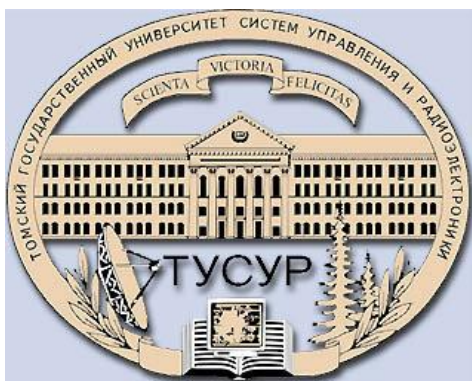


**Министерство образования и науки РФ  
Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
Томский государственный Университет Систем Управления и Радиоэлектроники  
(ТУСУР)**

УТВЕРЖДАЮ  
Проректор по учебной работе  
Л.А. Боков  
“ \_\_\_ ” \_\_\_\_\_ 2012 г.



В. М. Винокуров

**ЛАБОРАТОРНЫЙ ПРАКТИКУМ**

**по дисциплине «Сети связи и системы коммутации»**

По дисциплинам направлений подготовки «Телекоммуникации» специальностей 201100 (210405) «Радиосвязь, радиовещание и телевидение» и 071700 (210401) «Физика и техника оптической связи», реализуемых в рамках данного направления подготовки дипломированного специалиста

Учебное методическое пособие

Факультет радиотехнический

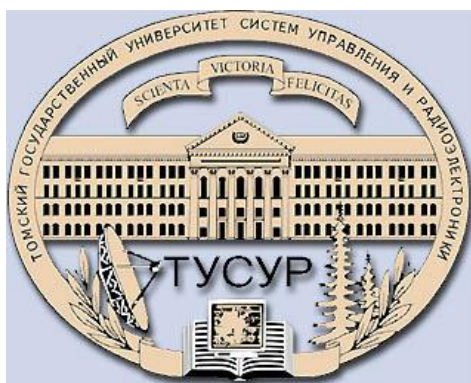
Обеспечивающая кафедра «Телекоммуникаций и основ радиотехники»

## СОДЕРЖАНИЕ

1. Изучение принципов построения сетей ISDN.....	3
2. Маршрутизация в телекоммуникационных сетях.....	17
3. Изучение языка общения "Человек - Машина" (MML).....	32
4. Технология синхронной цифровой иерархии SDH.....	42
5. Сжатие информации в СПД с использованием кода Хаффмена.....	57

**Министерство образования и науки РФ  
Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
Томский государственный Университет Систем Управления и Радиоэлектроники  
(ТУСУР)**

УТВЕРЖДАЮ  
Заведующий кафедрой ТОР, доцент  
\_\_\_\_\_ Е.П.Ворошилин  
<< \_\_\_\_ >> \_\_\_\_\_ 2012 г.



Кафедра Телекоммуникаций и Основ Радиотехники  
(ТОР)

**Методическое пособие по проведению компьютерных лабораторных работ по ISDN**

РАЗРАБОТАЛИ

Профессор кафедры ТОР \_\_\_\_\_ В. М. Винокуров

\_\_\_\_\_ Д. В. Фролов

“ \_\_\_\_ ” \_\_\_\_\_ 2012

Томск 2012

Программа разработана в ходе дипломного проектирования в Томском Государственном Университете Систем Управления и Радиоэлектроники (ТУСУРе) на кафедре Телекоммуникаций и Основ Радиотехники (ТОР) в 2005 г студентом гр.140-2 Фроловым Д.В. под руководством доцента Винокурова В.М. Программа переработана и модернизирована при участии студента гр. 586<sub>1</sub> Кангарова С. А. Работа прошла регистрацию в ОФАП Госкоорцентра Минобрнауки России и опубликована в журнале «Компьютерные учебные программы и инновации» – М: ГОСКООРЦЕНТР. – 2008. - N 7. стр. 92, свидетельство об отраслевой регистрации разработки №10437 от 16.04.2008; (Windows 2000/XP, Linux via Wine, алгоритмический язык DELPHI 7.0).

## Оглавление

1	Задачи и содержание лабораторного практикума.....	4
2	Методические указания по выполнению лабораторных работ..	6
2.1	Лабораторная работа №1 – Общие вопросы.....	6
2.1.1	Тест.....	7
2.1.2	Задачи.....	7
2.1.3	Тест – рисунки.....	7
2.1.4	Создание схемы абонентской установки.....	8
2.2	Лабораторная работа №2 – Абонентская установка.....	9
2.2.1	Составление кода АМІ (ЧПИ).....	10
2.2.2	Составление кадра интерфейса S, физического уровня.....	10
2.2.3	Составление кадра интерфейса S, канального уровня.....	12
2.2.4	Создание функциональной схемы сетевого окончания (NT).....	12
2.2.5	Составление кода 2В1Q.....	14
2.2.6	Составление кадра интерфейса U, определить размеры полей..	14
2.3	Лабораторная работа №3 – сигнализация DSS1.....	15
2.3.1	Активация интерфейса S.....	16
2.3.2	Активация интерфейса U.....	17
2.3.3	Установление соединения абонент – абонент.....	19
2.3.4	Деактивация интерфейса S.....	20
2.3.5	Расчет GoS.....	22
2.3.6	Расчет сигнализационной нагрузки на сети SS7.....	22
3	Контрольные вопросы.....	24
4	Литература.....	25

## 1 Задачи и содержание лабораторного практикума

Лабораторный практикум разработан с целью применения в учебном процессе при изучении темы «Цифровые сети с интегрированным обслуживанием (ISDN)», может быть использован в обучающих и контролируемых целях и дает углубленное понимание технологии ISDN.

Алгоритм взаимодействия модулей программы приведен на рисунке 1.1.

## 2 Методические указания по выполнению лабораторных работ.

Окно-заставка лабораторного практикума содержит панель меню с кнопками «Файл», «Лабораторная работа №» и «Помощь». Пользуясь кнопкой «Файл», преподаватель через пароль может производить коррекцию списка заданий. Под этой же кнопкой возможен просмотр результатов работы и выход из программы.

**Практикум включает в себя три лабораторных работы.** Выбор номера работы осуществляется кнопкой «Лабораторная работа №».

В центре окна-заставки расположена кнопка запуска практикума «ISDN» и приведены фамилии разработчиков.

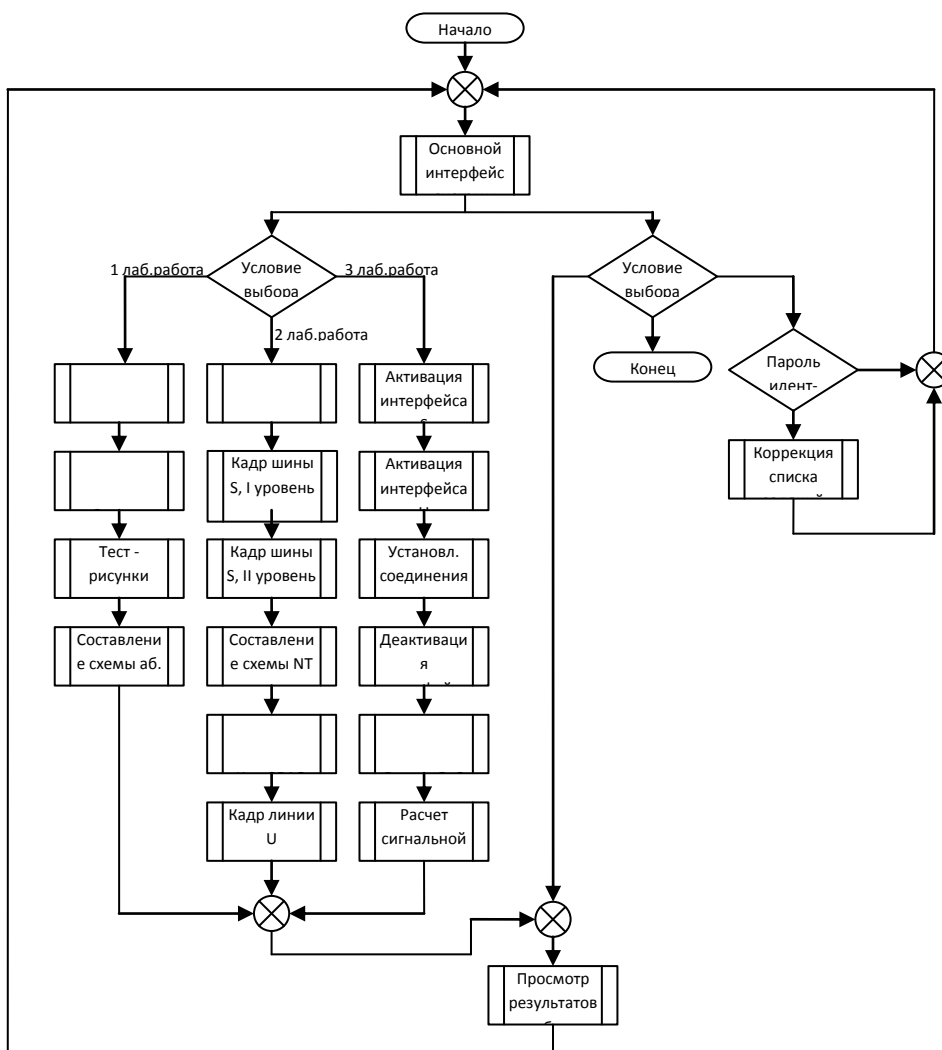


Рисунок 1.1 - Алгоритм взаимодействия модулей программы

### **Запуск лабораторной работы:**

Лабораторный практикум запускается нажатием кнопки «ISDN» окна-заставки. Рабочее поле программы содержит карту города с нанесённой на неё произвольно составленной схемой телефонной сети (линии красного цвета). Активные абоненты условно показаны значком телефонного аппарата (синий цвет).

В выпадающем меню «Лабораторная работа №» выбирается номер необходимой лабораторной работы. Переход к дальнейшим действиям задания осуществляется на этом этапе нажатием кнопки любого активного абонента.

Для запуска файла помощи в выпадающем меню выбирается необходимый вид помощи и запускается нажатием левой клавиши мышки. Файл помощи может быть вызван и во время выполнения задания (при помощи мышки выбирается пункт «Меню», а потом пункт «Помощь»).

## **2.1 Лабораторная работа №1 – Общие вопросы построения ISDN**

### **Цель работы:**

Проверить и закрепить общие понятия о технологии ISDN.

### **Содержание работы:**

Лабораторная работа №1 содержит четыре части:

1. **Тест** – необходимо ответить на вопросы, поиск ответов на которые углубляет знания, полученные на лекциях или практических занятиях.
2. **Задачи** – необходимо решить задачи и выбрать правильный ответ из предлагаемых ответов.
3. **Тест – рисунки** – на рисунках недостает одного элемента, из предложенных ответов необходимо выбрать недостающий элемент.
4. **Создание схемы абонентской установки** – необходимо по заданию собрать схему установки в помещении абонента.

### **2.1.1 Задание «Тест»**

#### **Ход работы:**

1. Ознакомиться с теоретическими сведениями по технологии ISDN.
2. Прочитать вопрос теста в верхней части окна и выбрать правильный ответ из числа предложенных. Нажать кнопку «Проверить», при этом в верхней части окна, на месте задания, появится текущая оценка результата «Правильно - Неправильно».
3. Для выполнения задания необходимо правильно ответить на 10 вопросов из 15 предложенных.

### **2.1.2 Задание «Задачи»**

#### **Ход работы:**

1. Прочитать текст задачи в верхней части окна.
2. Решить задачу и выбрать правильный ответ из числа предлагаемых ответов. Нажать кнопку «Проверить», при этом в верхней части окна, на месте задания, появится текущая оценка результата «Правильно - Неправильно».
3. Для выполнения задания необходимо правильно решить 8 задач из 10 предложенных. Формулы для решения задач выводятся путем логического размышления и правильного понимания текста задачи.

### **2.1.3 Задание «Тест – рисунки»**

#### **Ход работы:**

1. Прочитать текст задания в верхней части окна и выбрать правильный ответ из числа предлагаемых. Нажать кнопку «Проверить», при этом в верхней части окна, на месте задания, появится текущая оценка результата «Правильно - Неправильно».

2. Для выполнения задания необходимо правильно решить 4 задачи из 5 предложенных.

#### **2.1.4 Задание «Создание схемы абонентской установки»**

Окно программы «Составление схемы» содержит горизонтальную панель меню, два горизонтальных окна: «Задание» и «Действие», вертикальное окно «Элементы схемы», рабочее поле для составления схемы и кнопку «Проверить». Кнопкой «Меню» вызываются действия: «Переход», «Помощь» и «Выход». Действие «Переход» позволяет, используя пароль, произвольно перемещаться по заданиям «Практикума». Назначение действий «Помощь» и «Выход» стандартно.

##### **Ход работы:**

1. Ознакомиться с теоретическими сведениями по технологии ISDN, изучить возможные схемы абонентской установки.

2. Прочитать текст задания в верхней части окна, где указано число и тип терминалов, применяемых в заданной установке.

3. Составить схему абонентской установки. Схему необходимо смонтировать из её элементов, задаваемых с помощью 12 кнопок, расположенных в вертикальном окне справа от рабочего поля программы и снабженных названиями: ISDN-телефон, ISDN-факс, NTBA, и т. д. Постановка элемента на рабочее поле осуществляется путем нажатия на нужную кнопку с последующим щелчком левой клавишей мышки в зоне рабочего поля. Перемещение объекта осуществляется его «перетаскиванием» после однократного нажатия на левую клавишу мышки.

Необходимо заметить, что нажатие правой клавиши мыши на объекте в зоне рабочего поля вызывает меню действий «Удалить», «Соединить», «Разъединить».

После выставления необходимых объектов на рабочее поле, необходимо произвести соединение объектов. Соединение объектов осуществляется нажатием правой клавиши мыши на объекте и выбора соответствующей операции «Соединить», после чего необходимо указать второй объект для соединения левой клавишей мышки.

Линия соединения на рабочем поле имитирует двужильный провод в реальных условиях.

Если была поставлена лишняя связь между двумя объектами, ее можно удалить, щелкнув правой клавишей мышки на одном из объектов и выбрав операцию «Разъединить», после чего необходимо указать второй объект.

Необходимо учитывать что четырехпроводная S-шина состоит из двух двупроводных линий связи и при соединении объектов с шиной S следует предусматривать для них две связи, причем на разные двухпроводные линии.

4. После того как схема собрана, следует нажать кнопку «Проверить», после чего в окне действий выдается результат выполнения.

## 2.2 Лабораторная работа №2 – Изучение абонентской установки

### Цель работы:

Закрепить знания о структуре кадров физического уровня ЭМВОС и кодах АМІ и 2В1Q, системе сигнализации DSS1 и функциональном назначении блоков сетевого окончания NT.

### Содержание работы:

- **Представление сигнала кодом АМІ (ЧПИ)** – необходимо согласно заданию составить форму сигнала на физическом уровне ЭМВОС в интерфейсе S.
- **Составление кадра физического уровня интерфейса S** – необходимо составить кадр интерфейса S в виде битов кадра, в зависимости от направления передачи и в последующем, в зависимости от значения битов, составить вид сигнала в линии при помощи кода АМІ.
- **Составление кадра канального уровня интерфейса S** – необходимо составить кадр интерфейса S на втором уровне ЭМВОС.
- **Создание функциональной схемы сетевого окончания (NT)** – для более глубокого понимания процессов происходящих при согласовании шины S и линии U, необходимо составить функциональную схему сетевого окончания.
- **Представление сигнала кодом 2В1Q** – необходимо по заданию составить вид сигнала на физическом уровне ЭМВОС в интерфейсе U
- **Составление кадра интерфейса U, определение размеров полей** – необходимо прописать размеры полей кадра интерфейса U.

### 2.2.1 Задание «Представление сигнала кодом АМІ»

#### Ход работы:

1. Ознакомиться с кодовой таблицей кода АМІ (ЧПИ) - кода с чередующейся полярностью импульсов.
2. Задание содержится в верхней части окна в виде десятичного числа, которое необходимо закодировать
3. Привести десятичное число к двоичному виду.
4. В поле сигнала необходимо составить вид электрического сигнала, при помощи кнопок: «-1», «0», «+1», которые соответствуют уровням сигнала в реальной линии связи.

*Примечание* Возможно исправление формы сигнала при его неправильном построении. Для этого необходимо передвинуть указатель (прямоугольник красного цвета) нажатием левой клавиши мышки на необходимом участке и выставить правильный уровень.

5. Для проверки результата нажать на кнопку «Проверить». Оценка проделанной работы индицируется в верхней части окна, на месте задания. Правильное построение формы сигнала необходимо для перехода к следующему заданию.

### 2.2.2 Задание «Составление кадра физического уровня интерфейса S».

#### Ход работы:

1. Ознакомиться с форматом кадра интерфейса S на физическом уровне ЭМВОС.
2. Внимательно ознакомиться с текстом задания, приведенном в верхней части окна.
3. Составить структуру кадра, вставляя в поля его формата необходимые биты каналов и служебные биты при помощи соответствующих кнопок, расположенных в нижней части окна. При нажатии на кнопки в поле кадра на месте указателя вставляется соответствующий бит. Ошибочные биты в кадре корректируются в окне указателя, перемещаемого по кадру с помощью кнопок «Вправо» и «Влево», расположенными в



нижней части окна. Результат выполнения задания проверяется нажатием кнопки «Проверить».

4. Далее необходимо по полученному формату составить форму сигнала в линии, с использованием кода АМІ согласно методике пункта 2.2.1. Исправление формы сигнала при его неправильном построении также возможно. Для этого необходимо передвинуть указатель при помощи кнопок «Вправо» («Влево») или нажатием левой клавиши мыши на необходимый участок, после чего выставить правильный уровень сигнала при помощи кнопок: «-1», «0», «+1». Результат выполнения задания проверяется нажатием кнопки «Проверить». Правильное построение сигнала требуется для перехода к следующему заданию.

### **2.2.3 Задание «Составление формата кадра интерфейса S канального уровня».**

#### **Ход работы:**

1. Ознакомиться с форматом кадра интерфейса S на канальном уровне ЭМВОС.

2. Внимательно ознакомиться с текстом задания, расположенном в верхней части окна.

