-N) | 2,5x2(t)+ky(t-N) |
| Q(t) | 6, 7 | y2(t)-ky(t-N) | x(t)+kz2(t-N) | Nx(t)+ky2(t) |
| M(t) | 8, 9 | ky2(t)-x(t-N) | Nx(t)y(t)+kz(t-N) | kx(t)z(t)+y(t-N) |
|  | | 6, 7 | 8, 9 | |
| V(t) | 0, 1 | kx(t)y(t)+Nz(t)  2,5x2(t)-ky(t)+z(t-N) | x2(t)+ky(t)+z(t-N) |  |
| W(t) | 2, 3 | 3,5kx(t)-8Ny(t-N)z(t) |  |
| P(t) | 4, 5 | (x(t)-y(t))2-kz(t-N) | z2(t)-2,5ky(t-N) |  |
| Q(t) | 6, 7 | Nx(t)y(t)+0,5kz(t-N) | kx(t)y(t-N)+0,5z2(t) |  |
| M(t) | 8, 9 | 0,5ky(t-N)-x2(t) | x2(t)-ky(t-N)+3,5z(t) |  |

*Таблица* 2

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Параметры | Предпоследняя цифра шифра | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Константа k | 0,5 | 9,5 | 6,5 | 4,0 | 5,5 | 2,5 | 8,5 | 3,5 | 7,0 | 4,5 |
| Константа N | 4 | 6 | 8 | 10 | 7 | 9 | 5 | 6 | 8 | 3 |
| Длина входных сигналов (x, y, z), байт | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Длина выходных сигналов (V, W, P, Q, M), байт | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

Пояснительная записка к курсовому проекту должна содержать:

а) титульный лист, на котором указываются название дисциплины, темы курсового проекта: фамилия, имя и отчество студента, его шифр;

б) задание на курсовой проект;

в) алгоритм решения задачи;

г) программы на языке ассемблера и в машинных кодах;

д) выводы по работе.

4.2.2. Методические указания к выполнению курсового проекта

***Требования к блок-схеме алгоритма***

При вычерчивании блок-схемы алгоритма следует использовать графи-ческие символы согласно ГОСТ 19.003-80.

Структура блок-схемы, а также содержание записей, помещаемых внутри блоков или рядом с ними, не регламентируется. Но при этом необходимо придерживаться следующих правил и рекомендаций (ГОСТ 19.002-80):

1. Записи в блок-схемах должны быть ясными и однозначно определять, какой этап решения задачи данный блок отражает;

2. Номер блока ставится слева вверху в разрыве контура символа;

3. Нормативным направлением линий считается направление сверху вниз и слева направо;

4. Линии потока проводятся только по вертикали и горизонтали и подводятся к осевым линиям символов. Для линий, связывающих два элемента блок-схемы, допускается не более трех операторов.

5. Линии потока можно прерывать. Для указания связи между преры-ваниями используется символ «соединитель».

**Составление блок-схемы алгоритма и программы**

При составлении блок-схемы алгоритма заданный алгоритм решения задачи представляется графически в виде отдельных геометрических фигур (табл. 3), взаимное расположение которых отображает последовательность решения задач на проектируемом устройстве. Составленная блок-схема алго-ритма должна сопровождаться подробным описанием действий, выполняемых в каждом блоке.

Для составления блок-схемы алгоритма необходимо изучить особен-ности системы команд того микропроцессорного комплекта (МПК), на базе которого строится микропроцессорное устройство. В частности, для курсового проекта на базе МПК КР580, необходимо изучить структуру микропроцессора КР580ВМ80 и его систему команд, а также директивы ассемблера [1,3].

Рекомендуется использовать стандартные приемы программирования: организацию счета, временную задержку, сложение и вычитание, умножение и деления и др., описания которых даны в разделе 11[1]. Стандартные участки необходимо оформить в виде подпрограмм.

Разработка рабочей программы должна сопровождаться пояснениями (комментариями), которые пишутся в поле комментария ассемблерной строки, но игнорируются ассемблирующей программой при трансляции в машинные коды.

В практическом занятии № 24 рассматривается пример выполнения курсового проекта с написанием программ на языке ассемблер и в машинных кодах.

**4.3. Текущий контроль**

**Тест №1**

к разделу 1

1. Функция стрелки Пирса обозначается и определяется формулой

А)***;***

Б) ***;***

В) ;

Г)***.***

2. Условное обозначение элемента, выполняющего операцию конъюнкции, представлено на схеме

А) Б) В) Г)

&

∞

¥

1

3. Закон ассоциативности записывается в виде

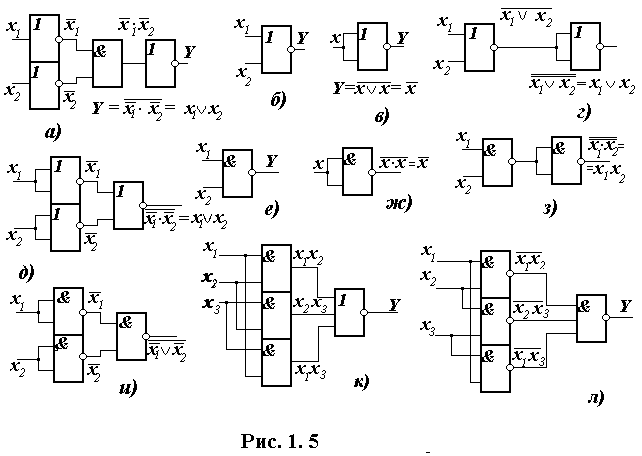
А)

Б) 

В) 

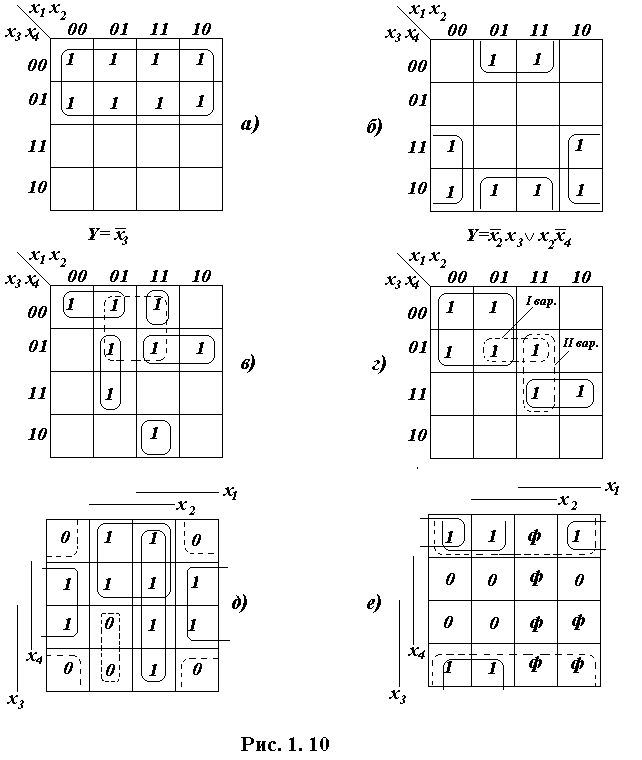
Г) 

4. Схема, приведенная на рисунке, реализует функцию



А) ИЛИ; Б) НЕ; В) И; Г) И – НЕ.

5. Карта Карно имеет вид, показанный на рисунке. Укажите реализуемую функцию



А) 

Б)

В) 

Г) 

**Тест №2**

к разделу 2

1. Если *р* – основание системы счисления, *k* ***–*** номер разряда, то общий алгоритм представления числа в какой-либо системе счисления определяется равенством

А) *N = … a2p2  + a1p1 + a0p0 + a-1p -1 + a -2p -2*… ;

Б) *N =( … a2p2  + a1p1 + a0p0)\*( a-1p -1 + a -2p -2*…*);*

В) *N =( … a2p2  + a1p1)/( a0p0+ a-1p -1 + a -2p -2*…*);*

Г) *N = … a -2p -2  ++ a-1p -1 + a0p0 + a1p1 + a2p2*…

2. В таблице десятичные числа закодированы

|  |  |
| --- | --- |
| 0 | 0000 |
| 1 | 0001 |
| 2 | 0010 |
| 3 | 0011 |
| 4 | 0100 |
| 5 | 0101 |
| 6 | 0110 |
| 7 | 0111 |
| 8 | 1000 |
| 9 | 1001 |

А) кодом Грея 7421;

Б) кодом 8421(BCD-код);

В) кодом Айкена 2421;

Г) кодом Джонсона.

3. При пересчете целой части десятичного числа в 8-ричное произ-водится операция

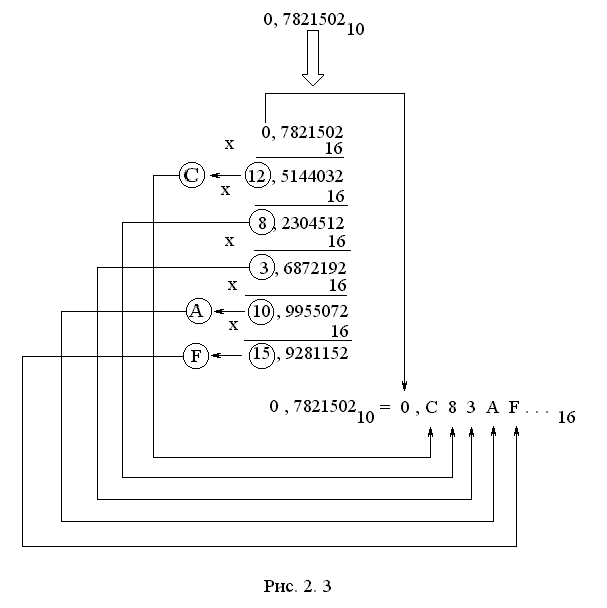
А) умножения на 8;

Б) деления на 8;

В) вычитания 8;

Г) сложения с 8.

4. На рисунке показана процедура перевода десятичного числа 0,782150210 в



А) двоично-десятичную систему счисления;

Б) восьмеричную систему счисления;

В) шестнадцатеричную систему счисления;

Г) двоично–кодированную десятичную систему счисления.

5. Выражение 

показывает процедуру перевода восьмеричного числа в

А) двоично-десятичное;

Б) восьмеричное;

В) октавное;

Г) двоичное.

**Тест №3**

к разделу 3

1. Ниже представлена таблица истинности

|  |  |  |
| --- | --- | --- |
| ***x1*** | ***x2*** | Y |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

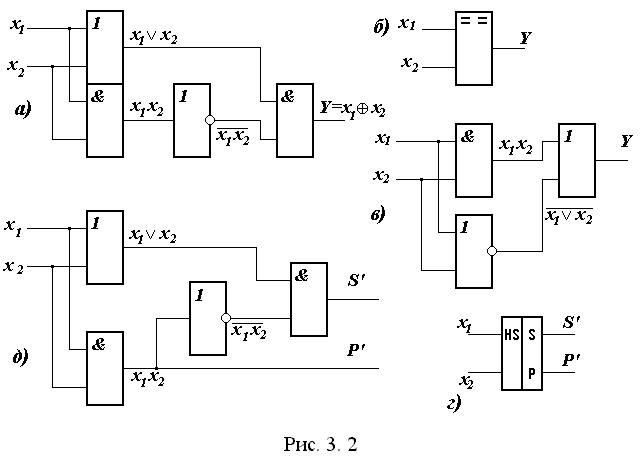
А) полусумматора;

Б) устройства равнозначности;

В) устройства неравнозначности;

Г) полного сумматора.