3. Составить структуру кадра, вставляя в поля его формата необходимые биты каналов и служебные биты при помощи соответствующих кнопок, расположенных в правой части окна. Кадр расположен вертикально, байты кадра расположены горизонтально. Биты в поле кадра на месте указателя вставляются нажатием на соответствующие кнопки. Исправление формы сигнала при его неправильном построении также возможно. Для этого необходимо передвинуть указатель при помощи кнопок «Вправо» («Влево») или нажатием левой клавиши мыши на необходимый участок, после чего выставить правильный уровень сигнала при помощи кнопок: «-1», «0», «+1». Результат выполнения задания проверяется нажатием кнопки «Проверить».

### **2.2.4 Задание «Создание функциональной схемы сетевого окончания (NT)».**

#### **Ход работы:**

1. Ознакомиться с устройством и функциональными блоками сетевого окончания NT (Network Terminal).

2. Ознакомиться с текстом задания, расположенным в верхней части окна.

3. На начальном этапе имеются три объекта (шина S, линия U, источник переменного тока), они помечены серым цветом. Над этими объектами нельзя выполнять никаких действий. Они являются моделями разъемов NT: разъема для подключения шины S, разъема для подключения линии U, разъема для подключения питания от сети 220 В соответственно.

6. Функциональная схема строится при помощи функциональных блоков, расположенных в правой части окна. Постановка блока на рабочее поле осуществляется при помощи нажатия левой клавишей мыши на кнопку соответствующего функционального блока и последующего нажатия в необходимом месте рабочего поля.

7. При нажатии правой клавиши мыши на любом функциональном блоке, появляется выпадающее меню действий.

4. После вынесения всех необходимых блоков на рабочее поле их необходимо соединить линиями связи. Существует два типа линий связи: однонаправленные – сигнал может проходить только в одну сторону, двунаправленные – сигнал может проходить в обе стороны.

8. Соединение линиями связи осуществляется в следующей последовательности: вначале необходимо вызвать выпадающее меню действий нажатием правой клавиши мыши на первом объекте, от которого требуется проложить линию связи. Затем выбирается пункт «Соединить однонаправленной линией» или «Соединить

двухнаправленной линией», в зависимости от необходимости. Для завершения соединения нажать левую кнопку мыши на втором соединяемом блоке.

5. После соединений, необходимо настроить каждый функциональный блок. Для этого вызывается выпадающее меню действий и выбирается пункт «Свойства». В свойствах объекта указываются функции, характерные для данного функционального блока.

6. После построения и настройки, для проверки собранной схемы необходимо нажать на кнопку «Проверить», результаты выполнения задания отражаются в нижней части экрана в полосе действий.

### **2.2.5 Задание «Составление кода 2B1Q».**

#### **Ход работы:**

1. Ознакомиться с теорией кода 2B1Q.
2. Прочитать текст задания в верхней части окна, где указано десятичное число, которое необходимо закодировать.
3. Привести десятичное число к двоичному виду.
4. В поле сигнала необходимо составить график формы электрического сигнала, при помощи кнопок: «-3», «-1», «+1», «+3», которые соответствуют уровням сигнала в реальной линии связи. Для помощи в нижней части окна расположена кодовая таблица кода 2B1Q (соответствие пары двухуровневых бит одному четырехуровневому биту).

При неправильном построении графика возможно исправление его формы. Для этого необходимо передвинуть указатель при помощи кнопок «Вправо» («Влево») или нажатием левой клавишей мыши на необходимый участок с последующей установкой необходимого уровня сигнала при помощи кнопок: «-3», «-1», «+1», «+3».

5. Результат выполнения задания проверяется нажатием кнопки «Проверить». Правильное построение сигнала требуется для перехода к следующему заданию.

### **2.2.6 Задание «Указать размеры полей формата кадра сигнала в интерфейсной точке U».**

#### **Ход работы:**

1. Ознакомиться с теоретическими сведениями по структуре кадров интерфейса U.
2. Прочитать текст задания в верхней части окна.
3. Указать размеры полей формата кадра сигнала в интерфейсной точке U, записывая необходимые цифры в соответствующем окне рисунка.
4. После расстановки размеров полей кадра нажмите кнопку «Проверить». Результаты задания отражаются в нижней части окна «Правильно - Неправильно».

### 2.3 Лабораторная работа №3 – Сигнализация DSS1

#### Цель работы:

Проверить и закрепить знания об абонентской сигнализации DSS1, провести расчет качества обслуживания и расчет сигнализационной нагрузки на сети SS7.

#### Содержание работы:

- **Активация интерфейса S.** –В соответствии с заданием необходимо произвести активацию интерфейса S (последовательный обмен сигналами).
- **Активация интерфейса U.** - В соответствии с заданием необходимо произвести активацию интерфейса U (последовательный обмен сигналами).
- **Установление соединения абонент – абонент.** – Производя последовательный обмен соответствующими сигналами, выполнить простейшую процедуру установления и разрушение соединения «абонент - абонент».
- **Деактивация интерфейса S.** – Производя последовательный обмен соответствующими сигналами, произвести процедуру деактивации интерфейса S.
- **Расчет GoS.** –Используя график "Зависимость вероятности потерь от изменения емкости пучка каналов при заданном профиле нагрузки " вычислить величину GoS (Grade of Service – качество обслуживания). График построен по методике, описанной в работе В.А.Ершова и Н.А.Кузнецова «Мультисервисные телекоммуникационные сети», изданной МГТУ им.Баумана в 2003г (432 с).
- **Расчет сигнализационной нагрузки на сети SS7.** –По заданным требованиям необходимо вычислить сигнализационную нагрузку, создаваемую информационными каналами на систему сигнализации.

#### 2.3.1 Задание «Активация интерфейса S».

##### Описание окна задания «Активация интерфейса S»

Пример окна приведен на рисунке 2.1. Здесь приняты следующие обозначения:

- 1 – текст задания,
- 2 – схема участка цепи,
- 3 – сигналы в шине S,
- 4 – выпадающее меню сигналов,
- 5 – кнопка «Передать»,
- 6 – окно действий.

#### Ход работы:

1. Ознакомиться с теоретическими сведениями по DSS1 (активация/деактивация интерфейсов, установление соединения).
2. Прочитать текст задания в верхней части окна.
3. В соответствии с заданием и предшествующей последовательностью сигналов, выбрать в выпадающем меню сигнал, который должен быть послан к NT на текущем шаге активации интерфейса.
4. Для передачи сигнала в линию в сторону NT, необходимо нажать кнопку «Передать».
5. Проследить последовательность передачи сигналов и при необходимости повторить пункты 3-4. Для успешного выполнения задания «Активация интерфейса S» необходимо в соответствии с заданием произвести соответствующую серию последовательных передач сигналов. Переход к следующему заданию производится автоматически.

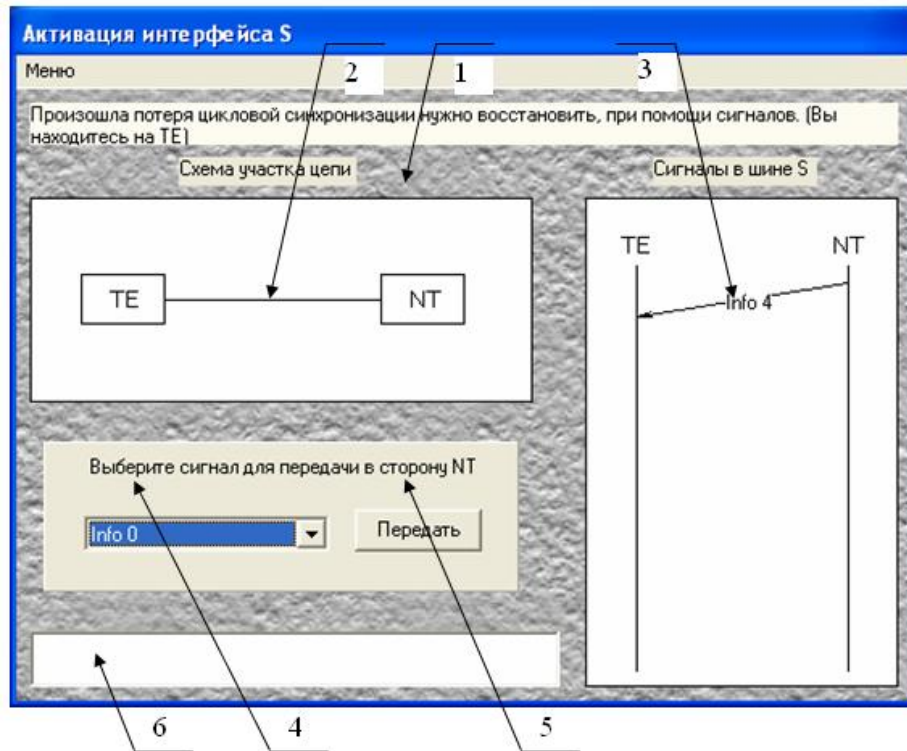


Рисунок 2.1 – Окно задания «Активация интерфейса S»

### 2.3.2 Задание «Активация интерфейса U».

#### Описание окна задания «Активация интерфейса U»

Пример окна приведен на рисунке 2.2. Приняты следующие обозначения:

- 1 – текст задания,
- 2 – схема участка цепи,
- 3 – сигналы в шине S и линии U,
- 4 – выпадающее меню сигналов,
- 5 – кнопка «Передать»,
- 6 – окно действий.

#### Ход работы:

1. Ознакомиться с теоретическими сведениями по DSS1 (активация/деактивация интерфейсов, установление соединения.)
2. Прочитать текст задания в верхней части окна.
3. В соответствии с заданием и предшествующей последовательностью сигналов, выбрать в выпадающем меню сигнал, который должен быть послан от NT к LT на текущем шаге активации интерфейса U.
4. Для передачи сигнала в линию в сторону LT, необходимо нажать кнопку «Передать».

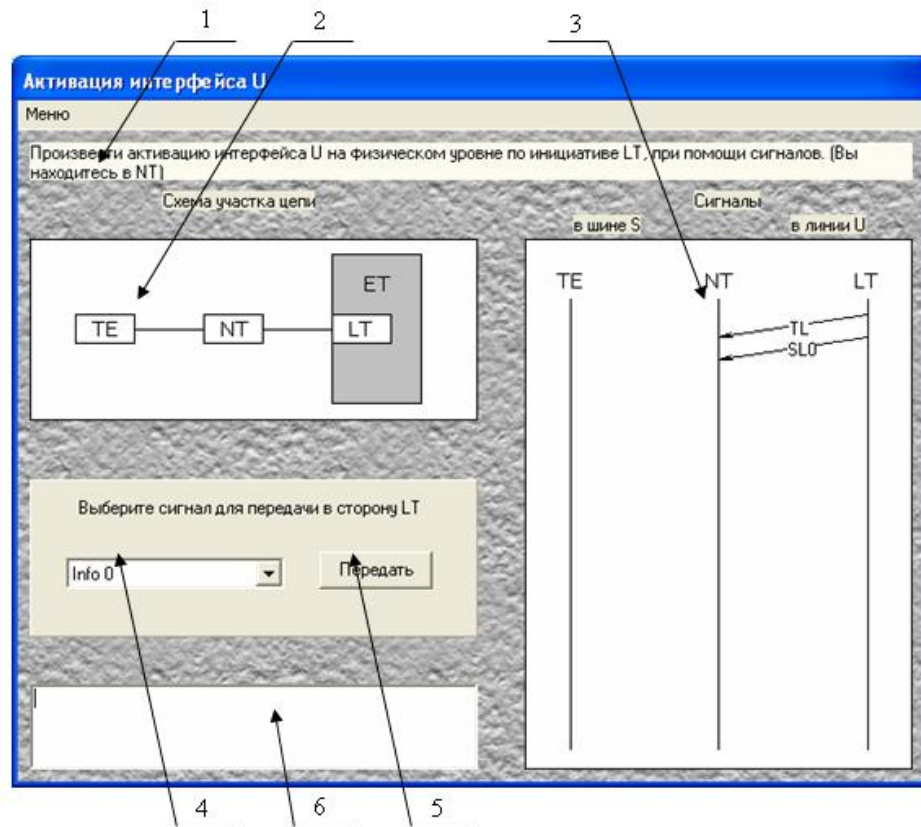


Рисунок 2.2 – Окно задания «Активация интерфейса U»

5. Проследить последовательность передачи сигналов и при необходимости повторить пункты 3-4. Для успешного выполнения задания «Активация интерфейса U», необходимо в соответствии с заданием произвести соответствующую серию последовательных передач сигналов. Переход к следующему заданию производится автоматически.

### 2.3.3 Задание «Установление соединения «абонент - абонент».

#### Описание окна задания « Установление соединения «абонент - абонент»

Ввиду большей трудности подробной процедуры установления соединения будем использовать простейшую процедуру установления соединения, которая также хорошо отражает главные моменты обмена служебными сигналами при установлении соединения. В данном задании требуется произвести простейшую процедуру установления соединения «абонент - абонент» при помощи последовательного обмена сигналами, передать информацию и произвести процедуру разъединения. Окно задания «Установление соединения» представлено на рисунке 2.3.

Пример окна приведен на рисунке 2.3. Приняты следующие обозначения:

- 1 – текст задания,
- 2 – сигналы в абонентских и межстанционных линиях,
- 3 – кнопка «Передать»,
- 4 – выпадающее меню сигналов,
- 5 – статус процесса
- 6 – окно действий.

#### Ход работы:

1. Ознакомиться с теоретическими сведениями по DSS1 (активация/деактивация интерфейсов, установление соединения).
2. Прочитать текст задания в верхней части окна.

3. В соответствии с заданием и предшествующей последовательностью сигналов, выбрать в выпадающем меню сигнал, который должен быть послан от вызывающей стороны к коммутатору А на текущем шаге установления соединения.

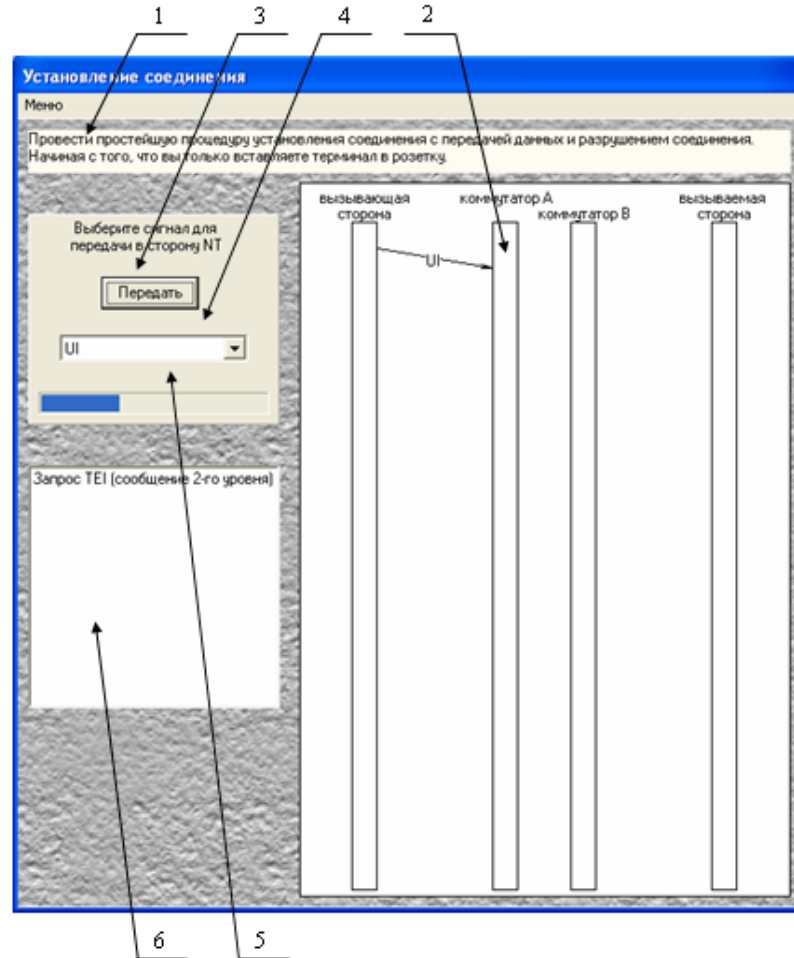


Рисунок 2.3 – Окно задания «Установка соединения»

4. Для передачи сигнала в линию от вызывающей стороны к коммутатору А, необходимо нажать кнопку «Передать».

5. Проследить последовательность передачи сигналов и при необходимости повторить пункты 3-4. Для успешного выполнения задания «Установка соединения», необходимо в соответствии с заданием произвести соответствующую серию последовательных передач сигналов. Переход к следующему заданию производится автоматически.

#### 2.3.4 Задание «Деактивация интерфейса S».

##### Описание окна задания «Деактивация интерфейса S»

Пример окна приведен на рисунке 2.4. Здесь приняты следующие обозначения: 1 – текст задания, 2 – схема участка цепи, 3 – сигналы в шине S, 4 – выпадающее меню сигналов, 5 – кнопка «Передать», 6 – окно действий.

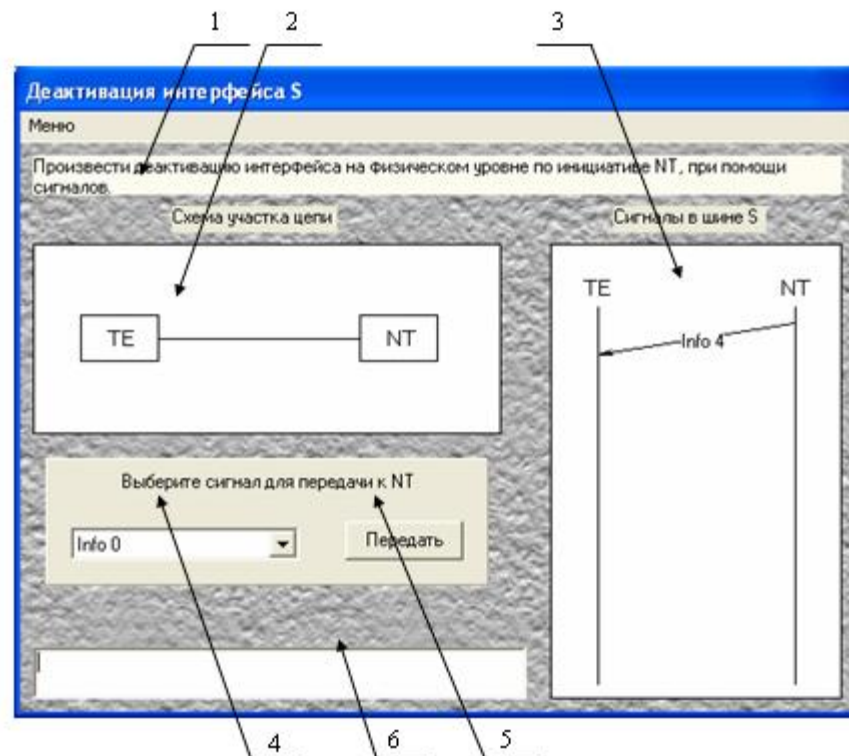


Рисунок 2.4 – Окно задания «Деактивация интерфейса S»

#### Ход работы:

1. Ознакомиться с теоретическими сведениями по DSS1 (активация/деактивация интерфейсов, установление соединения).
2. Прочитать текст задания в верхней части окна.
3. В соответствии с заданием и предшествующей последовательностью сигналов, выбрать в выпадающем меню сигнал, который должен быть послан к NT на текущем шаге деактивации интерфейса.
4. Для передачи сигнала в линию в сторону NT, необходимо нажать кнопку «Передать».
5. Проследить последовательность передачи сигналов и при необходимости повторить пункты 3-4. Для успешного выполнения задания «Деактивация интерфейса S», необходимо в соответствии с заданием произвести соответствующую серию последовательных передач сигналов. Переход к следующему заданию производится автоматически.