2. На рисунке приведена принципиальная схема



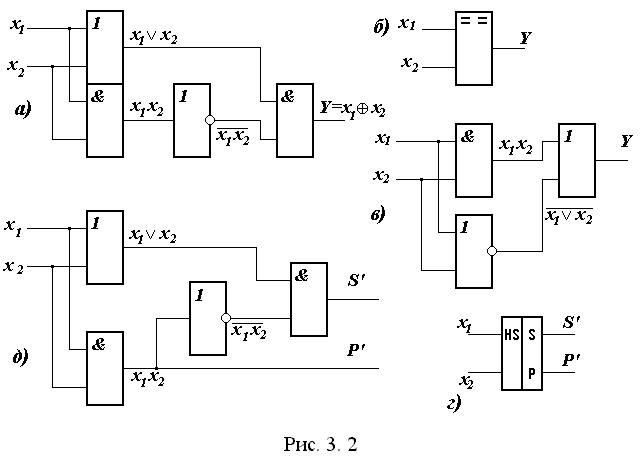
А) устройства неравнозначности;

Б) устройства равнозначности;

В) устройства полусумматора;

Г) устройства отрицания дизъюнкции.

3. На рисунке приведена принципиальная схема



А) полного сумматора;

Б) устройства отрицания дизъюнкции;

В) устройства равнозначности;

Г) устройства полусумматора.

4. Ниже представлена таблица истинности

|  |  |  |  |
| --- | --- | --- | --- |
| ***x1*** | ***x2*** | ***S*** | ***P*** |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

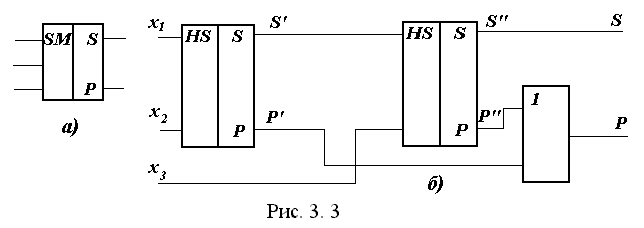
А) полусумматора;

Б) устройства равнозначности;

В) полного сумматора;

Г) устройства отрицания дизъюнкции.

5. На рисунке приведена принципиальная схема



А) устройства равнозначности;

Б) полусумматора;

В) полного сумматора;

Г) устройство сравнения кодов.

**Тест №4**

к разделу 4

1. Приведенная ниже таблица переключений характеризует работу

|  |  |  |  |
| --- | --- | --- | --- |
| ***S****n* | ***R****n* | ***Q****n* | ***Qn+****1* |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | X |
| 1 | 1 | 1 | X |

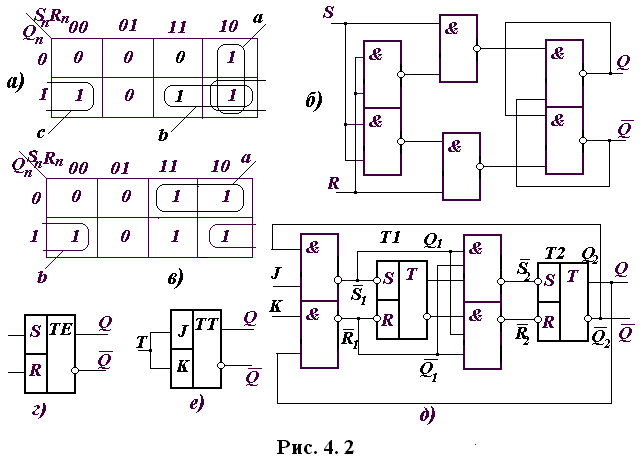
А) JK – триггера;

Б) асинхронного RS – триггера;

В) D – триггера;

Г) синхронного RS – триггера.

2. На рисунке карты Карно проведена минимизация функции Qn+1 графи-ческим методом, дизъюнктивная форма записи которой будет иметь вид

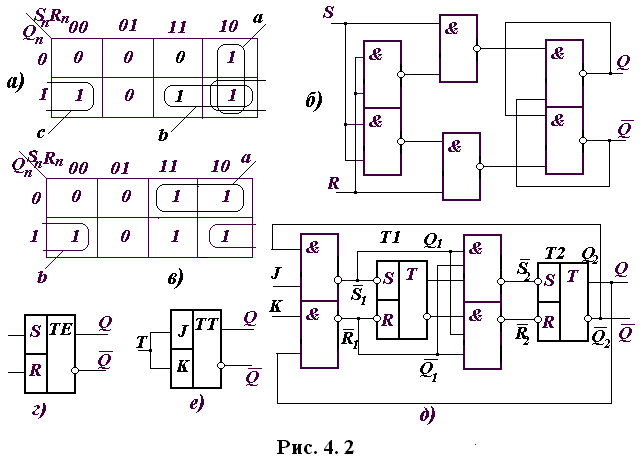


***b***

А) 

Б)  
В)  
Г) 

3. Ниже приведена принципиальная схема



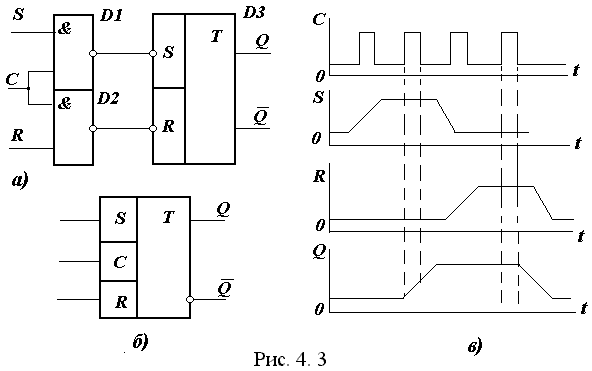
А) JK – триггера;

Б) RS – триггера с инверсными входами;

В) Е – триггера;

Г) синхронного RS – триггера.