#### 2.3.5 Задание «Расчет GoS».

##### Ход работы:

1. Ознакомиться с теоретическими сведениями по оценке качества обслуживания.
2. Прочитать текст задания в верхней части окна.
3. По графику в соответствии с требованиями задания вычислить вероятность потерь.
4. По графику в соответствии с требованиями задания и найденной в п.3 вероятностью потерь, вычислить необходимое количество каналов.
5. Вычисленные значения вероятности потерь и количества каналов занести в соответствующие поля, находящиеся в правой части окна.
6. Для проверки результата необходимо нажать кнопку «Проверить».

### 2.3.6 Задание «Расчет сигнализационной нагрузки на сети SS7».

Технология ISDN является очень хорошим дополнением для получения широкого спектра услуг, но в виду большей распространенности на телефонных сетях системы сигнализации ОКС-7, технологии должны взаимодействовать между собой. Поэтому, установление соединения между абонентами ISDN, происходит через ОКС-7, соответственно ISDN вносит свою лепту в нагрузку на сети ОКС-7.

#### Ход работы:

1. Ознакомиться с теоретическими сведениями и методикой расчета сигнализационной нагрузки на сети SS7.
2. Прочитать текст задания в верхней части окна.
3. В соответствии с данными для расчета и методикой вычисления, произвести расчет сигнальной нагрузки на сети SS7.
4. Результат расчета занести в соответствующее поле  $A_{link}$ , находящееся в нижней части окна.
5. Для проверки результата необходимо нажать кнопку «Проверить».

### 3 Контрольные вопросы:

1. Чему равна частота дискретизации при аналого-цифровом преобразовании в ISDN?
2. Каков основной режим передачи информационных сигналов в ISDN?
3. Какова скорость передачи в цифровом телефонном канале в РФ?
4. Какое общее число логических каналов в BRI?
5. Какое оборудование подключается к интерфейсу S/T?
6. Может ли BRI состоять из каналов 1B+D?
7. Сколько бит в кадре первого уровня в интерфейсе S?
8. Сколько бит В-каналов в кадре интерфейса S?
9. Какова скорость передачи кадров в интерфейсе S?
10. Какая скорость передачи по каналу В?
11. Какая информация передается по каналу В?
12. Какова максимальная длина шины S при конфигурации «от точки к точке»?
13. Какова максимальная длина шины S при конфигурации «короткая пассивная шина»?
14. Каково максимальное количество устройств на шине S при BRA?
15. Каково максимально возможное число одновременно работающих устройств в ISDN при доступе на базовой скорости?
16. Как называется процедура объединения каналов?
17. Какая максимальная длина шины U допускается при использовании провода с диаметром медной жилы 0,4 мм.?



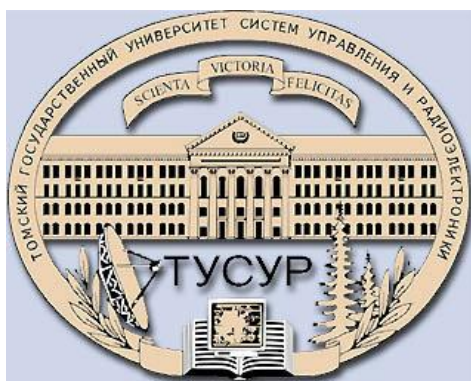
**Министерство образования и науки РФ**  
**Федеральное государственное бюджетное образовательное учреждение**  
**высшего профессионального образования**  
**Томский государственный Университет Систем Управления и Радиоэлектроники**  
**(ТУСУР)**  
**(ТОР)**

УТВЕРЖДАЮ

Заведующий кафедрой ТОР, доцент

\_\_\_\_\_ Е.П.Ворошилин

<< \_\_ >> \_\_\_\_\_ 2012 г.



**Маршрутизация в телекоммуникационных сетях**

Лабораторная работа по дисциплине «Сети связи и системы коммуникации»

РАЗРАБОТАЛ

Профессор кафедры ТОР

\_\_\_\_\_ В. М. Винокуров

“ \_\_ ” \_\_\_\_\_ 2012

Программа разработана в ходе дипломного проектирования в Томском Государственном Университете Систем Управления и Радиоэлектроники (ТУСУРе) на кафедре Телекоммуникаций и Основ Радиотехники (ТОР) в 2005 г студентом гр.140-2 Кургузовым В.А под руководством доцента Винокурова В.М. Программа переработана и модернизирована при участии студента гр. 525<sub>3</sub> Парфёнова Ю. Работа прошла регистрацию в ОФАП Госкоорцентра Минобрнауки России и опубликована в журнале «Компьютерные учебные программы и инновации» – М: ГОСКООРЦЕНТР. – 2008. - N 7. стр. 89, свидетельство об отраслевой регистрации разработки №10440 от 16.04.2008; (Windows 2000/XP, Linux, алгоритмический язык C++).

### 1 Цель работы.

Знакомство с методами маршрутизации сообщений на сетях связи.

### 2 Введение

Для передачи информации в сети связи от одного узла к другому может использоваться один из нескольких путей, проходящих через различные узлы. При этом порядок выбора таких путей определяется так называемым **планом распределения информации (ПРИ)**. Планом распределения информации некоторого узла сети связи называется заданная очередность выбора исходящих направлений из этого узла ко всем остальным узлам сети связи. Если задан план распределения информации для каждого узла, то говорят, что задан план распределения информации для всей сети связи.

Порядок выбора исходящих направлений (ветвей из  $i$ -го узла) ко всем остальным узлам сети, т. е. план распределения информации для узла, можно представить **матрицей маршрутов** для  $i$ -го узла вида:

$$M_i = \begin{matrix} & UK_1 & UK_2 & \dots & UK_N \\ \beta_{i,i_1} & \left\| \begin{matrix} m_{i_1,1} & m_{i_1,2} & \dots & m_{i_1,N} \\ \beta_{i,i_2} & \left\| \begin{matrix} m_{i_2,1} & m_{i_2,2} & \dots & m_{i_2,N} \\ \vdots & \vdots & & \vdots \\ \beta_{i,i_n} & \left\| \begin{matrix} m_{i_n,1} & m_{i_n,2} & \dots & m_{i_n,N} \end{matrix} \right. \end{matrix} \right. \end{matrix} \right. \end{matrix} \end{matrix}.$$

В матрице маршрутов число столбцов равно  $(N-1)$ , где  $N$ — число узлов на сети (столбец в матрице  $M_i$  для узла  $i$  не отводится), а число строк — числу  $n$  узлов, смежных с рассматриваемым узлом. Элемент  $m_{j,r}$  указывает номер очередности выбора ветви  $\beta_j$ , при передаче информации к узлу  $r$ , т. е.  $m_{j,r} \in \{1, 2, \dots, n\}$ .

Так как при длительных перегрузках и повреждениях отдельных ветвей (пучков, каналов) план распределения информации может оказаться не оптимальным, т. е. не обеспечивающим реализацию максимальной пропускной способности сети, то для его оптимизации необходимо перераспределить порядок выбора путей, т. е. осуществить коррекцию плана распределения информации.

Методы коррекции плана распределения информации называются динамическими

методами управления телетрафиком. Система управления, обеспечивающая динамическое управление телетрафиком (т. е. реализующая динамический метод управления телетрафиком), называется **системой динамического управления**.

Все существующие методы динамического управления телетрафиком, т. е. методы коррекции матрицы маршрутов, можно разделить на две группы: **детерминированные** и **статистические**. В свою очередь, и детерминированные, и статистические методы делятся на **разовые** и **групповые**.

**Разовые детерминированные методы** позволяют получать матрицу маршрутов только для одной заявки, тогда как групповые — для группы заявок. При этом матрицы маршрутов вычисляются применительно к ситуации на сети, сложившейся на данный момент времени, без учета предшествующих ситуаций. **Статистические методы** позволяют корректировать матрицы маршрутов на основе статистики о возможной длине пути в том или ином направлении, полученной в результате обслуживания предыдущих заявок. При этом разовые статистические методы позволяют корректировать матрицу маршрутов после обслуживания каждой заявки, а групповые — после обслуживания нескольких заявок.

В данной работе план распределения информации формируется на основе метода рельефов. Кроме того, используется алгоритм Дейкстры, положенный в основу популярного протокола маршрутизации OSPF.

### 3 Теоретическая часть

#### 3.1 Метод рельефов

Метод рельефов предусматривает формирование плана распределения информации на сети по числу транзитных узлов коммутации. Возможно использование и других параметров.

Суть данного метода состоит в следующем. Пусть  $K$ —произвольный узел коммутации сети связи. **К-рельефом** называется процедура присвоения значения числовой функции («высоты» в рельефе) каждому тракту передачи сообщения. К-рельеф строится следующим образом. Из  $K$ -го узла коммутации по всем исходящим трактам передается число 1. Все узлы коммутации, в которые поступило число 1, передают по всем исходящим трактам передачи сообщения, кроме тех трактов, по которым поступила эта единица, число 2. Далее узлы коммутации, на которые поступило число 2, передают по исходящему тракту передачи сообщения, кроме тех трактов, по которым поступила 2, число 3 и т.д., до тех пор, пока все тракты передачи сообщения не будут пронумерованы. Говорят, что тракт передачи сообщения имеет **высоту  $n$** , если он обозначен числом  $n$  в К-рельефе.

Указанным способом формируется рельеф из каждого узла коммутации сети связи.

Тракт передачи сообщения с минимальной высотой, называется исходящим трактом **первого выбора**. Тракты передачи сообщений с большими высотами, соответственно являются исходящими трактами второго, третьего и т.д. выбора.

#### Пример 3.1.

Построим рельеф на сети относительно узла коммутации  $A$  (рис. 3.1).

Узел коммутации  $A$  по исходящему тракту передачи сообщения  $AB$ ,  $AC$ ,  $AD$  передает число 1 и присваивает им это значение. Узлы  $B$ ,  $C$  и  $D$  передают по трактам  $BG$ ,  $BC$ ,  $CI$ ,  $CK$ ,  $CD$  и  $DK$  в узлы  $G$ ,  $I$ ,  $H$  и  $K$  число 2. В свою очередь, узлы  $G$ ,  $I$ ,  $H$  и  $K$  передают по тракту передачи сообщения  $GL$ ,  $GI$ ,  $IL$ ,  $IM$ ,  $IH$ ,  $HK$  и  $KO$  число 3. Следовательно, перечисленным трактам присваивается число 3. узел коммутации  $L$ ,  $M$ ,  $H$ ,  $O$ , в свою очередь, передают по тракту передачи сообщения  $LM$ ,  $MN$ ,  $HM$ ,  $HO$  и  $ON$  число 4 и этим трактам присваивается число 4. Таким образом, на сети строится  $A$  - рельеф (рис. 3.2).

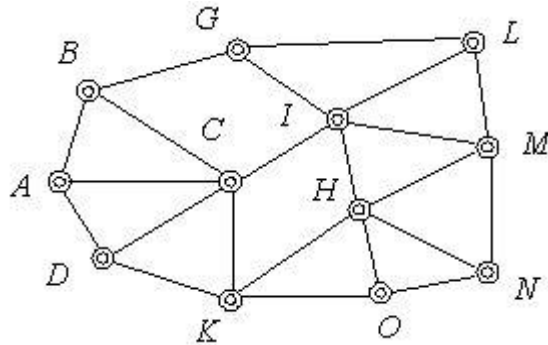


Рис. 3.1 - Формирование ПРИ методом рельефов

Чтобы найти кратчайший маршрут от произвольного узла коммутации к узлу *A* достаточно в каждом узле коммутации выбирать исходящий тракт передачи сообщения с меньшим весом. Например, кратчайший маршрут от узла коммутации *N* до узла коммутации *A* будет следующий:  $\mu^1_{N,A} = \{NO, OK, KD, DA\}$  или  $\mu^2_{N,A} = \{NM, MI, IC, CA\}$ ,  $\mu^3_{N,A} = \{NO, OK, KC, CA\}$ .

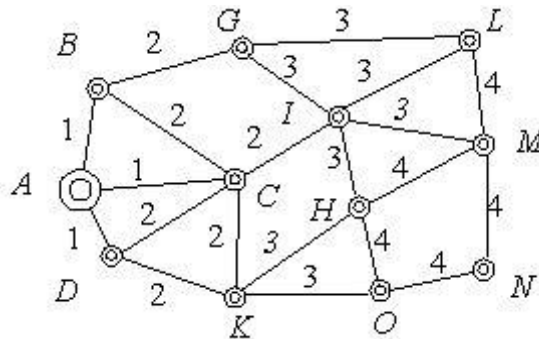


Рисунок 3.2 - Формирование A - рельефа

В случае загрузки или неисправности элементов сети, а также в случае ввода в эксплуатацию новых узлов необходимо переформировывать рельеф. При восстановлении или освобождении элементов сети восстанавливается первоначальный рельеф.

Метод рельефов позволяет определить не только исходящие тракты передачи сообщений первого выбора, но и второго, третьего и так далее. Недостатком данного метода является необходимость передавать информацию о формировании рельефов между всеми узлами, что значительно загружает сеть связи.

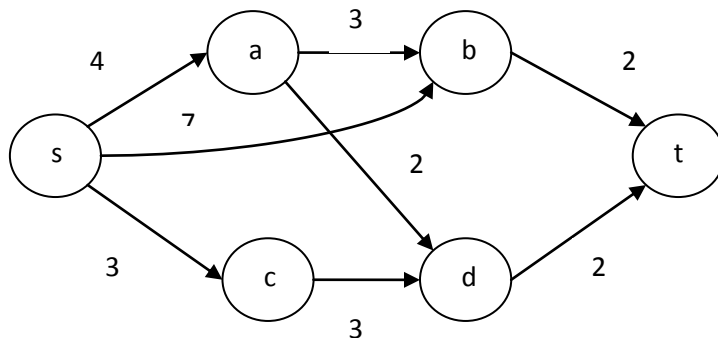
**Пример 3.2**

Дано: схема сети рис.3.3 с исходящим узлом "s" и входящим "t".

Требуется найти кратчайший путь из "s" в "t" с помощью алгоритма Дейкстры.

**Решение**

Шаг 1. Окрашена только вершина "s". Метрика исходящего узла равна нулю  $d(s)=0$ ,



а для всех вершин "x", не совпадающих с "s", метрики равны  $d(x)=\infty$ .

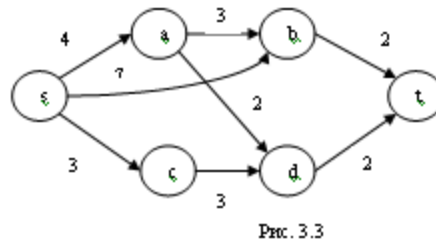


Рис. 3.3

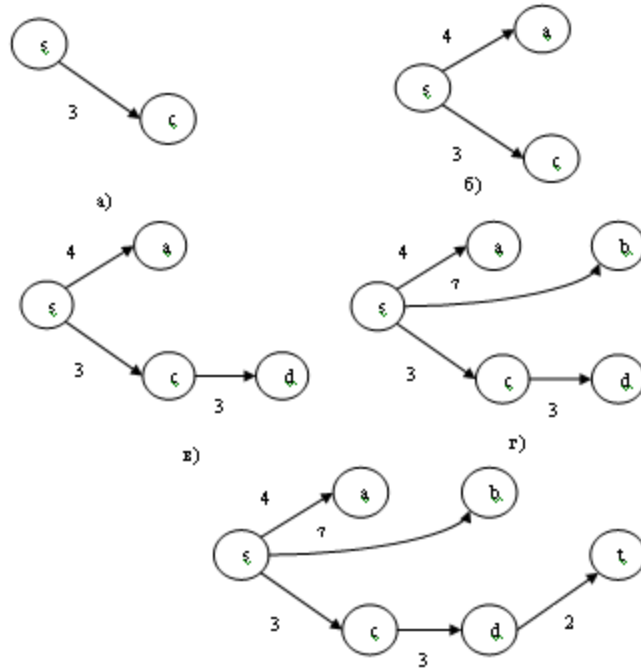


Рис. 3.4- Растущее ориентированное дерево кратчайших расстояний

Шаг 2 ( $y = s$ ).

$$d(a) := \min(d(a), d(s) + a(s, a)) = \min(\infty, 0 + 4) = 4$$

$$d(b) := \min(d(b), d(s) + a(s, b)) = \min(\infty, 0 + 7) = 7$$

$$d(c) := \min(d(c), d(s) + a(s, c)) = \min(\infty, 0 + 3) = 3$$

$$d(d) := \min(d(d), d(s) + a(s, d)) = \min(\infty, 0 + \infty) = \infty$$

$$d(t) := \min(d(t), d(s) + a(s, t)) = \min(\infty, 0 + \infty) = \infty$$

Поскольку величина  $d(c)=3$  является минимальной из величин  $d(a)$ ,  $d(b)$ ,  $d(c)$ ,  $d(d)$ , и  $d(t)$ , то вершина 'с' окрашивается. Так же окрашивается и дуга (s,c) (рис.3.4а).

Шаг 3. Поскольку вершина "t" остается неокрашенной, осуществляется переход к шагу 2.

Шаг 2 ( $y = c$ )

Поскольку величина  $d(a)=4$  является минимальной из величин  $d(a)$ ,  $d(b)$ ,  $d(d)$ , и  $d(t)$ ,

$$d(a) := \min(d(a), d(c) + a(c, a)) = \min(4, 3 + \infty) = 4$$

$$d(b) := \min(d(b), d(c) + a(c, b)) = \min(7, 3 + \infty) = 7$$

$$d(d) := \min(d(d), d(c) + a(c, d)) = \min(\infty, 3 + 3) = 6$$

$$d(t) := \min(d(t), d(c) + a(c, t)) = \min(\infty, 3 + \infty) = \infty$$

то вершина 'а' окрашивается. Так же окрашивается и дуга (s,a), которая определяет величину  $d(a)$ . Текущее дерево кратчайших путей теперь состоит из дуг (s,c) и (s,a) (рис.3.4б).

*Шаг 3.* Поскольку вершина "t" остается неокрашенной, осуществляется переход к шагу 2.

*Шаг 2* ( $y = a$ )

Поскольку величина  $d(d)=4$  является минимальной из величин  $d(b)$ ,  $d(d)$ , и  $d(t)$ , то

$$d(b) := \min(d(b), d(a) + a(a, b)) = \min(7, 4 + 3) = 7$$

$$d(d) := \min(d(d), d(a) + a(a, d)) = \min(6, 4 + 2) = 6$$

$$d(t) := \min(d(t), d(a) + a(a, t)) = \min(\infty, 4 + \infty) = \infty$$

вершина "d" окрашивается. Можно считать, что величину  $d(d)$  определяют как дуга (c,d), так и дуга (a,d). Поэтому можно окрасить любую из этих дуг. Окрасим, например дугу (c,d). Текущее дерево кратчайших путей теперь состоит из дуг (s,c), (s,a) и (s,d) (рис.3.4в).

*Шаг 3.* Поскольку вершина "t" остается неокрашенной, осуществляется переход к шагу 2.

*Шаг 2* ( $y = d$ )

$$d(b) := \min(d(b), d(d) + a(d, b)) = \min(7, 6 + \infty) = 7$$

$$d(t) := \min(d(t), d(d) + a(d, t)) = \min(\infty, 6 + 2) = 8$$

Поскольку величина  $d(b)=7 = \min(d(b), d(t))$  меньше величины  $d(t)$ , то вершина "b" окрашивается. Так же окрашивается и дуга (s, b), которая определяет величину  $d(b)$ . Текущее дерево кратчайших путей теперь состоит из дуг (s,c), (s,a),(c,d) и (s,b) (рис.3.4г).

*Шаг 3.* Поскольку вершина t остается неокрашенной, осуществляется переход к шагу 2.

*Шаг 2* ( $y = b$ )

$$d(t) := \min(d(t), d(b) + a(b, t)) = \min(8, 7 + 2) = 8$$

Итак, вершина "t"наконец-то окрашивается. Вместе с нею окрашивается дуга (d,t), определяющая величину  $d(t)$ . Окончательно построенное дерево кратчайших путей состоит из дуг (s,c), (s,a), (c,d), (s,b) и (d,t) (рис.3.4д).

Кратчайший путь, соединяющий вершину "s" с вершиной "t", состоит из дуг (s,c), (c,d) и (d,t) и имеет длину  $3+3+2=8$ . Это не единственный кратчайший путь между вершинами "s" и "t".

#### 4 Описание лабораторного макета

Для того чтобы начать выполнять лабораторную работу, запустите файл **control.exe**. Кроме этого файла в текущем каталоге находятся файл Help.hlp, который можно использовать как электронный учебник по данной технологии, так как он запускается независимо от исполняемого файла (рис. 4.1).

После запуска исполняемого файла (control.exe) появится первое окно, в котором предлагается ознакомиться с принципами маршрутизации (то есть, прочитать электронный учебник), а затем пройти входной тест. Процесс обучения можно пропустить – для этого выберите «Начать тест без обучения» (рис.4.2).

Входной тест – это 10 простых вопросов по методам маршрутизации (рис. 4.3). При помощи клавиши «Далее» Вы можете прокручивать вопросы; кнопка «Выход» - выход из программы. Для того чтобы пройти входной тест и приступить к выполнению лабораторной работы, Вам необходимо ответить на любых 8 вопросов за неограниченное время. Выбор ответа на вопрос осуществляется при помощи указателя мыши – слева от вариантов ответа изображены квадраты, которые необходимо активизировать (рис. 4.3).

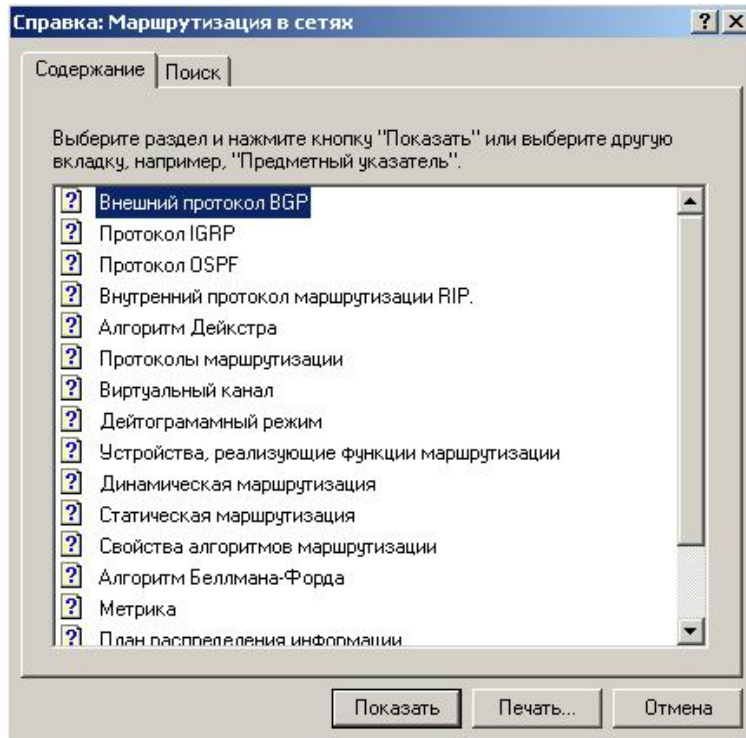


Рисунок 4.1 - Внешний вид окна приложения Help.hlp (файл помощи)

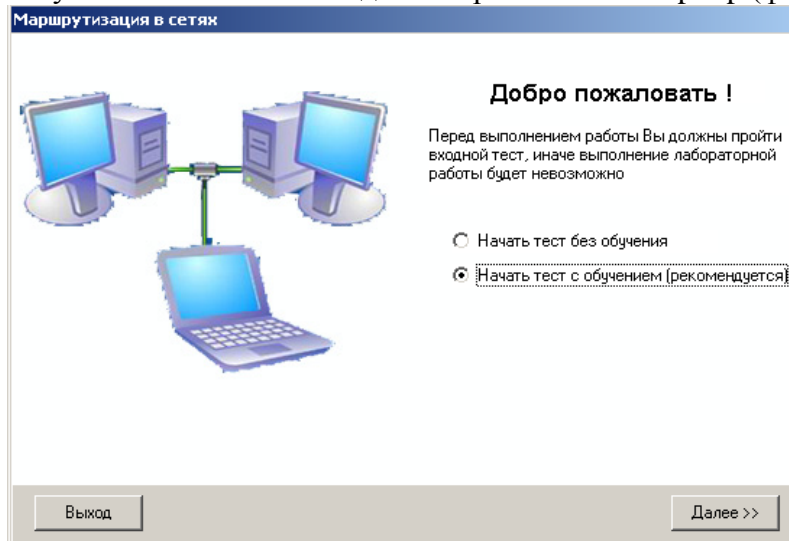


Рисунок 4.2 Внешний вид первого окна

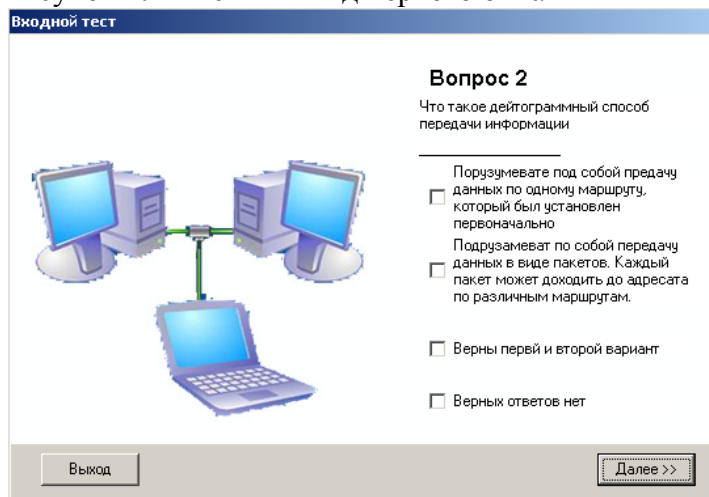


Рисунок 4.3 - Внешний вид окна с входным тестом

После прохождения входного теста Вам будет предложено выбрать один из шести вариантов задания (рис 4.4).

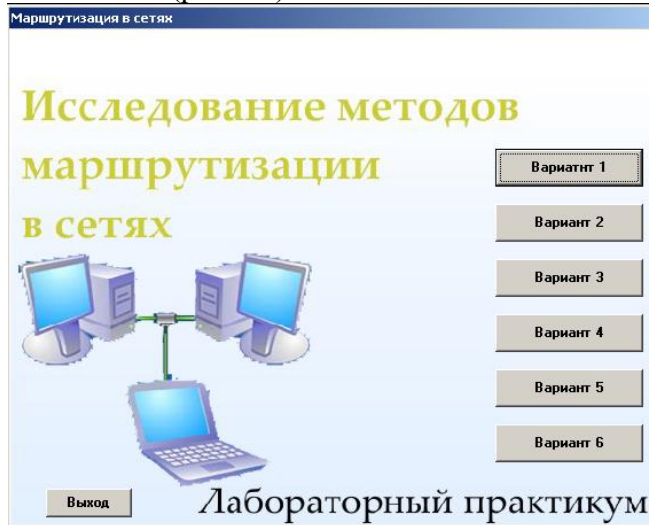


Рис. 4.4 - Выбор варианта

Затем появится основное окно компьютерной лабораторной работы (рис. 4.5).

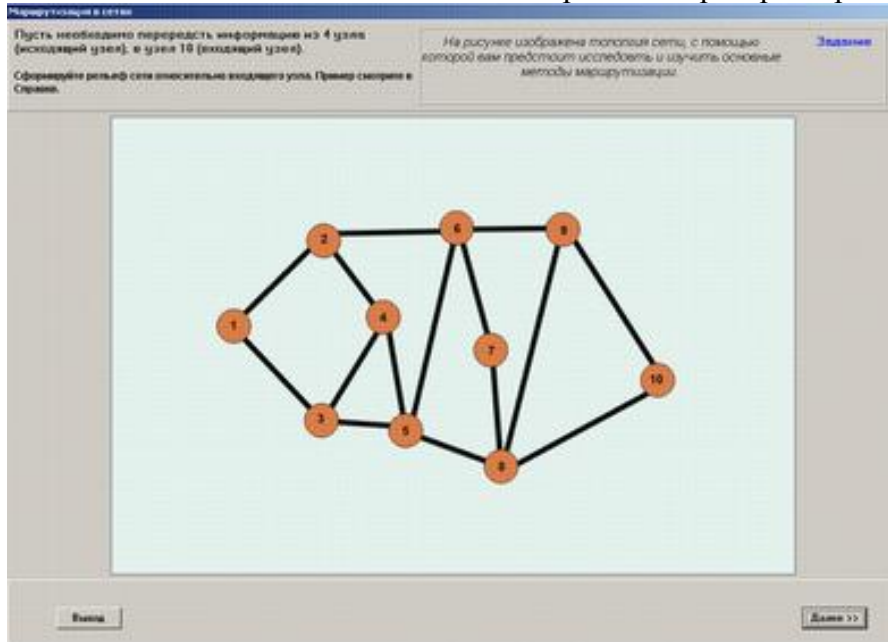


Рисунок 4.5 – Внешний вид окна задания

## 5 Порядок выполнения лабораторной работы

**Процедура 1.** Ответить на вопросы входного теста

**Процедура 2.** Выбрать вариант согласно заданию. Появится окно, в котором изображена топология сети, на примере которой Вам предстоит исследовать метод рельефов. В верхнем левом углу приведено общее задание на лабораторную работу.

*Выполнять задание нужно пошагово.*

Шаг 1. Скопировать схему сети в отчет или черновик отчёта и сформировать на ней рельеф относительно входящего узла.

Шаг 2. В окне “Задание 1” указать размерность матрицы рельефа исходящего узла

Шаг 3. В окне “Задание 2” заполнить заголовки матрицы. Заголовки столбцов заполняются номерами узлов сети, за исключением номера исходящего узла. Заголовки строк заполняются номерами ребер графа сети, примыкающих к исходящему узлу, в



порядке возрастания их номеров. (Например, при номере исходящего узла 3, номера 3-1, 3-4, 3-10 для каждого из трех ребер соединяющих узел 3 с узлами 1, 4, 10).

Шаг 4. В окне “Задание 3” заполнить клетки столбца матрицы рельефов, соответствующего входящему узлу, высотами ребер графа, примыкающих к исходящему узлу.

Шаг 5. Используя результат шага 4, определить ближайший к исходящему промежуточный узел в направлении первого кратчайшего пути к входящему узлу. В качестве пути первого выбора в сети выбрать из матрицы ребро с минимальной высотой.

Шаг 6. Узел, вычисленный в шаге 5, назвать исходящим. Повторять процедуру шага 5 до прохождения маршрута по всей сети вплоть до заданного входящего узла.

Шаг 7. В отчете отметить выбранный маршрут и рельеф сети относительно входящего узла.

**Процедура 3.** Осуществить выбор кратчайшего пути в выбранной схеме сети с помощью матричного метода (алгоритма Дейкстры), используя произвольно выбранную метрику для каждого ребра графа. Результат отобразить в отчете.

**Процедура 4.** Осуществить выбор кратчайшего пути в выбранной схеме сети с помощью матричного метода (алгоритма Дейкстры), используя одинаковую и равную 1 для каждого ребра графа метрику. Результат отобразить в отчете.

**Процедура 5.** Сравнить результаты **процедуры 4** и шага **7 процедуры 2**. Сделать выводы.

### 5 Пример выполнения задания

Пример иллюстрирует выполнение **Этапов 2 и 4** лабораторной работы, сопровождаемое соответствующими комментариями.

**Процедура 1.** Предположим, входной контроль пройден успешно.

**Процедура 2.** Пусть выбраны схема и задание, соответствующие рис. 5.1. Задание гласит: «Пусть необходимо передать информацию из узла 4 (исходящего узла), в узел 10 (входящий узел). Сформируйте рельеф сети относительно входящего узла. Пример смотрите в «Справке»».

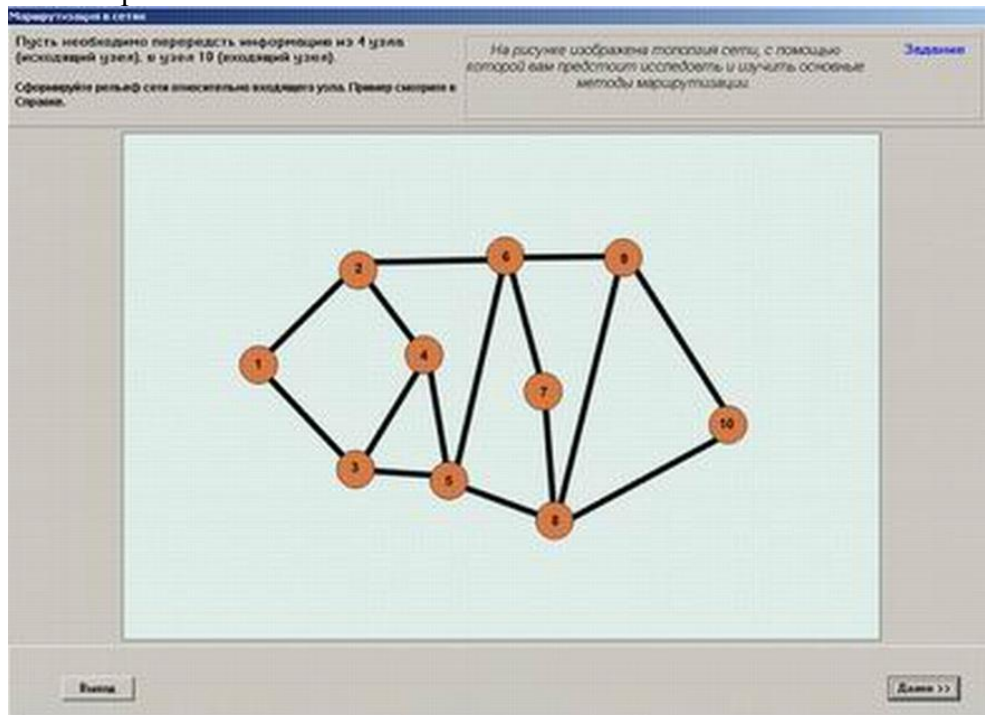


Рисунок 5.1 – Вид окна задания.

Сформируем рельеф относительно входящего узла. Способ формирования рельефа описан выше, в пункте 3.1. Результат представлен на рис. 5.2.

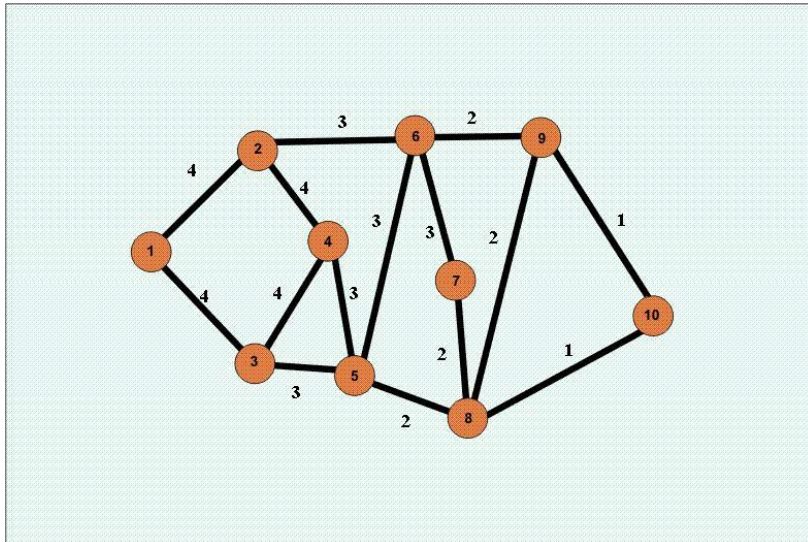


Рисунок 5.2 – Рельеф сети.