4. На рисунке представлена схема



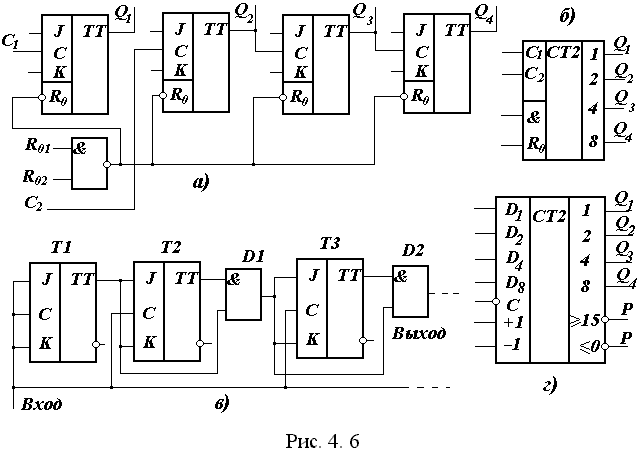
А) асинхронного RS – триггера;

Б) счетного триггера;

В) Е – триггера;

Г) синхронного RS – триггера.

5. Ниже приведена схема счетчика



А) с параллельным переносом;

Б) с комбинированным переносом;

В) с последовательным переносом;

Г) с внешним переносом.

**Тест №5**

к разделу 5

1. Устройство, преобразующее сигнал из аналоговой формы в цифровую, на-

зывается

А) формирователь импульсов;

Б) цифроаналоговый преобразователь;

В) кодопреобразователь;

Г) аналого-цифровой преобразователь.

2. Шаг квантования определяется

А) формой уровней квантования;

Б) расстоянием между импульсами в цифровом сигнале;

В) расстоянием между уровнями квантования;

Г) минимальной амплитудой входного сигнала.

3. Максимальная ошибка квантования равна

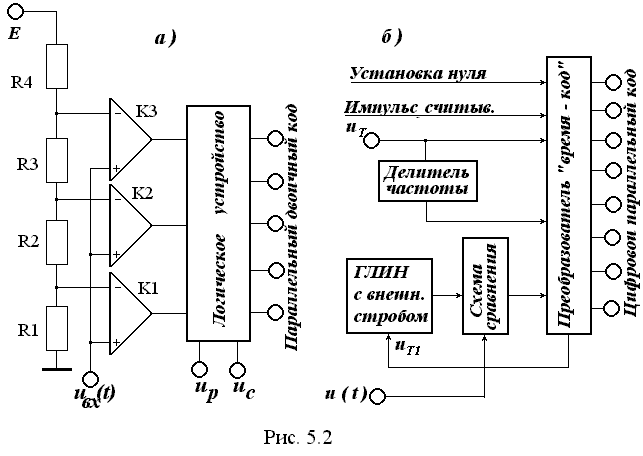
А) периоду дискретизации;

Б) шагу квантования;

В) 1/2 шага квантования;

Г) 1/4 периода дискретизации.

4. Ниже на рисунке представлена схема АЦП с промежуточным преобразованием



А) частоты во временной интервал;

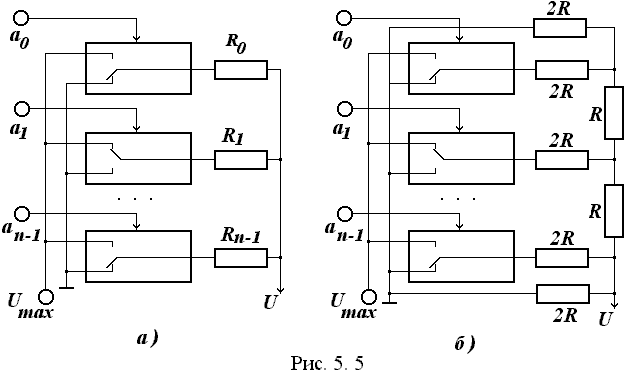
Б) тока во временной интервал;

В) пульсаций во временной интервал;

Г) напряжения во временной интервал.

5. Ниже на рисунке ({*аn-1an-2…a1a0*}- входное двоичное число) представлена

схема блока ЦАП с



А) вычитанием напряжений;

Б) дифференцированием напряжений;

В) умножением напряжений;

Г) суммированием напряжений.

**Тест №6**

к разделу 6

1. Для хранения отдельных кодовых слов обычно используются

А) оперативные запоминающие устройства;

Б) постоянные запоминающие устройства;

В) регистры;

Г) перепрограммируемые запоминающие устройства.

2. Байт равен

А) 12 битам;

Б) 4 битам;

В) 16 битам;

Г) 8 битам.

3. Количество информации, которое может храниться в ЗУ, определяет его

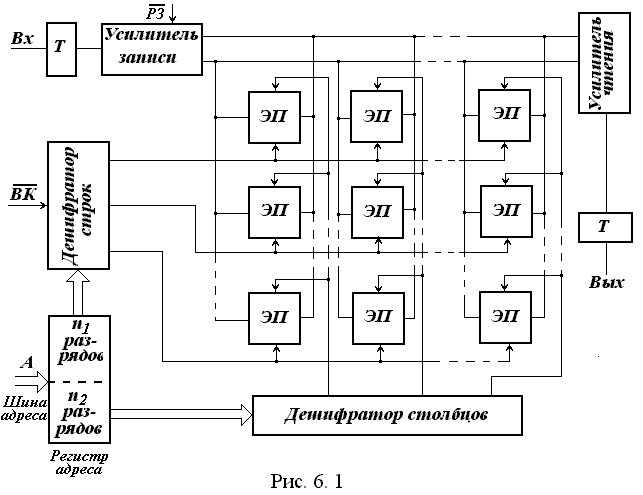
А) мощность;

Б) коммутативность;

В) емкость;

Г) адекватность.