После формирования рельефа нажатием кнопки «Далее» переходим к «Заданию 2».

## Задание 2

**Шаг 1.** Укажем размерность матрицы исходящего узла. Исходящий узел 4, имеет три примыкающих тракта передачи сообщения (4-2, 4-3, 4-5) и девять узлов сети могут получить от него информацию. Запишем цифры 3 и 9 в соответствующие окна (рис. 5.3). После нажатия кнопки «Далее» программа даёт оценку правильности выполнения шагового задания и выводит на экран таблицу, которую нужно будет использовать в задании 2.

Маршрутизация в сетях

Пусть необходимо перередать информацию из 4 узла (исходящий узел), в узел 10 (входящий узел).

Укажите размерность матрицы.

**Задание 1**

$M_4 =$    $\times$

Для продолжения нажмите "Далее>>"

Для вызова справки нажмите кнопку "F1"

Рис. 5.3 –Размерность матрицы

**Шаг 2.** В «Задании 2» следует заполнить заголовки матрицы. Заголовки столбцов заполняются номерами узлов сети в порядке их возрастания, за исключением номера исходящего узла. Заголовки строк заполняются номерами ребер графа сети примыкающих к исходящему узлу. Результат заполнения заголовков таблицы для данного примера показан на рис. 5.4. После нажатия кнопки «Далее» программа даёт оценку правильности

выполнения шагового задания. Если задание выполнено неверно, то программа не допустит к выполнению следующего задания.

**Маршрутизация в сетях**

Пусть необходимо передеть информацию из 4 узла (исходящий узел), в узел 10 (входящий узел). Задание 2

Заполните заголовки матрицы для исходящего узла.

М 4 =  ×

Задание 1 выполнено верно

	1	2	3	5	6	7	8	9	10
4-2									
4-3									
4-5									

М 4

Для продолжения нажмите "Далее>>"

Выход Для вызова справки нажмите кнопку "F1" Далее >>

Рисунок 5.4 – Матрица с заголовками.

**Шаг 3.** В окне “Задание 3” следует заполнить клетки столбца матрицы рельефов, соответствующего входящему узлу (узлу 10 в данном случае), высотами ребер графа (рис.5.5), примыкающих к исходящему узлу 4.

**Маршрутизация в сетях**

Пусть необходимо передеть информацию из 4 узла (исходящий узел), в узел 10 (входящий узел). Задание 4

Укажите размерность матрицы исходящего узла

М 4 =  ×

Задание 1 выполнено верно

	1	2	3	5	6	7	8	9	10
4-2									4
4-3									4
4-5									3

М 4

Задание 2 выполнено верно.

Задание 3 выполнено успешно.

Этап 1 завершен.

М 5 =  ×

Для продолжения нажмите "Далее>>"

Выход Для вызова справки нажмите кнопку "F1" Далее >>

Рисунок 5.5 – Вид окна с выполненным этапом 1.

Шаг 4. Узел с наименьшим весом на пути первого выбора назначается исходящим и для него повторяются вышеописанные процедуры. Результаты применения процедур показаны на рисунке 5.6.

**Маршрутизация в сетях**

Пусть необходимо передереть информацию из 4 узла (исходящий узел), в узел 10 (входящий узел).

Заполните столбец матрицы для входящего узла. **Задание 9**

**М 4** = 3 × 9

	1	2	3	5	6	7	8	9	10
4-2									4
4-3									4
4-5									3

Задание 1 выполнено верно

Задание 2 выполнено верно.

Задание 3 выполнено успешно.

Этап 1 завершен.

**М 5** = 4 × 9

	1	2	3	4	6	7	8	9	10
5-3									3
5-4									3
5-6									3
5-8									2

Задание 4 выполнено верно

Задание 5 выполнено верно! Заголовки таблицы заполнены правильно

Задание 6 выполнено успешно.

Этап 2 выполнен.

**М 8** = 4 × 9

	1	2	3	4	5	6	7	9	10
8-5									2
8-7									2
8-9									2
8-10									1

Задание 7 выполнено верно

Задание 8 выполнено верно!

Задание 9 выполнено успешно!

Этап 3 выполнен.

Для продолжения нажмите "Далее>>"

Выход

Для вызова справки нажмите кнопку "F1"

Далее >>

Рисунок 5.6 – Вид окна с выполненными всеми этапами задания.

**Процедура 3.** Следует осуществить выбор кратчайшего пути в выбранной схеме сети с помощью матричного метода (алгоритма Дейкстры), используя произвольно выбранную метрику для каждого ребра графа.

При выполнении данного пункта на экране появится окно, в котором нужно построить топологию исходной сети. Метрика каждого ребра выбирается произвольно. После построения топологии сети в качестве алгоритма расчета в «Построителе маршрутов» выбирается метод Дейкстры, и программа определяет кратчайший путь. Узлы кратчайшего пути прорисовываются желтым цветом.

При построении маршрута используются следующие кнопки



- узел сети,



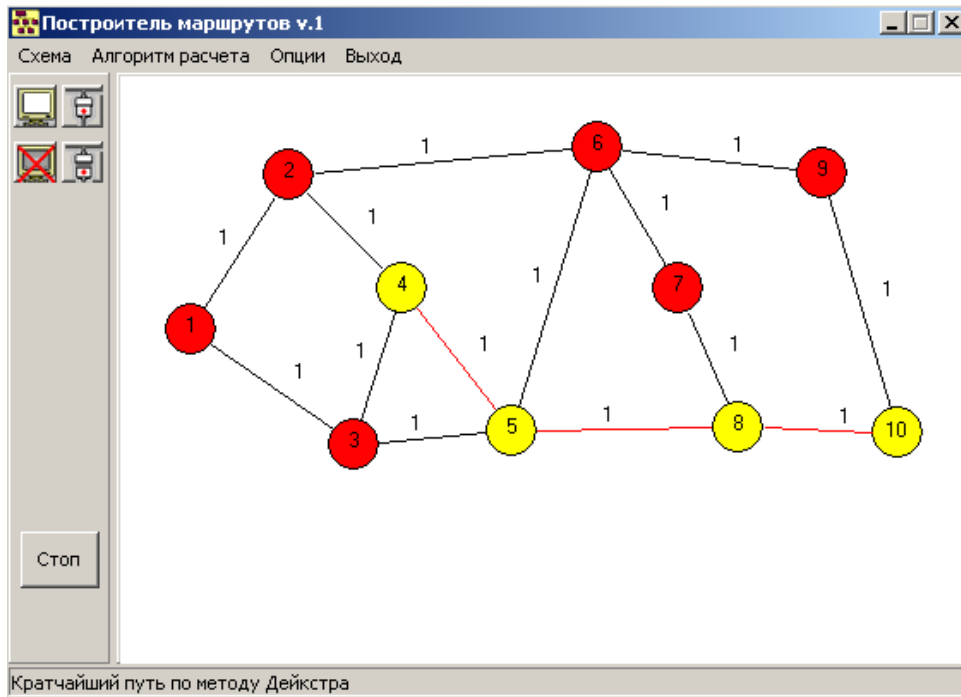
- линия соединения,



- удаление узла сети. Узел сети удаляется вместе с исходящими трактами.

**Процедура 4.** Следует осуществить выбор кратчайшего пути в выбранной схеме сети с помощью матричного метода (алгоритма Дейкстры), используя одинаковую и равную 1 для каждого ребра графа метрику.

Процедура выполняется аналогично *процедуре 3*. Метрика каждого ребра принимается равной 1. Результат отображен на рисунке 5.7.



## 6 Порядок оформления отчета

Отчет должен содержать следующие сведения.

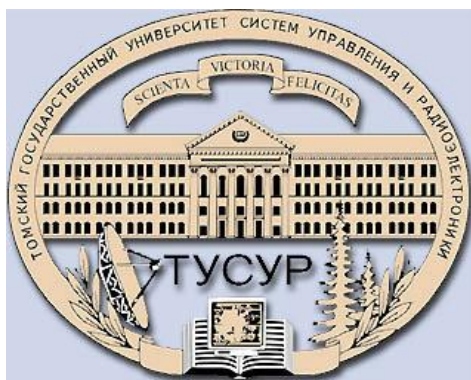
- 1 Цель работы
- 2 Результаты входного тестирования
- 3 Рельеф сети относительно входящего узла
- 4 Заполненные полностью таблицы маршрутизации для транзитных узлов
- 5 Вывод из сравнения результатов *процедуры 4* и шага 7 *процедуры 2* «Порядка выполнения лабораторной работы».
- 6 Распечатка результирующих модульных окон.
- 7 Выводы по работе.

Выводы по работе должны включать:

1. Оценка уровня работы и ее полезности для изучения темы.
2. Замечания по усовершенствованию программы.

Министерство образования и науки РФ  
Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
Томский государственный Университет Систем Управления и Радиоэлектроники  
(ТУСУР)

УТВЕРЖДАЮ  
Заведующий кафедрой ТОР, доцент  
\_\_\_\_\_ Е.П.Ворошилин  
\_\_\_\_ >> \_\_\_\_\_ <<  
\_\_\_\_\_ 2012 г.



Кафедра Телекоммуникаций и Основ Радиотехники  
(ТОР)

**«Изучение языка общения "Человек - Машина" (ММЛ)»**

РАЗРАБОТАЛИ  
Профессор кафедры ТОР \_\_\_\_\_ В. М. Винокуров  
Студент группы 145-5 \_\_\_\_\_ И. В. Федорчук

“ \_\_\_ ” \_\_\_\_\_ 2012

Томск 2012

## *1 Введение*

В современных цифровых телефонных станциях неотъемлемым компонентом является программное обеспечение. с помощью таких программ производится надзор, обслуживание и управление практически всеми узлами связи станции, а также управление работой центрального процессора.

В данном методическом пособии за основу взято ПО для телефонных станций фирмы “Siemens”. Для решения задач, а также для общения с оборудованием используется MML-Machine Man Language (Язык Человек-Машина). MML разделяется на входной язык – запросы и команды персонала, и выходной – ответные сообщения системы.

## 1. Цель работы

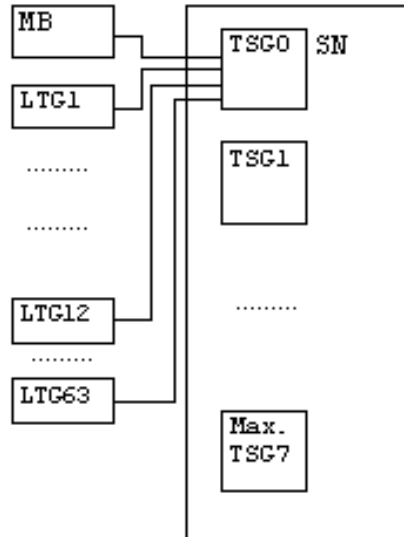
Целью работы является ознакомление со входным и выходным языком MML в программном обеспечении цифровых телефонных станций/

## 2. Входной язык MML.

Команда состоит из объекта-привязки команды (например, line/trunk group (LTG) – группа линий символов) и выполняемого действия (например, status inquiry – вопрос о статусе).



Рис.1 Структура формата команды



TSG - time stage group

SN - switching network

Рис. 2 - Структура периферии ЦАТС

Для упрощения записи действия и объекта их имена сокращают и используют аббревиатуру. Например, вместо фразы “Status inquiry of line/trunk group” используется аббревиатура “STAT LTG”.



Можно разделять действие и объект с помощью пробела, но это не обязательно.

Одновременно с командой можно отправлять системе и параметры (рис. 2). Параметр состоит из имени параметра и значения параметра. Для параметров также используется сокращенная аббревиатура. Точным аргументом LTG значения параметра будет, например:

TSG 0 with LTG Nr. 12

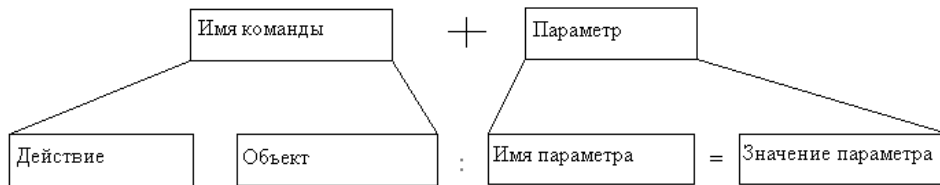


Рис.3 Структура расширенного формата команды

Если команда имеет параметр, то он должен быть введен после объекта и разделён с ним через двоеточие. Имя параметра и значение параметра соединяются знаком "=".

Значение параметра может иметь одно или несколько аргументов, которые снова могут быть разделены различными информационными блоками. В рассматриваемом примере один информационный объект является порядковым номером TSG (см. рис. 1), а другой одним из номеров LTG и они разделяются через тире "-".

Конец команды обозначается точкой с запятой ";".

Пример: STATLTG:LTG=0-12;

Команда может иметь много параметров. Их структура следующая:

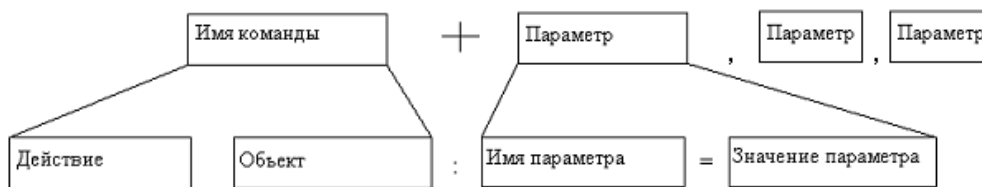


Рис.3 Структура расширенного формата команды с несколькими параметрами

Каждый параметр отделяют от других параметров запятой.

Следующий пример команды конфигурирует LTG с номером 0-12 в активное рабочее состояние:

```
CONFLTG:LTG=0-12,OST=ACT;
```

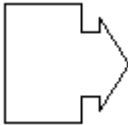
Где OST - рабочее состояние,

ACT активизировать.

Нажатием клавиши ENTER запускается команда и начинается её обработка центральным процессором СР.

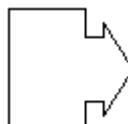
Если нужно отобразить различные директории, можно выполнить команду DISPSUB несколько раз или объединить аргументы параметра с помощью "&" в одну команду:

DISPSUB:DN=5700;  
DISPSUB:DN=5710;  
Можно также выполнять интервал параметров аргумента с использованием "&&".



DISPSUB:DN=5700&5710;

DISPSUB:DN=7300;  
DISPSUB:DN=7301;  
DISPSUB:DN=7302;  
DISPSUB:DN=7303;



DISPSUB:DN=7300&&7303;

Если требуется адресовать все возможные значения аргументов параметров, нужно ввести букву X.

### **Пример 1.**

Команда "Inquire status all DLU"

DLU - Digital Line Unit нумеруются от 10 до 2550,

Первый вариант: STATDLU:DLU=10&&2550;

Второй вариант: STATDLU:DLU=x.

Когда вводятся аргументы параметра, которые заменяют старую величину новой, их разделяют знаком "/".

### **Пример 2.**

Следующая команда изменяет период времени, заменяя старую величину 3ч 15мин на новую 4ч 30мин:

MOD CBLIMIT:TIME=4-30/3-15;

Это были примеры некоторых команд из тысяч существующих. Так, команды слагаются из 11 стандартных операторов.

Таблица 2.1 -Основные команды MML

Вопрос	Проверка	Создание	Изменение	Удаление
Display DISP	Diagnose DIAG	Create CR	Modify MOD	Cansel CAN
Status STAT		Enter ENTR	Activate ACT	Delete DEL
			Deactivate DACT	
			Configure CONF	

Перечислим основные особенности приведённых в таблице 2.1 групп операторов:

- **вопросы** не критичны, поскольку они ничего не изменяют в системе;
- **проверка** может быть выполнена только на заблокированном устройстве;
- какие-либо **создания** модифицируют базу данных. Если они не подтверждены, то их очень трудно отменить;
- **модификации** могут вызвать потерю соединений. Некоторыми командами можно обрезать обменные компоненты сети;
- **удаления** могут вызвать потери данных, которые тяжело восстановить.

Таблица 2.2 -Описание основных команд MML

ACT	Активирует объект, который уже существует
CAN DEL	Удаляет специальный ввод
CONF	Изменяет рабочее состояние объекта
CR ENTR	Присоединяет данные к объекту, который уже существует
DACT	Отменяет активацию введённую с помощью ACT
DIAG	Проверяет объект на дефекты

DISP	Отображает спец. данные объекта
MOD	Присоединяет и изменяет данные объекта
STAT	Отражает рабочее состояние объекта (объектов)

### 3. Выходной язык MML

События, происходящие в EWSD системе, носят название выходных. Выходные события отражаются на экране дисплея с помощью выходного языка MML.

Расположение информации на экране дисплея О&М показано на рисунке 3.1.

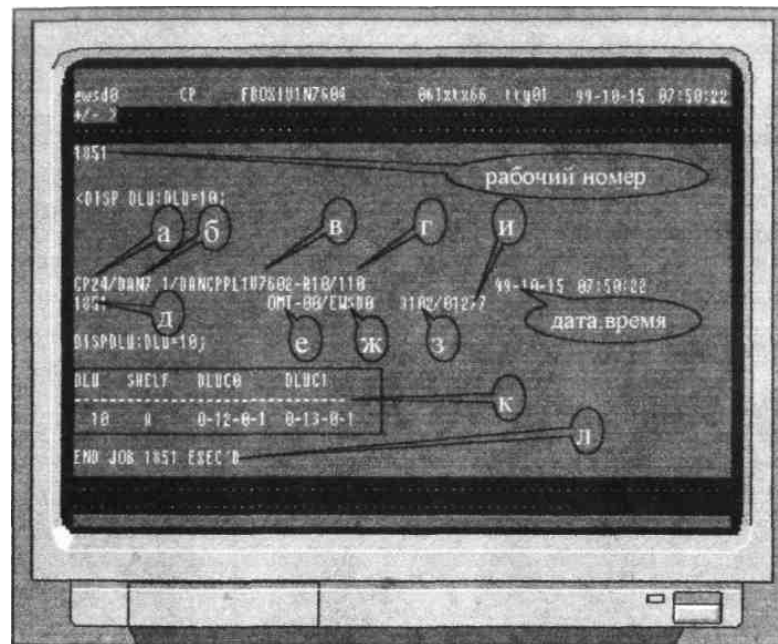


Рисунок 3.1- Вид О&М терминала при выводе выходного сообщения:

- а)* первая часть заголовка сообщения - имя станции или DCP, который сгенерировал сообщение;
- б)* обменный идентификатор;
- в)* метка CP, или программного обеспечения DCP. В данном случае это датская APS версия 7.1.
- г)* состояние CP
- д)* рабочий номер, располагаемый при входе;
- е)* идентификатор ОМТ, откуда вы ввели команду;
- ж)* идентификатор пользователя;
- з)* номер группы сообщения или типа данных, на которые сообщение распределено;

- и) номер сообщения выходной маски;
- к) сообщение;
- л) информация о завершении работы (в примере для работы №1851).