4. Ниже на схеме изображена типовая структура микросхемы (ЭП-элемент памяти)



А) постоянного запоминающего устройства;  
Б) оперативного запоминающего устройства;  
В) перепрограммируемого запоминающего устройства;

Г) стековой памяти.

5. Главное отличие ПЗУ от ОЗУ заключается в том, что информация в ячейки памяти

А) записывается многократно и в процессе эксплуатации используется только режим чтения;

Б) записывается многократно и в процессе эксплуатации используется только режим записи;

В) записывается однократно и в процессе эксплуатации используется только режим чтения;

Г) записывается однократно и в процессе эксплуатации используется только режим перезаписи.

**Тест № 7**

к разделу 7

1. Микропроцессорные системы — это построенные на основе МП цифровые устройства и системы, предназначенные

А) для унификации данных, автоматизации и контроля;

Б) для обработки данных, автоматизации контроля и управления;

В) для оперативного отображения поступающих потоков данных;

Г) для построения наиболее быстродействующих цифровых устройств.

2. На схеме представлена архитектура МП системы на базе МПК КР580, в которой не показаны

С1

### 

Источник питания

Периферийные устройства

Средства ввода-вывода

Микропроцессор КР580ВМ80

Системная шина

Шина адреса А15-А0 (16 линий)

Шина данных D7-D0 (8 линий)

Шина управления (10 линий)

ПЗУ

(про-грамма)

ОЗУ

(данные)

Память

Порты ввода

Порты вывода

А) генератор тактовых импульсов;

Б) линии обратной связи;

В) периферийные блоки;

Г) сервисные системы;

3. На рисунке представлена структурная схема МП КР580ВМ80, в которой отсутствует

HOLD

DBIN

WAIT

READY

RESET

Устройство управления (УУ)

Дешифратор выбора регистров

WR

SYNC

INTE

HLDA

Внутренняя шина данных (8) линий)

Регистр-аккумулятор (А)

Счетчик команд

C2

C1

INT

.

Регистр W

Регистр Z

Мультиплексор

Буферный регистр адреса

Буферный регистр данных

Арифметико-логическое устройство (АЛУ)

Регистр признаков (F)

Схема десятичной коррекции

Указатель стека

Регистр B

Регистр D

Регистр C

Регистр E

Регистр H

Регистр L

А) регистр десятичной коррекции;

Б) регистр АЛУ;

В) регистр команд;

Г) регистр устройства управления.

4. Регистр-аккумулятор А предназначен для

А) временного хранения команд;

Б) временного хранения операнда или результатов арифметических и логических операций, выполняемых АЛУ;

В) приёма и хранения адресной части исполняемой команды;

Г) приёма и сохранения кода очередной команды, адрес ко­торой хранится в счетчике команд.

5. Входной сигнал запроса прерывания INT выполняет функции

А) управления шинами адреса и данных;  
Б) синхронизации;  
В) управления состоянием микропроцессора;

Г) запроса пе­риферийных устройств на обмен информацией.

**Тест № 8**

к разделу 8

1. БИС программируемого последовательного интерфейса КР580ВВ51

предназначена для

А) ор­ганизации работы МП систем в режиме реального времени;  
Б) организации высокоскоростного об­мена данными между памятью и внеш-ними устройствами;

В) организации обмена между МП и внешними уст­ройствами в последова-тельном коде;

Г) организации ввода-вывода инфор­мации различного формата и позволяет реализовать большинство протоколов обмена в параллельном коде.

2. БИС программируемого контроллера прямого доступа к памяти КР580ВТ57 предназначена для

А) ввода и вывода информации в микропроцессор;

Б) ор­ганизации работы МП систем в асинхронном режиме;

В) организации ввода-вывода инфор­мации различного формата в последова-тельном коде;

Г) организации высокоскоростного об­мена данными между памятью

и внешними устройствами.

3. БИС программируемого контроллера клавиатуры и дисплея КР580ВВ79 предназначена для

А) ввода и вывода информации в МП сис­темах;

Б) организации высокоскоростного об­мена данными между памятью и внеш-ними устройствами;

В) сокращения средств программного обеспечения и реальных затрат времени при выполнении прерываний в системах с приоритетами многих уровней;

Г) ор­ганизации работы МП систем в режиме реального времени.

4. БИС генератора тактовых импульсов КР580ГФ24 предназначена для

А) ор­ганизации работы МП систем в режиме реального времени;

Б) ввода и вывода информации в МП сис­темах;

В) сокращения средств программного обеспечения;

Г) обеспечения синхронизации МП системы.

5. БИС шинных формирователей КР580ВА86 и КР580ВА87(8-разрядные параллельные приемопередатчики) предназначены для

А) ускорения работы МП систем в режиме реального времени;

Б) реализации различных буферных схем в МП системах;

В) синхронизации МП систем;

Г) сокращения средств программного обеспечения.

**Тест № 9**

к разделу 9

1. В программную модель МП системы включаются только те регистры, к ко­торым можно обращаться

А) подключаясь гальванически;

Б) программно;

В) программно-аппаратным способом;

Г) только в последовательном коде.

2. Основным указателем памяти служит H-пара, которая адресуется в ко­мандах как РОН с адресом

А)110;

Б) 011;

В) 101;

Г) 100.