Система всегда отвечает на входные MML команды с помощью ответных сообщений.



Если команда введена правильно, оператор получает подтверждение.

**EXEC'D** полное завершение задания.

**ACCEPTED** показывает, что команда MML, была принята, но это может занять какое-то время, пока фактическая обработка начата. Это сообщение типично для диагностики, что занимает много времени до завершения.

**STARTED** информирует, что процесс с длительным временем обработки запущен.

Диагностика считается завершённой только по выводу сообщения **EXEC'D**.

**PART. EXEC'D** команда MML ещё не завершена.

**NOT EXEC'D** MML команда не была исполнена. Сообщение о причине невыполнения должно быть выведено на экран.

Сообщения о неудачах показывают, что нечто введённое не было корректным. Это может произойти, если вы совершили синтаксическую ошибку или не имеете прав на ввод этой команды.

Повторные сообщения спрашивают оператора на продолжение ввода.

Перед тем, как диагностировать LTG, нужно сконфигурировать их в MBL. Вы можете выполнить конфигурацию только после подтверждающего сообщения.

Таблица 3.1. Наиболее важные операционные состояния

<i>Статус</i>	<i>Расшифровка</i>	<i>Объяснение</i>	<i>Установка с помощью CONF</i>
<i>ACT</i>	Active	Нормальное активное состояние	+
<i>CBL</i>	Conditionally blocked	Выход соединения поддержан, новые не будут установлены	+
<i>MAS</i>	Master	Центральный процессор CP113	-
<i>MBL</i>	Maintenance blocked	Устройство, доступное только для эксплуатации	+
<i>NAC</i>	Not accessible	Устройство соединено с системой через дефектный блок и не может быть идентифицировано	-
<i>PLA</i>	Planned	Устройство спланировано, но еще не доступно физически	+
<i>SPR</i>	Spare	Спаренный процессор CP113	+
<i>STB</i>	Stand by	Устройство находится в резервном состоянии	+
<i>UNA</i>	Unavailable	Устройство дефектно	-

Подтверждающие сообщения предупреждают, что блокирование LTG исключает переключающуюся периферию из сервиса.

***FINISHED*** - работа полностью выполнена.

**STOPPED** - рабочий таймер был остановлен. Выполнение может быть отменено либо продолжено позже.

Аварийное сообщение указывает на то, что есть определённая ошибка, которая может испортить ОС. В зависимости от приоритета аварии, ОАМ персонал должен выполнить соответствующие действия в соответствии с процедурой SYP100 (см. приложение).

#### ***4. Порядок выполнения работы***

4.1 Ознакомьтесь с материалом, изложенным в пунктах 2 и 3 данного методического пособия.

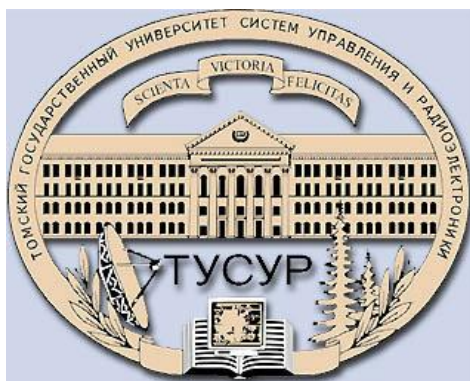
4.2 Запустите файл *Start-te.bat*. После демонстрации заставки Сименс нажмите кнопку Menu в панели задач. В предложенном списке выберите опцию Man Maschine Language MML двойным щелчком мыши. Продвижение по изложению описания осуществляется с помощью нажатия на изображения мыши в правом нижнем углу.

4.2 Изучите весь излагаемый материал по MML, отвечая на предложенные вопросы.

4.3 Ответьте на тесты и предъявите результат преподавателю.

**Министерство образования и науки РФ  
Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
Томский государственный Университет Систем Управления и  
Радиоэлектроники  
(ТУСУР)**

Кафедра Телекоммуникаций и Основ Радиотехники (ТОР)



УТВЕРЖДАЮ

Заведующий кафедрой ТОР, доцент

\_\_\_\_\_ Е.П.Ворошилин

<< \_\_\_ >> \_\_\_\_\_ 2012 г.

## Технология синхронной цифровой иерархии SDH

Лабораторная работа по дисциплине «Сети связи и системы коммутации»

РАЗРАБОТАЛИ

Профессор кафедры ТОР \_\_\_\_\_ В. М. Винокуров

студент гр. 147-2 \_\_\_\_\_ Ч.Р. Хусаинов

“ \_\_\_ ” \_\_\_\_\_ 2012

2012



Программа разработана в ходе дипломного проектирования в Томском Государственном Университете Систем Управления и Радиоэлектроники (ТУСУРе) на кафедре Телекоммуникаций и Основ Радиотехники (ТОР) в 2003 г студентом гр. 147-2 Хусаиновым Ч.Р. под руководством доцента Винокурова В.М. Работа прошла регистрацию в ОФАП Госкоорцентра Минобрнауки России и опубликована в журнале «Компьютерные учебные программы и инновации» – М: ГОСКООРЦЕНТР. – 2008. - N 7. стр. 93, свидетельство об отраслевой регистрации разработки №10439 от 16.04.2008; (Windows 2000/XP, Linux via Wine, алгоритмический язык DELPHI 7.0). Программа переработана и модернизирована в 2011г.

---

## 1. Цель работы.

Знакомство с технологией SDH.

## 2. Описание лабораторной установки.

Для того чтобы начать выполнять лабораторную работу запустите файл sdh.exe. Кроме этого файла в текущем каталоге находятся файл sdh.chm, который можно использовать как электронный учебник по данной технологии, так как он запускается независимо от исполняемого файла (см рис.1). В каталогах /Images, /Questions, /Answer содержатся текстовые и графические файлы-ресурсы.

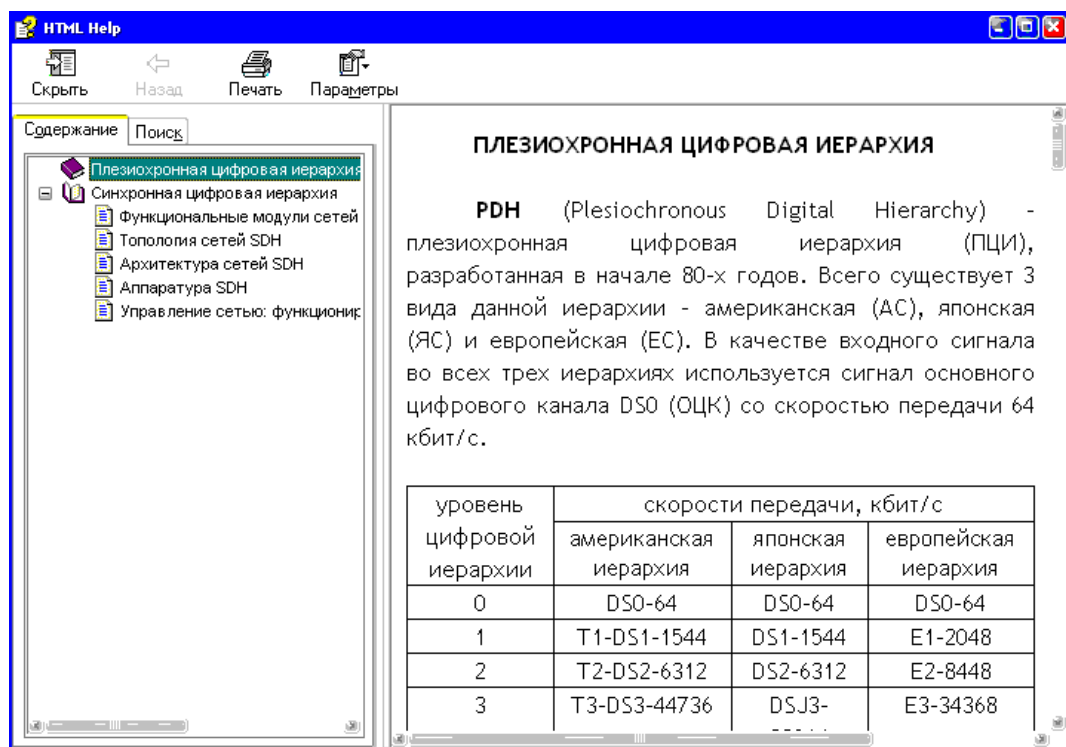


Рис. 1 Внешний вид окна приложения SDH.CHM (файл помощи)

После запуска исполняемого файла (sdh.exe) появится первое окно, в котором Вам будет предложено ознакомиться с принципами технологии SDH (то есть прочитать электронный учебник), а затем пройти входной тест.

Процесс обучение можно пропустить – для этого выберите «Начать тест без обучения» (см. рис. 1).

Входной тест – это 8 простых вопросов по технологии SDH (см. рис. 2). При помощи клавиш «Далее» и «Назад» Вы можете прокручивать вопросы; кнопка «Выход» - выход из программы. Для того чтобы пройти входной тест и приступить к выполнению лабораторной работы Вам необходимо ответить на любых 6 вопросов не более чем за 15 минут (пройденное время отображается в нижнем левом углу). Выбор ответа на вопрос осуществляется при помощи указателя мыши – слева от вариантов ответа изображены квадраты, которые необходимо активизировать (см. рис. 3).

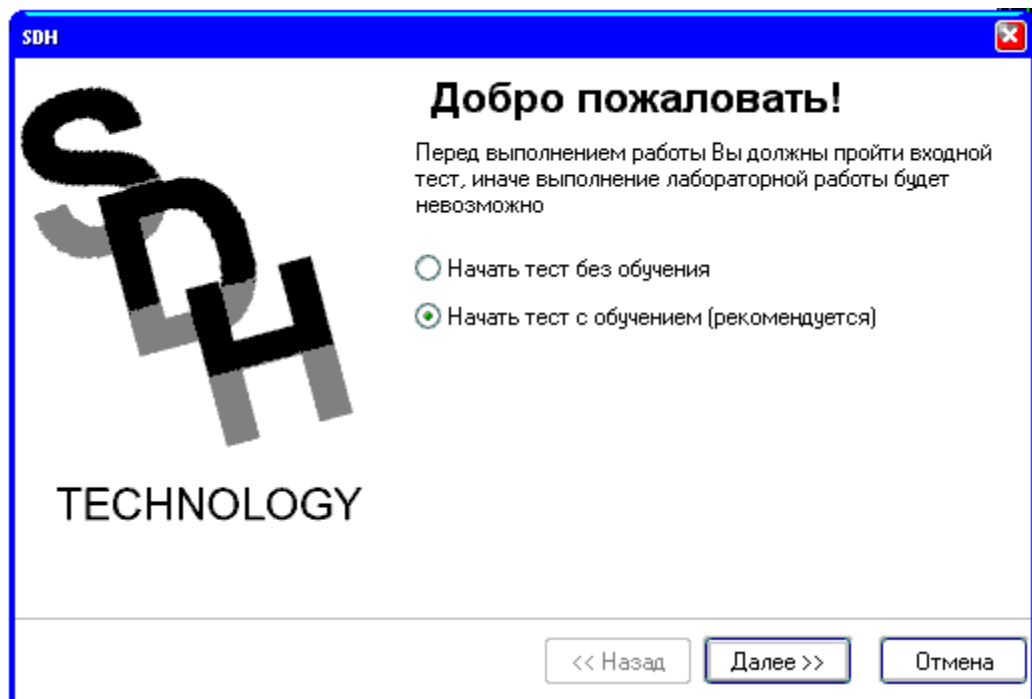


Рис. 2 Внешний вид первого окна

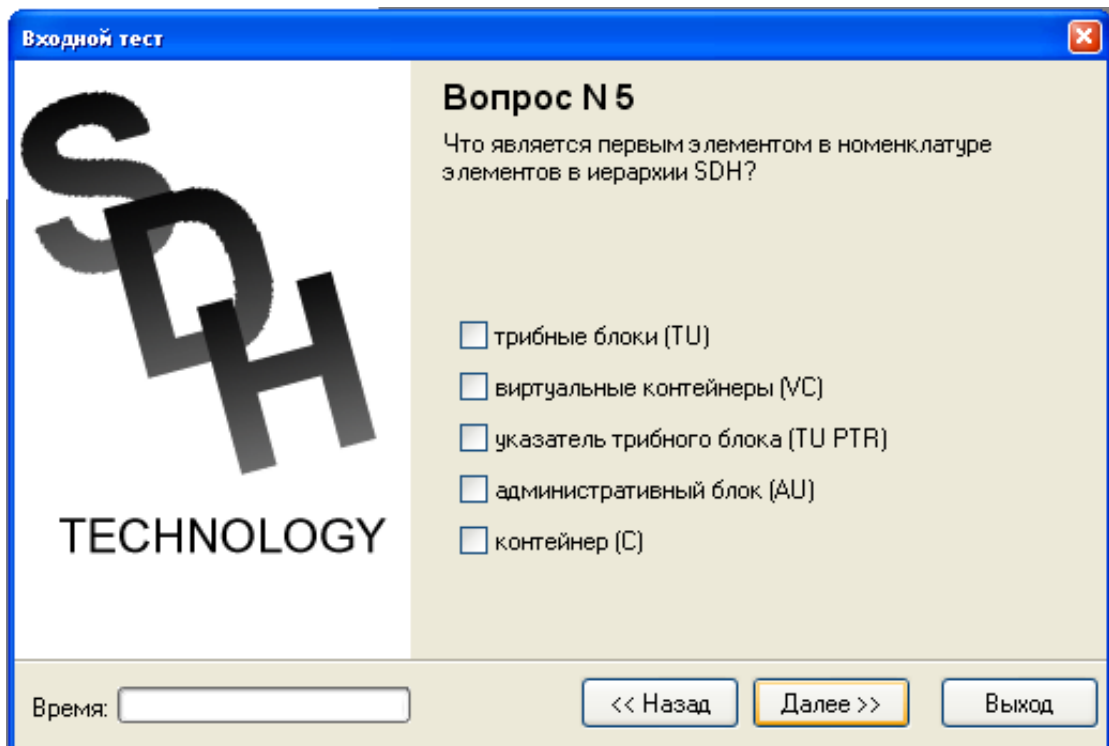


Рис. 3 Внешний вид окна с входным тестом

После прохождения входного теста Вам будет предложено выбрать один из шести вариантов задания (см. рис 4).

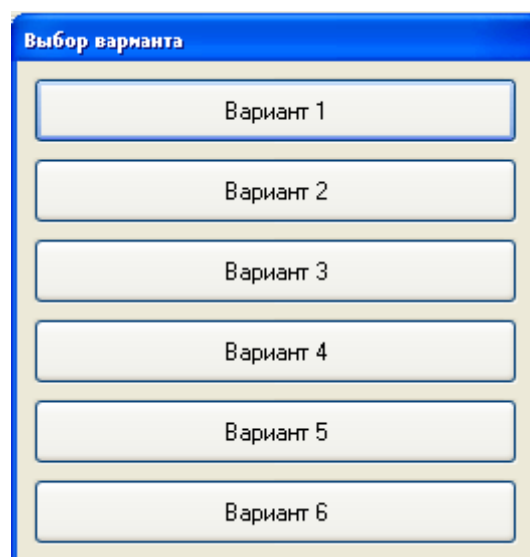






Рис.4 Выбор варианта

Затем появится основное окно компьютерной лабораторной работы (см. рис. 5).



Рис. 5 Внешний вид основного окна лабораторной работы

Лабораторная работа состоит из четырех частей:

-  - синтезирование из компонентного сигнала PDH синхронного транспортного модуля STM-1;
-  - тестовые задания по функциональным блокам сетей SDH и выбору оборудования;
-  - расчет межстанционного трафика;
-  - выделение компонентного сигнала PDH из синхронного транспортного модуля STM-1;

Задание считается выполненным, если иконка поменяла монохромное изображение на цветное. Например:



Для того чтобы начать любое из заданий необходимо два раза щелкнуть по требуемой иконке.

### 3. Лабораторные задания.

#### Знакомство со структурой и методами формирования потоков STM-1.

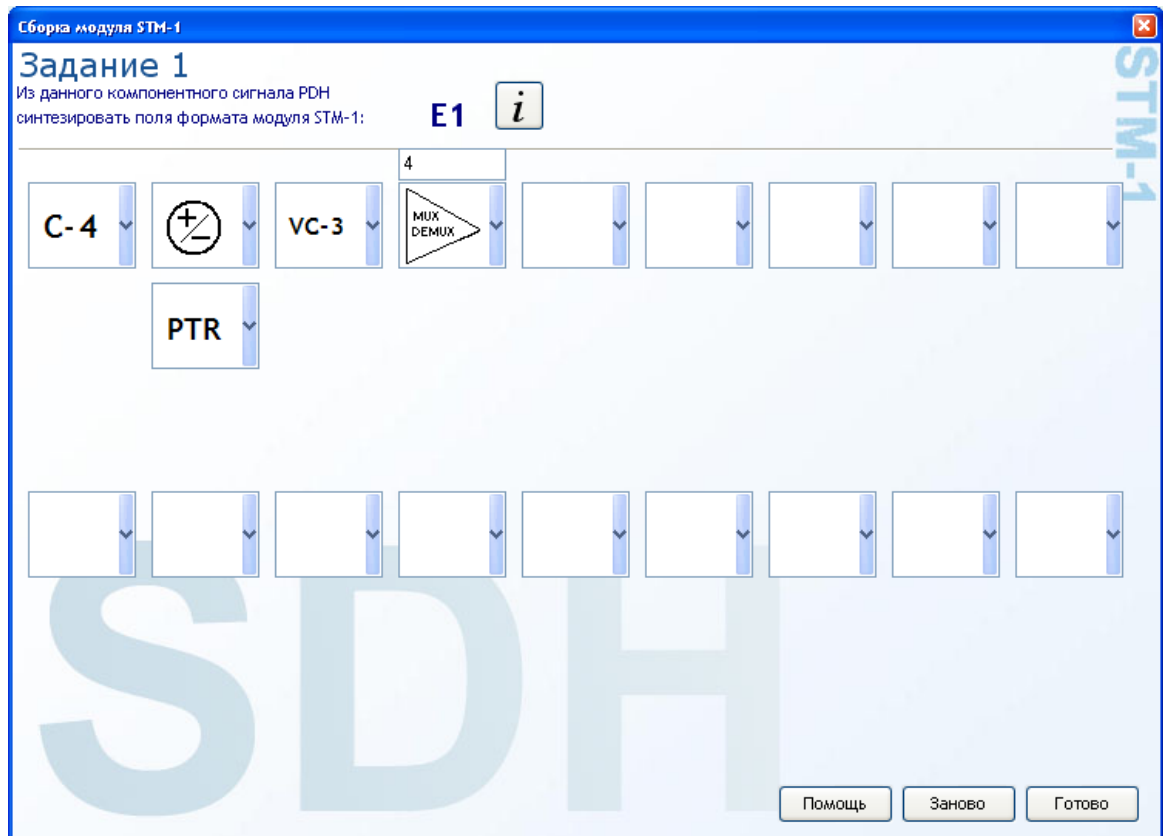


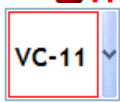








Рис. 6 Внешний вид окна «Сборка/разборка модуля STM-1»

При выборе иконки  появится окно «Сборка модуля STM-1» (см. рис. 6). Смысл данной работы состоит в следующем: используя всю номенклатуру элементов иерархии SDH, операции мультиплексирования и добавления служебной информации синтезировать синхронный транспортный модуль STM-1 из предложенного компонентного сигнала PDH. При выполнении этой операции необходимо опираться на схему мультиплексирования PDH трибов в технологии SDH по схеме ETSI, которая есть в учебнике (см. рис. 1).

Например, это сигнал E1 (как на рис.6). Тогда в первом окне нужно выбрать контейнер C-12, затем операцию добавления служебной информации – внизу появится окно для выбора служебной операции (РОН – маршрутный заголовок, SOH – секционный заголовок, PTR - указатель). Операцию следует продолжать до тех пор, пока не сформируется блок STM-1, то есть до присоединения к группе административных блоков (AUG) секционного заголовка (SOH).

В нижнем правом углу находятся три кнопки – «Помощь», «Заново» и «Готово». При нажатии на первую кнопку появляется помощь, на вторую – очищаются все формы; при нажатии на третью – программа проверяет правильность заполнения всех форм. Если работа выполнена некорректно, то появляется сообщение  **Неверно** программа показывает форму, в которой произошла ошибка . В противном случае окно закрывается и задание считается  выполненным. При выборе операции мультиплексирования сверху появляется поле для ввода – в нем необходимо указать коэффициент мультиплексирования  Также при выборе операции добавления служебной информации  появляется дополнительное поле , в котором необходимо указать какую служебную информацию Вы добавите. 

При выборе иконки  также появляется окно «Сборка /разборка модуля STM-1» (см. рис. 6). Смысл данной работы состоит в обратном: используя всю номенклатуру элементов иерархии SDH, операции демультиплексирования и извлечения служебной информации выделить предложенный компонентный сигнал PDH из синхронного транспортного модуля STM-1. Процесс заполнения аналогичен вышеописанному. Начинать нужно с операции извлечения из STM-1 секционного заголовка SOH.

### **Порядок выполнения работы «Сборка модуля STM-1».**

Запустить файл sdh.exe.

Ознакомиться с электронным учебником (см пункт 2).

Пройти входной тест.

Два раза кликнуть на иконке



Заполнить формы (см. рис.6) в соответствии с правилами выполнения данной лабораторной работы (см пункт 3.1).

После заполнения форм нажать на клавишу «Готово» и программа выйдет в основное окно (см рис. 5).

При успешном выполнении данной части работы иконка меняет свое изображение на цветное.

Если иконка не меняла свое изображение, то перейдите к пункту 3.2.4 или приступите к выполнению следующей части лабораторной работы.

### **Порядок выполнения работы «Выделение компонентного сигнала из модуля STM-1».**

Запустить файл sdh.exe.

Ознакомиться с электронным учебником (см пункт 2).

Пройти входной тест.

Два раза кликнуть на иконке



Заполнить формы (см. рис.6) в соответствии с правилами выполнения данной лабораторной работы (см пункт 3.1).

После заполнения форм нажать на клавишу «Готово» и программа выйдет в основное окно (см рис. 5).

При успешном выполнении данной части работы иконка меняет свое изображение на цветное.

Если иконка не меняла свое изображение, то перейдите к пункту 3.3.4 или приступите к выполнению следующей части лабораторной работы.

### **Ознакомление с функциональными блоками и устройствами SDH.**

При двойном нажатии на  появляются тестовые задания связанные с функциональными блоками сетей SDH (см. рис. 7).



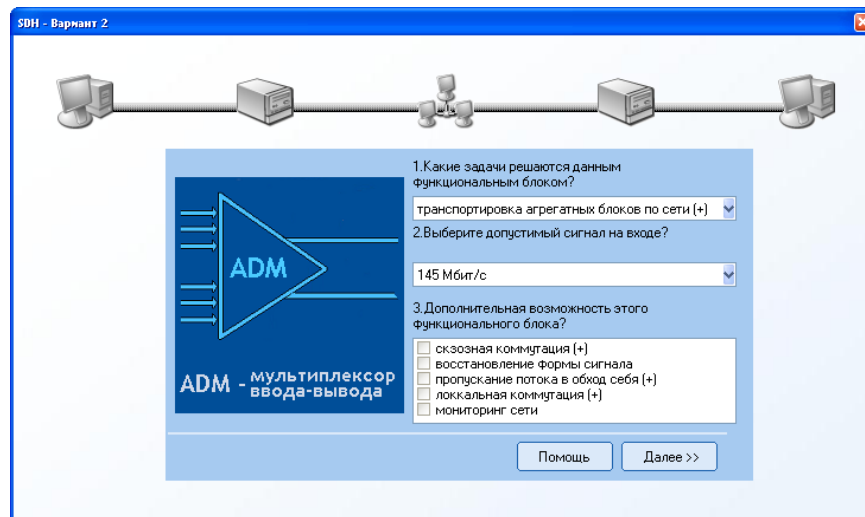


Рис. 7 Внешний вид окна с тестовыми заданиями

Первые два вопроса связаны с функциональными блоками – мультиплексорами (синхронными, терминальными, ввода/вывода), коммутаторами, концентраторами и регенераторами. Слева изображается графическое изображение блока и название, слева – три вопроса и варианты ответа. Если на первые два вопроса нужно отвечать однозначно, то правильных ответов на третий вопрос может быть от одного до пяти.

В следующих двух окнах предлагается выбрать оборудование из предложенного списка (см. рис. 8). В качестве ответа необходимо ввести название фирмы производителя (NOKIA, SIEMENS и т.д.).

В последнем задании (см. рис. 9) требуется правильно определить функциональный блок, изображенный на рисунке слева знаком вопроса. Для этого предлагается шесть кнопок - концентратор (HUB), регенератор (REG), терминальный мультиплексор (TM), мультиплексор ввода/вывода (ADM), синхронный коммутатор (SDXC) и управляющая система (OS).

В любом задании можно обратиться к электронному учебнику; для этого необходимо нажать клавишу «Помощь».

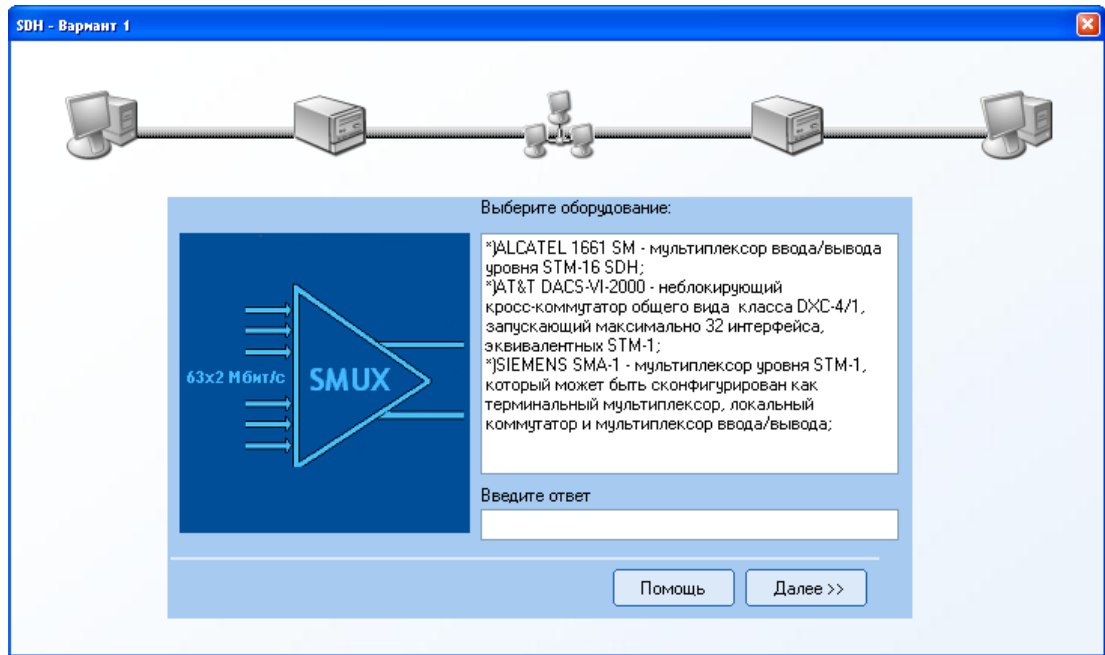


Рис. 8 Внешний вид окна с тестовыми заданиями

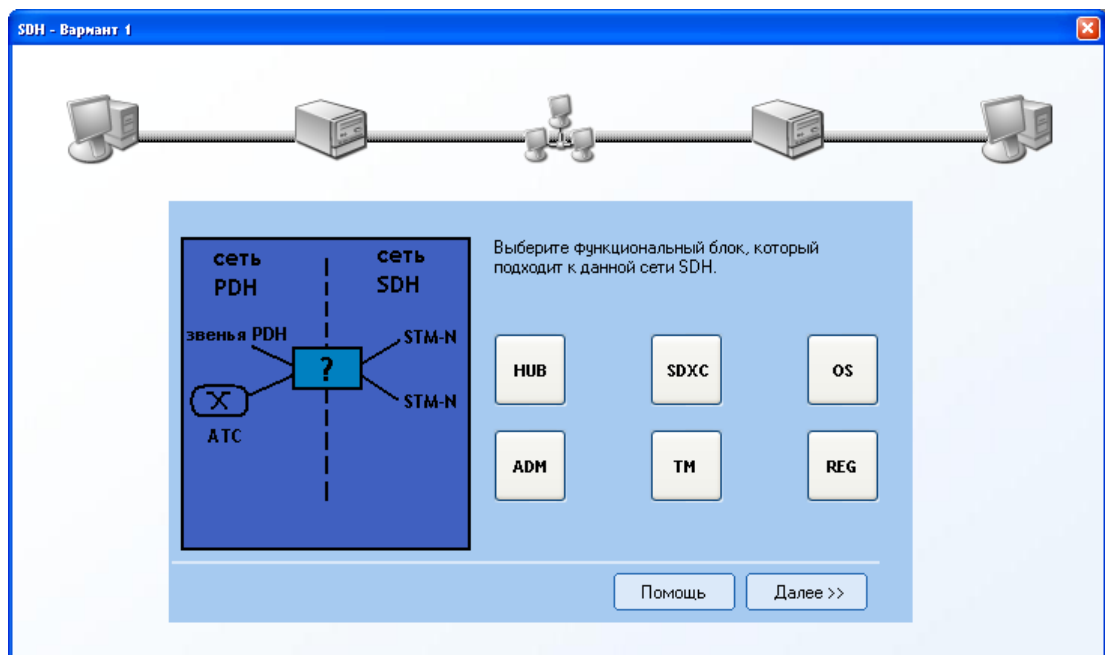


Рис. 9 Внешний вид окна с тестовыми заданиями

**Порядок выполнения работы «Ознакомление с функциональными блоками и устройствами SDH».**

Запустить файл sdh.exe.

Ознакомиться с электронным учебником (см пункт 2).

Пройти входной тест.



Два раза кликнуть на иконке

Ответить на предложенные тесты (см. рис.6) в соответствии с пунктом 3.4.

После ответа на последний тест программа выйдет в основное окно (см рис. 5).

При успешном выполнении данной части работы иконка меняет свое изображение на цветное.

Если иконка не поменяла свое изображение, то перейдите к пункту 3.5.4 или приступите к выполнению следующей части лабораторной работы.

### **Пример использования технологии SDH в телефонной сети.**

#### **Исходные предпосылки**

Пусть проектируется ГТС с использованием двунаправленного кольца SDH со 100% резервированием в качестве межстанционных СЛ. Сеть содержит  $N$  узловых станций (РАТС, АМТС, УСС). Примем, что ГТС имеет статус сети с явными потерями нагрузки, т.е. с отсутствием потерь.

Нагрузка от абонентов каждой РАТС оценивается по нормам ВСС РФ. Часть этой нагрузки будет присутствовать в межстанционном обмене. Количество каналов, необходимое для каждого межстанционного обмена рассчитывается по формуле Эрланга:

$$B = \frac{A^N}{N! \sum_{z=0}^N \frac{A^z}{z!}},$$

где  $A$  – нагрузка в эрлангах;

$N$  – количество каналов;

$B$  – вероятность блокировки.

Задаваясь величиной  $B$  в пределах 0,001 – 0,005, согласно нормам, можно определить для заданной величины  $A$  ресурс  $N$ .

В качестве каналов доступа узлов коммутации (РАТС, АМТС, УСС) к

первичной сети, реализованной на базе SDH, будем использовать плезиохронные системы передачи ИКМ -30 (стандарт E1).

Для расчета количества цифровых потоков типа E1, необходимых для реализации пучков соединительных линий (каналов) между различными станциями сети, следует учитывать:

- 1) число соединительных линий в направлении связи;
- 2) тип используемых соединительных линий (односторонние или двухсторонние);
- 3) тип используемой системы сигнализации.

При использовании односторонних линий и децентрализованной системы сигнализации (2ВСК, "2 из 6» и т.д.), для расчета требуемого числа потоков E1 от *i*-ой станции к *j*-ой станции, воспользуемся формулой:

$$N_{ijикм} = E_n [(V_{сл} - 1) / 30 + 1], \quad (3.1)$$

где  $N_{ijикм}$  - требуемое число цифровых потоков E1 от *i*-ой станции к *j*-ой станции;

$V_{сл}$  - число соединительных линий (каналов) между *i*-ой и *j*-ой станциями, ( $V_{сл} = V_{ij} + V_{ji}$ );

$E_n$  – знак целой части числа.

При использовании двухсторонних пучков и централизованной системы сигнализации (ОКС№7) воспользуемся формулой:

$$N_{ijикм} = E_n [(V_{сл} - 61) / 31 + 1] + 2 \quad (3.2)$$

Формула 5.2 справедлива, если  $V_{сл} > 60$  каналов. В противном случае необходимо использовать формулу 5.1, заменив  $V_{сл}$  на  $V_{ij}$ .

Для определения типа синхронного транспортного модуля используются следующие результаты:

- структура оптимального кольца с указанием местоположения мультиплексоров ввода-вывода (ADM) на данном кольце;
- схема взаимодействия ADM с узлами коммутации ГТС (РАТС, УВС и т.д.), АМТС и УСС;
- количество цифровых потоков стандарта E1 между различными узлами коммутации телефонной сети.

На основании вышеуказанных данных строится матрица  $M$  емкостей кратчайших путей и ребер.

Матрица  $M$  включает:

- перечень взаимодействующих узлов коммутации (станций) сети в соответствии со схемой построения ГТС;
- количество цифровых потоков стандарта E1 между различными узлами коммутации (станциями) ГТС;
- перечень участков кольца, которые используются для создания основных и резервных путей (маршрутов) для передачи цифровых потоков E1 между

различными узлами (станциями) ГТС.

Рассмотрим пример составления матрицы  $M$ .

Пусть известны:

- Схема построения ГТС (рисунок 10).
- Структура кольца (рисунок 11).
- Число ИКМ трактов передачи цифровых потоков Е1 между станциями ГТС (таблица 5.2).

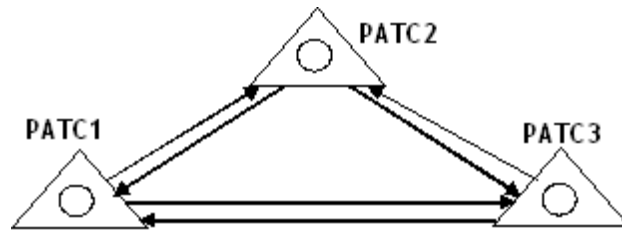


Рис.10 Схема построения ГТС

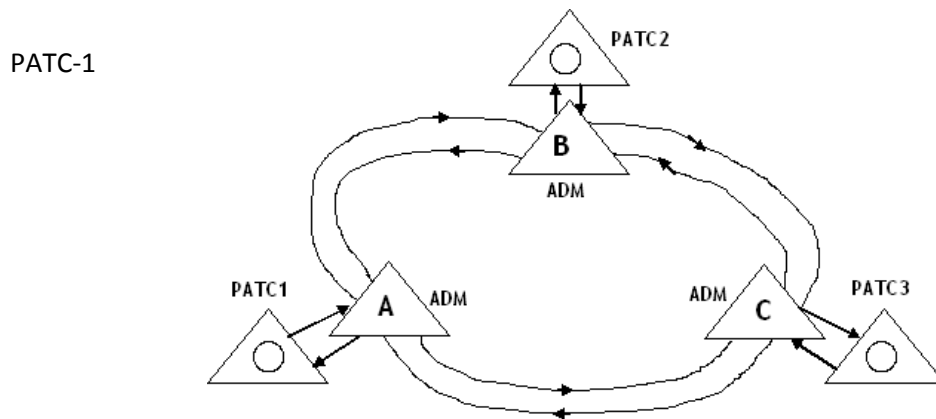


Рис. 11 Структура кольца

Для построения первичной сети на базе SDH используем двунаправленное кольцо со 100% резервированием в случае аварии на участках кольца.

Матрица  $M$  емкостей кратчайших путей и ребер для рассматриваемого примера представлена в таблице 1.