3. При прямой адресации в поле операнда (2-й и 3-й байты команды) содер-жится полный

А) 16-битный адрес байта памяти;

Б) 32-битный адрес байта памяти;

В) 8-битный адрес байта памяти;

Г) адрес регистра с адресом байта памяти.

4. При непосредственной адресации операндом является

А) 2-й байт самой команды;

Б) 1-й байт самой команды;

В) 1-й и 2-й байты самой команды;

Г) 3-й байт самой команды.

5. Команды с регистровой адресацией являются

А) 2-байтными;

Б) 1-байтными;

В) 3-байтными;

Г) 4-байтными.

**Тест № 10**

к разделу 10

1. Команды пересылок предписывают МП

А) форматирование данных;

Б) передачу данных из одного блока в другой;

В) считывание данных из одного блока;

Г) копирование данных.

2. Передача данных из памяти в регистр осуществляется командой

А) MVI R;   
Б) MOV R1, R2;   
В) MOV R, M;  
Г) MOV M, R.

3. Обмен данными между парами регистров H—L и D—E осуществляется ко-мандой

А) LHLD;

Б) XTHL;

В) XCHG;

Г) SHLD.

4. Сложение содержимого регистра и аккумуля­тора осуществляется командой

А) ADC M;

Б) ADD R;

В) ADC R;

Г) ADD M.

5. Циклический сдвиг со­держимого аккумуля­тора влево осуществляется ко-мандой

А) CMP M;

Б) RLC;

В) RRC;  
Г) CMP M.

**Тест № 11**

к разделу 11

1. На машинном языке все элементы программы (коды операций, адреса, дан-ные) представляются в

А) в16-ричной форме;

Б) двоично-десятичной форме;

В) 8-ричной форме;

Г) в двоичной форме.

2. Укажите достоинства программирования в машинных кодах:

А) легко модифицировать программу при введении пропущенных ко­манд;

Б) при вводе программы не требуется наличия трансляторов;

В) легко обнаружить ошибки и их исправить;

Г) облегчается отслеживание абсолютных адресов памяти.

3. Язык ассемблер — это язык

А) машинного кодирования;

Б) помехоустойчивого кодирования;

В) символического кодирования;

Г) эффективного кодирования.

4. Недопустимая метка

А) DONE:;

Б) 8ABC:;

В) C5FA:;

Г) M1:.

5. Поле комментария полностью игнориру­ется ассемблирующей программой и

А) начинается с точки;

Б) начинается с точки с запятой;

В) с запятой;

Г) с многоточия.

**Тест № 12**

к разделу 12

1. Память МК

А) состоит из разделенных памяти программ и памяти данных;

Б) организована в общем адресном пространстве;

В) состоит из ячеек оперативной памяти и регистров;

Г) программируется при изготовлении МК.

2. Ввод данных в МК

А) может осуществляться только в последовательном коде;

Б) производится через порты ввода/вывода;

В) выполняется периферийным адаптером;

Г) осуществляется только однобайтовыми словами в параллельном коде.

3. При использовании прямой адресации в МК

А) операнд прямо записывается в тело команды;

Б) адрес операнда размещается в теле команды;

В) адрес операнда записывается после выполнения операции в аккумулятор;

Г) операнд помещается прямо в регистр R0.

4. Команды пересылки МК КМ1816ВЕ48

А) выполняют пересылку операндов произвольной разрядности;

Б) могут осуществлять побайтовую пересылку операндов любой разрядности;

В) выполняются не менее чем за 2 машинных цикла;

Г) выполняют пересылку только 4- или 8-битных операндов.

5. Команды логических операций МК КМ1816ВЕ позволяют

А) выполнять операции И; ИЛИ над битами;

Б) выполнять операцию исключающее ИЛИ над битами;

В) выполнять операцию инверсии над битами;

Г) выполнять операцию И-НЕ над байтами.

**4.3.1. ПРАВИЛЬНЫЕ ОТВЕТЫ НА ТЕСТЫ РУБЕЖНОГО**

**КОНТРОЛЯ**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № теста | Раздел | Номера вопросов/Номера правильных ответов | | | | | |
| Номер  вопроса | 1 | 2 | 3 | 4 | 5 |
| 1 | 1 | *Правильный*  *ответ* | *Г* | *А* | *А* | *Б* | *Б* |
| 2 | 2 | *Правильный*  *ответ* | *А* | *Б* | *Б* | *В* | *Г* |
| 3 | 3 | *Правильный*  *ответ* | *В* | *А* | *В* | *А* | *В* |
| 4 | 4 | *Правильный*  *ответ* | *Б* | *В* | *В* | *Г* | *В* |
| 5 | 5 | *Правильный*  *ответ* | *Г* | *В* | *В* | *Г* | *Г* |
| 6 | 6 | *Правильный*  *ответ* | *В* | *Г* | *В* | *Б* | *В* |
| 7 | 7 | *Правильный*  *ответ* | *Б* | *А* | *В* | *Б* | *Г* |
| 8 | 8 | *Правильный*  *ответ* | *В* | *Г* | *А* | *Г* | *Б* |
| 9 | 9 | *Правильный*  *ответ* | *Б* | *А* | *А* | *А* | *Б* |
| 10 | 10 | *Правильный*  *ответ* | *Б* | *В* | *В* | *Б* | *Б* |
| 11 | 11 | *Правильный*  *ответ* | *Г* | *Б* | *В* | *Б* | *Б* |
| 12 | 12 | *Правильный*  *ответ* | *А* | *Б* | *Б* | *Г* | *В* |

**4.4. Итоговый контроль**

4.4.1. Билеты для зачета

### Билет 1

1. Логические функции и формы их представления. Понятие о сингулярных и бинарных функциях.
2. Цифровые фазовращатели. Особенности их синтеза.

Билет 2

1. Графический метод минимизации логических функций с помощью карт Карно-Вейча.
2. Цифроаналоговые преобразователи, их статические и динамические характеристики. Области применения.

### Билет 3

1. Основные логические элементы. Понятие о положительной и отрицатель-ной логике.
2. Принцип построения матрицы сопротивлений типа R-2R.

### Билет 4

1. Слабоопределенные (недоопределенные) логические функции. Методы их минимизации.
2. АЦП и их характеристики. Принцип схемотехнического решения.

### Билет 5

1. Мультиплексоры и демультиплексоры. Примеры использования, каскадирование.
2. Основные типы погрешностей в АЦП. Построение АЦП параллельного действия.

#### Билет 6

1. Коммутаторы аналоговых сигналов.
2. Синтез счетчиков с произвольным коэффициентом (модулем) счета.

#### Билет 7

1. Синтез цифровых схем сравнения и область их применения.
2. Кольцевые счетчики Джонсона. Их реализация на JK-триггерах.

#### Билет 8

1. Логический синтез схем полусумматора и полного сумматора.
2. Регистры сдвига. Их функциональные возможности и реализация.

#### Билет 9

1. Основные системы счисления, используемые в цифровых устройствах. Алгоритмы перевода чисел из одной системы счисления в другую.
2. Синхронные триггеры типа D. Работа в счетном режиме.

#### Билет 10

1. Сложение, вычитание и умножение двоичных чисел.
2. Универсальный триггер типа JK. Таблица переключений. Работа в счетном режиме.

#### Билет 11

1. Принцип деления двоичных чисел.
2. Мажоритарная функция типа «2 из 3-х» и синтез ее с помощью карты Карно в базисе И-ИЛИ-НЕ.

#### Билет 12

1. Преобразователи кодов. Преобразование двоично-десятичного кода в двоичный.
2. Преобразование аналогового сигнала в дискретный и цифровой. Понятие о дискретизации и квантовании сигнала.

Билет 13

1. Индикаторные устройства. Преобразование двоично-десятичного кода в код семисегментного индикатора.
2. Последовательностные устройства (конечные автоматы). Основные поня-тия.

#### Билет 14

1. Комбинационные устройства. Задачи синтеза.
2. Программируемые логические матрицы (ПЛМ). Синтез на ПЛМ переключательных функций.

#### Билет 15

1. Стандартные интегральные схемы. Интегральные схемы по ТТЛ, КМОП, ЭСЛ и И2Л-технологии. Сравнительный анализ серий микросхем.
2. Понятие о минтермах и макстермах при образовании совершенной дизъ-юнктивной (конъюнктивной) нормальной формы. Понятие о поиске тупи-ковой (минимальной) формы.

#### Билет 16

1. Логические элементы с открытым коллектором и с тремя состояниями выхода. Логические функции, реализуемые на этих элементах.
2. Алгебраический метод минимизации логических функций.

Билет 17

1. Наиболее распространенные системы счисления чисел. Виды двоично-десятичного кодирования: код 8-4-2-1, код Айкена, код «с избытком три», код «два из пяти», циклические коды Джонсона и Грея.
2. Синтез асинхронного RS-триггера с инверсными входами.

#### Билет 18

1. Формы представления чисел с фиксированной и плавающей точкой. Понятие об обратном и дополнительном коде числа.
2. Синтез асинхронного JK-триггера. Таблица переключений, работа в счет-ном режиме.

#### Билет 19

1. Основные теоремы, аксиомы и тождества алгебры логики, применяемые для упрощения логических выражений. Принцип двойственности, теорема Де Моргана.
2. Синхронные триггеры типа RSC. Таблица переключений.

#### Билет 20

1. Способы представления логических функций. Логические базисы для реализации функционально полной системы. Понятие об универсальных логических элементах.
2. Устройство неравнозначности (сумматор по модулю два) и его свойства.

4.4.2. Экзаменационные билеты

**Билет 1**

1. Структурная схема однокристального МП КР580ВМ80.
2. Особенности применения в ассемблере директив EQU и SET.

#### Билет 2

1. Трехшинная архитектура МП системы на базе МПК КР580.
2. Особенности применения в ассемблере директив условного ассембли-рования IF и ENDIF.

#### Билет 3

1. Особенности мультиплексной шины адреса/данных для МП К1810ВМ86.
2. Особенности применения в ассемблере группы директив определения. Формат директив DB, DW, DS.

#### Билет 4

1. Разрядно-модульные МП. Принципы организации микропрограммного управления на примере МП К1804ВМ1.
2. Особенности применения в ассемблере макрокоманд. Формат определения макрокоманды и обращения к макрокоманде.

**Билет 5**

1. Однокристальные микроЭВМ. Структурная схема. Методы и организация ввода/вывода.
2. Структуры данных для МПК КР580ВМ80 в виде одномерного и двумерного массива. Программное применение индексной адресации.

#### Билет 6

1. Основные параметры запоминающих устройств. Информационная емкость, разрядность, быстродействие, время хранения информации.
2. Структура данных для МПК КР580ВМ80 в виде очереди. Понятие длины очереди.

#### Билет 7

1. Оперативные запоминающие устройства (ОЗУ). Типовая структура ОЗУ матричного типа. Схема наращивания разрядности. ОЗУ динамического типа.
2. Принцип организации стековой памяти для МПК КР580. Принцип загрузки и извлечения данных из стека.

#### Билет 8

1. Постоянные запоминающие устройства (ПЗУ). Классификация ПЗУ по способу занесения информации.
2. Подпрограмма как законченный сегмент (модуль) программы. Порядок вызова и возврата в основную программу на примере МПК КР580. Вложенные подпрограммы.

#### Билет 9

1. Особенности ПЗУ, программируемого пользователем. Структурная схема.
2. Организация арифметических подпрограмм в МПК КР580. Форматы представления чисел в микропроцессорных системах.

#### Билет 10

1. Перепрограммируемые ПЗУ (ППЗУ). Достоинства и недостатки ППЗУ. Способы стирания информации.
2. Генератор тактовых импульсов КР580ГФ24. Структурная схема генератора. Подключение генератора к ЦПЭ КР580ВМ80. Временная привязка сигналов СИНХРО и СТРОБ СОСТОЯНИЯ.

#### Билет 11

1. Программная модель МП КР580ВМ80 (структурная схема с точки зрения программиста)
2. Основные сигналы управления МП КР580ВМ80. Понятие о сигналах H-активного и L-активного уровней.

#### Билет 12

1. Режимы адресации МПК КР580. Прямая адресация. Достоинства и недостатки. Привести примеры.
2. Системный контроллер КР580ВК28 (КР580ВК38). Структурная схема, назначение, принцип действия.

#### Билет 13

1. Классификация каналов связи в МП системе на базе МПК КР580. Параллельный и последовательный каналы. Принципы организации асинхронной и синхронной приемопередачи данных.
2. Режимы адресации МПК КР580. Непосредственная адресация. Достоинства и недостатки. Привести примеры.

#### Билет 14

1. Регистровая адресация в МПК КР580. Достоинства и недостатки. Привести примеры.
2. Интерфейс последовательного ввода/вывода. Назначение стартового и стопового битов при асинхронной передаче данных. Структурная схема последовательного интерфейса КР580ВВ51.

#### Билет 15

1. Косвенная адресация в МПК КР580. Возможность программного осуществления индексной адресации. Привести примеры.
2. Программируемый параллельный интерфейс ввода/вывода КР580ВВ55. Структурная схема и принцип действия.

#### Билет 16

1. Система команд МПК КР580. Разделение системы команд на группы по функциональному признаку.
2. Программируемый таймер КР580ВИ53. Назначение, структурная схема, режим работы и способ программирования.

#### Билет 17

1. Особенности программирования на языке ассемблер МПК КР580. Поля ассемблерной строки. Стандартный формат бланка для ассемблерных программ.
2. Программируемый контроллер прерываний КР580ВН59. Структурная схема и принцип действия.

#### Билет 18

1. Программирование на машинном языке. Формат бланка при программировании на машинном языке МПК КР580.
2. Программируемый контроллер прямого доступа к памяти КР580ВТ57. Структурная схема, временная диаграмма и способ программирования.

#### Билет 19

1. Понятие о мнемонике как о ключевом слове ассемблера. Поле операнда. Адреса памяти как операнды. Команды, имеющие пустое поле операнда.
2. Особенности микропроцессора Z80. Технические характеристики. Способы адресации. Работа центрального процессора по выполнению команд.

#### Билет 20

1. Однокристальная микроЭВМ (микроконтроллер) КМ1816. Архитектура. Организация памяти. Логика условных переходов, счетчик команд, организация прерываний.
2. Организация работы ЦПЭ КР580ВМ80. Машинные циклы и такты. Слово состояния.

**Содерж а н и е**

**1. Информация о дисциплине………………………………….…….……..3**

1. 1.1. Предисловие ……………………………………………………..…….…3
2. 1.2. Содержание дисциплины и виды учебной работы………….…..……...4

**2. Рабочие учебные материалы………………………………………….…6**

2.1. Рабочая программа …………..………………………….…………….…6

2.2. Тематический план дисциплины ……….……………….………......… 14

2.3. Структурно-логическая схема дисциплины ………….…………….. 22

2.4. Временной график изучения дисциплины………….……...…….….… 24

2.5. Практический блок …………………...…………………….……..….… 25

2.5.1. Практические занятия……………………………………….…..…….…25

2.5.2. Лабораторные работы……………………………………….…...………30

2.6. Балльно-рейтинговая система оценки знаний …..………………….…. 34

**3. Информационные ресурсы дисциплины……………….……………36**

3.1. Библиографический список…………………………………………..…36

3.2. Опорный конспект………………….………………………….……......37

3.3. Глоссарий…………………………………..…………..………..…....….49

3.4. Технические и программные средства обеспечения дисциплины……57

3.5. Методические указания к проведению практических занятий…….....58

**4. Блок контроля освоения дисциплины…………………..….…………136**

4.1. Задания на контрольную работу и методические указания к их выпол-нению ………………….……………………………………………………..…136

4.1.1. Задания на контрольную работу………………………….…………136

4.1.2. Методические указания к выполнению контрольной работы……..139

4.2. Задание на курсовой проект и методические указания к его выполнению …………………………………………………………………...……………...141

4.2.1. Задание на курсовой проект…………………………………………..141

4.2.2. Методические указания к выполнению курсового проекта…..…….142

4.3. Текущий контроль…………………………………………….…………144

4.3.1. Правильные ответы на тесты рубежного контроля…………………162

4.4. Итоговый контроль ………………………………………….…….……..163

4.4.1. Билеты для зачета……………………………………….………….….163

4.4.2. Экзаменационные билеты……………………………………….…....165

**Соколов Олег Леонидович**

**Голод Олег Саулович**

Цифровые устройства и микропроцессоры

Учебно-методический комплекс

Редактор И. Н. Садчикова

Сводный темплан 2010 г.

ЛР № 020308 от 14.02.97

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подписано в печать Формат 60х84 1/1

Б.кн. – журн. П.л. Бл. Изд-во СЗТУ

Тираж Заказ

Северо-Западный государственный заочный технический университет,

Издательство СЗТУ, член Издательско-полиграфической ассоциации

университетов России

191186, Санкт-Петербург, ул. Миллионная, д.5