Таблица 1 – Матрица  $M$  кратчайших путей и ребер


Исходящая станция	Входящая станция	Количество цифровых потоков E1	Путь передачи	Участки кольца					
				A→B	B→A	B→C	C→B	C→A	A→C
РАТС-1 (А)	РАТС-2 (В)	15 15	Осн. Рез.	15			15		15
	РАТС-3 (С)	20 20	Осн. Рез.	20		20			20
РАТС-2 (В)	РАТС-3 (А)		Осн. Рез.		15	15		15	
	РАТС-3 (С)	25 25	Осн. Рез.		25	25			25
РАТС-3 (С)	РАТС-1 (А)	20 20	Осн. Рез.		20		20	20	
	РАТС-2 (В)	25 25	Осн. Рез.	25			25	25	
Суммарное число цифровых потоков E1 передаваемых по участкам кольца.				60	60	60	60	60	60


После заполнения матрицы М для всех взаимодействующих станций и узлов телефонной сети определяется суммарное число трактов E1 для каждого участка кольца первичной сети. Далее выбираем участок кольца на котором передается максимальное количество цифровых потоков E1.

Тип синхронного транспортного модуля выбирается с учетом стандартных уровней STM.

Если  $0 < S \leq 63$ , то выбираем STM-1  
 $63 < S \leq 252$ , то выбираем STM-4  
 $252 < S \leq 1008$  то выбираем STM-16, где S – количество потоков E1.

### Расчет межстанционной нагрузки и выбор уровня потока STM.

Предположим, что описанная процедура проделана и результаты сведены в таблицу. При выборе иконки  появляется окно с таблицами (см. рис. 10). В верхней части окна изображена структура кольца (используется двунаправленное кольцо со 100% резервированием) и таблица с данными о числе соединительных линий между узлами данного кольца (данную таблицу можно редактировать). Также к исходным данным следует отнести количество узлов, которое варьируется от 3 до 7.

Следующие три таблицы необходимо заполнить.  Первые две отражает число ИКМ трактов передачи цифровых потоков E1 между узлами; первая – в случае двунаправленных

пучков и централизованной системой сигнализации (ОКС N 7); вторая – в случае однонаправленных пучков и децентрализованной системы сигнализации (2ВСК). При заполнении этих таблиц используйте формулы 3.1-3.2. Третья таблица – «Матрица кратчайших путей и ребер» заполняется в тех ячейках, где проставлены знаки вопроса.

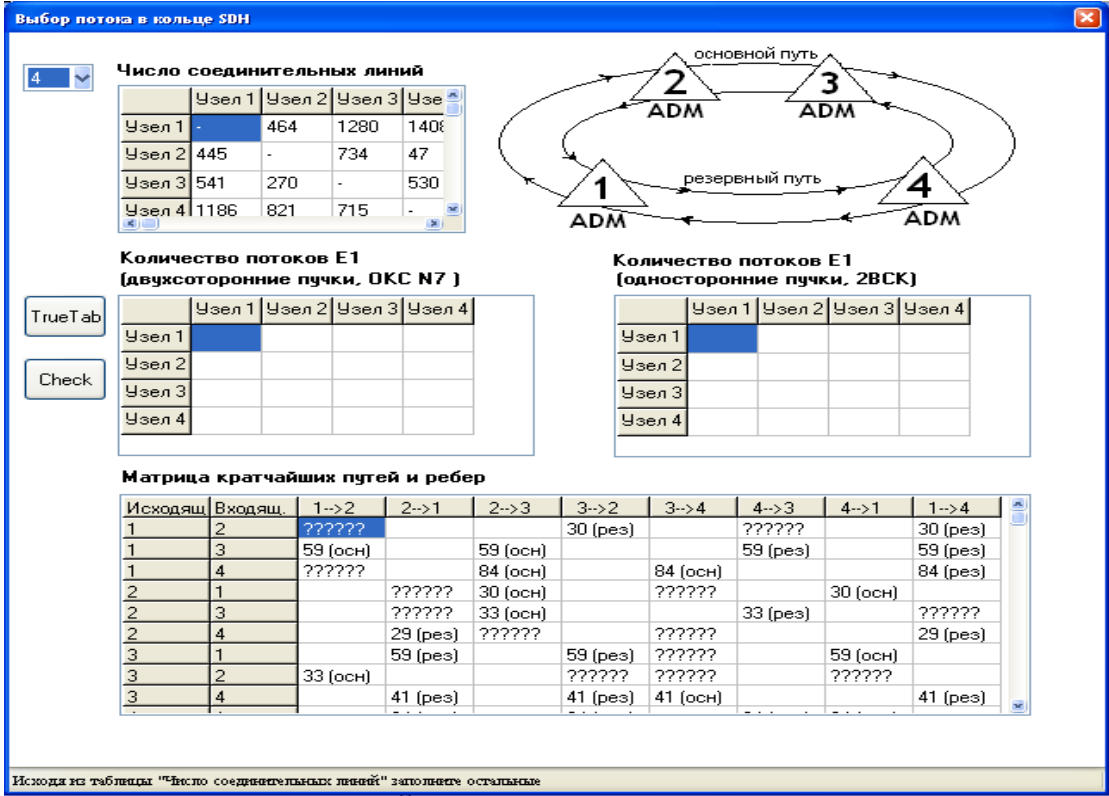






Рис. 10 Внешний вид окна «Расчет межстанционной нагрузки и выбор потока STM»

После заполнения всех необходимых таблиц нужно  нажать кнопку «Готово». Если хотя бы в одной ячейке был произведен некорректный ввод, то пр  **Неверно** ет сообщение

. В случае правильно заполнения таблиц появятся три кнопки – STM-1, STM-4, STM-16. Далее необходимо выбрать уровень потока STM-1 для данного технологического кольца. После нажатие на одну из кнопок окно закрывается и появляется основное окно лабораторной работы (см рис. 5). Если уровень потока определен правильно, то иконка  становится цветной.

## Порядок выполнения работы «Ознакомление с функциональными блоками и устройствами SDH».

1. Запустить файл sdh.exe.
2. Ознакомиться с электронным учебником (см пункт 2).
3. Пройти входной тест.
4. Два раза кликнуть на иконке 
5. Заполнить таблицу «Число соединительных линий» (если необходимо).
6. Заполнить таблицы «Число потоков E1» и «Матрица кратчайших путей и ребер» в соответствии с пунктом 3.6.2.
7. После заполнения таблиц нажать клавишу «Готово».
8. В случае неправильного заполнения появится сообщение «Неверно»; тогда необходимо перейти к пункту 3.7.6
9. Если таблицы заполнены правильно, то программа предложит выбрать уровень потока для данного кольца - STM-1, STM-4, STM-16.
10. После выбора потока программа выйдет в основное меню (см рис. 5).
11. При успешном выполнении данной части работы иконка меняет свое изображение на цветное.
12. Если иконка не меняла свое изображение, то перейдите к пункту 3.7.4 или приступите к выполнению следующей части лабораторной работы.

## Порядок оформления отчета

Отчет должен содержать следующие сведения.

- 8 Цель работы
- 9 Результаты входного тестирования
- 10 Блок схему программы.
- 11 Последовательность обращений к программе с оценками.
- 12 Распечатка модульных окон (если это возможно).
- 13 Выводы по работе.

Выводы по работе должны включать:

1. Оценка уровня работы и ее полезности для изучения темы.
2. Замечания по усовершенствованию программы.



**Министерство образования и науки РФ  
Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
Томский государственный Университет Систем Управления и Радиоэлектроники  
(ТУСУР)**

УТВЕРЖДАЮ

Заведующий кафедрой ТОР, доцент

\_\_\_\_\_ Е.П.Ворошилин

<< \_\_ >> \_\_\_\_\_ 2012 г.



Кафедра Телекоммуникаций и Основ Радиотехники

(ТОР)

**Сжатие информации в СПД с использованием кода Хаффмена.**

Методическое пособие по проведению компьютерных лабораторных работ по курсу «Сети связи и системы коммутации»

РАЗРАБОТАЛИ

Профессор каф. ТОР \_\_\_\_\_ В. М. Винокуров

\_\_\_\_\_ А.В.Комендатенко

“ \_\_ ” \_\_\_\_\_ 2012

Томск 2012

Программа разработана в ходе дипломного проектирования в Томском Государственном Университете Систем Управления и Радиоэлектроники (ТУСУРе) на кафедре Телекоммуникаций и Основ Радиотехники (ТОР) в 2000 г студентом гр.146<sub>3</sub> Комендатенко А.В. под руководством доцента Винокурова В.М. Программа переработана и модернизирована в 2011г.

*В работе приведены указания по выполнению лабораторной работы. Представлен материал для самостоятельной подготовки с необходимыми для ее выполнения теоретическими выкладками. Отображена последовательность действий при работе с макетом, а также форма представленного отчета.*

**ОГЛАВЛЕНИЕ**

ДОМАШНЕЕ ЗАДАНИЕ.....	60
КРАТКАЯ ТЕОРИЯ.....	61
1 Основные понятия.....	61
2 Основные характеристики источника сообщений.....	62
3 Задача статистического кодирования.....	63
4 Геометрическая интерпретация кода.....	64
5 Метод Хаффмена.....	67
6 Передача сообщения в канале с помехами.....	69
ОПИСАНИЕ ЛАБОРАТОРНОГО МАКЕТА.....	70
1 Назначение эмулируемых блоков.....	70
2 Работа с программой.....	71
ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ.....	71
1 Допуск к работе.....	71
2 Практическая часть.....	72
2.1 Исследование характеристик кода при равномерном кодировании простейшего... .алфавита.....	72
2.2 Исследование характеристик кода при эффективном кодировании простейшего... алфавита.....	73
2.3 Исследование длительности кодовой последовательности при равномерном и... эффективном кодировании простого сообщения.....	73
2.4 Декодирование блоковой последовательности содержащей ошибочные символы.. .....	74
2.5 Декодирование непрерывной последовательности содержащей ошибочные..... символы.....	75
2.6 Оформление отчета.....	75
КОНТРОЛЬНЫЕ ВОПРОСЫ.....	<b>Ошибка!</b>
	<b>Закладка не определена.</b>
СПИСОК ЛИТЕРАТУРЫ.....	<b>Ошибка!</b>
	<b>Закладка не определена.</b>

## ЦЕЛЬ РАБОТЫ

*Целью настоящей работы является исследование основных характеристик кода Хаффмена на примере сравнения его с равномерным двоичным кодом, а также знакомство с эффектом трека ошибок, возникающем при передаче по каналу связи дискретных сообщений, если в канале присутствует источник помех.*

## ДОМАШНЕЕ ЗАДАНИЕ

1. Ознакомиться с теорией дискретных сообщений, кодированием методом Хаффмена, основными характеристиками кода.
2. В соответствии с заданным преподавателем вариантом произвести двоичное кодирование алфавита источника, применяя:
  - а) равномерное кодирование;
  - б) эффективное кодирование методом Хаффмена.
3. Определить основные характеристики кода ( $\bar{l}$ ,  $H(x)$ ,  $H_{max}$ ,  $r(H)$ ,  $\kappa_{оэ}$ ,  $\kappa_{сс}$ ). Соответствующий заданному варианту алфавит источника и распределение вероятностей сообщений выписать из таблицы 1.

**Таблица 1 - Распределение вероятностей букв алфавита источника**

<i>A</i>	<i>НОМЕР ВАРИАНТА</i>					
	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>6</i>
<i>a<sub>1</sub></i>	<i>0,37</i>	<i>0,07</i>	<i>0,14</i>	<i>0,67</i>	<i>0,23</i>	<i>0,19</i>
<i>a<sub>2</sub></i>	<i>0,1</i>	<i>0,44</i>	<i>0,09</i>	<i>0,03</i>	<i>0,29</i>	<i>0,22</i>
<i>a<sub>3</sub></i>	<i>0,02</i>	<i>0,16</i>	<i>0,57</i>	<i>0,01</i>	<i>0,08</i>	<i>0,18</i>
<i>a<sub>4</sub></i>	<i>0,25</i>	<i>0,29</i>	<i>0,05</i>	<i>0,22</i>	<i>0,1</i>	<i>0,21</i>
<i>a<sub>5</sub></i>	<i>0,26</i>	<i>0,04</i>	<i>0,15</i>	<i>0,07</i>	<i>0,3</i>	<i>0,2</i>

## КРАТКАЯ ТЕОРИЯ

### 1 Основные понятия

**Сообщение** является носителем некоторой информации, отражающей состояние какой-либо материальной системы. Как правило, сообщение формируется человеком (возможно устройством), который является наблюдателем материальной системы. Основная функция сообщения – передача информации другому человеку (устройству), если процесс наблюдения за материальной системой последнему недоступен. В теории сообщений материальную систему вместе с независимым наблюдателем называют **источником сообщений**. Передача сообщений осуществляется посредством физического процесса, в роли которого может выступать **сигнал**. Средства обработки сигнала и среда, в которой он впоследствии распространяется, называют **каналом связи**. Канал связи позволяет передавать сообщение от источника к получателю. Если получателю сообщения известен закон, по которому связываются переданное сообщение и принятый сигнал, то получатель способен восстановить переданную информацию.

Канал, как неидеальная среда, обладает свойством искажать передаваемые сигналы, что отражается на неоднозначности при восстановлении сообщения. Это связано с наличием в канале случайных нестационарных процессов, которые в радиотехнике называют **шумами** и **помехами**.

Источник сообщений отслеживает непрерывный ряд сведений о состоянии материальной системы, но выдаваемые им сообщения могут представлять конечное (счетное) множество  $A$ . Даже если  $A$  является бесконечным множеством, то на практике работу источника можно построить так, что в любом случае множество  $A$  останется счетным. Если это возможно, то такие источники назовем **дискретными**, в противном случае мы будем иметь дело с **непрерывными** источниками сообщений.

*Если источник вырабатывает множество сообщений  $A$ , то для передачи информации необходимо сопоставить каждое сообщение этого множество с некоторой функцией из множества  $Z$ . Все выбранные сигналы<sup>1</sup> объединим и назовем множеством  $Z_0$ . Желательно, чтобы соответствие между сообщением и сигналом было взаимно однозначным или, в крайнем случае, чтобы каждому сигналу из множества  $Z_0$  соответствовало единственное сообщение из  $A$ . Такой подход позволяет восстановить переданное сообщение.*

Зачастую множество  $A$  содержит огромное количество сообщений (и уж тем более для непрерывного источника). Поэтому, вместо словарного сопоставления множества  $A$  с сигналами  $z(t)$  каждое сообщение разбивают на элементарные составляющие. Как правило, эти элементарные составляющие называют буквами, которые объединяют во множество  $X$ . Легко убедиться в конечности множества  $X$  и, кроме того, в сокращении множества  $X$  относительно исходного множества  $A$ . Аппаратное решение этого действия осуществляется на уровне источника. К примеру разбиения исходного множества на элементарные составляющие можно привести читаемый нами текст. В качестве элементарного сообщения выберем все входящие в него буквы русского алфавита и сопутствующие символы (пробелы, знаки препинания и т.п.). Новое множество получилось гораздо меньше исходного и вряд ли количество сообщений в нем превысит пятидесяти.

---

<sup>1</sup> Здесь, и в дальнейшем, мы не будем выделять разницы между функциями  $z(t)$  и понятием сигнал, поскольку при наших введениях соответствий это одно и то же.

Множество сообщений, получаемое при расчленении исходного множества на элементарные сообщения, называют **алфавитом** источника. Число элементов, входящих в алфавит источника, отождествляется с понятием **объем алфавита** и обозначается буквой  $l$ . Нетрудно представить себе отображение любого сообщения множества  $A$  в виде:

$$a = \left( x^{(1)}, x^{(2)}, \dots, x^{(L)} \right) \quad (1.1)$$

Расчленение исходного множества на элементарные составляющие позволяет (как мы уже убедились) сократить объем алфавита источника и, тем самым, сократить выбор множества функций из  $Z$ , поскольку теперь нам необходимо ввести соответствие между буквами алфавита источника  $x_k$  и функциями  $z(t)$ . Всего нам потребуется 1 функций  $z(t)$  для передачи сообщений из множества  $A$ .

Если объем алфавита источника слишком велик, то можно пойти на дальнейшее преобразование. Суть его заключается в следующем. Множество элементарных сообщений  $X$  приводится к новому множеству  $Y$ , объем которого не превышает значения 1, т.е. если множество  $Y$  представлено символами  $\{y_1, y_2, \dots, y_m\}$ , то число  $m < l$ . Теперь каждую букву  $x^{(k)}$  можно представить в виде последовательности символов  $y$  по аналогии с (1.1). Преобразование алфавита  $X$  в алфавит  $Y$  является процессом **кодирования**; свод правил, по которым осуществляется кодирование, называется **кодом**; объем  $m$  алфавита  $Y$  называется **основанием кода**.

## 2 Основные характеристики источника сообщений

К основным характеристикам источника сообщений относятся: количество информации, приходящееся на одно сообщение; энтропия источника сообщений; производительность источника; избыточность алфавита.

Для того, чтобы можно было сравнивать между собой сообщения  $a_i$  алфавита  $A$ , вырабатываемые источником с вероятностями  $p(a_i)$ , вводится понятие **количество информации**. При использовании кода с основанием  $m = 2$  количество информации определяется выражением:

$$I(a_i) = \log_2 \left( \frac{1}{p(a_i)} \right) = -\log_2 (p(a_i)), \quad \left[ \frac{\text{бит}}{\text{сообщение}} \right], \quad (2.1)$$

и показывает, сколько информации содержится в одном бите сообщения.

Из выражения (2.1) нетрудно заметить, что чем выше вероятность сообщения, тем меньшее количество информации оно несет, и наоборот. Это вполне согласуется с нормальными представлениями. Т.е., если источник вырабатывает единственное сообщение ( $p(a_i) = 1$ ), то получатель сообщения заранее его знает и, следовательно, с его получением он (получатель) не почерпнет для себя ничего нового.

Усреднение количества информации по всему объему алфавита дает количественную меру, называемую **энтропией источника**. Основным выражением для энтропии источника является:

$$H(A) = -\sum_{i=1}^l p(a_i) \cdot \log_2 p(a_i) \quad \left[ \frac{\text{бит}}{\text{сообщение}} \right]. \quad (2.2)$$

По аналогии с представлением в физике, энтропия показывает степень неопределенности состояния системы (в нашем случае системой выступает источник сообщений). Если источник вырабатывает единственное сообщение с вероятностью  $p(a_i) = 1$ , то состояние системы предопределено и энтропия равна нулю. Нетрудно показать из (2.1), что энтропия

источника принимает максимальное значение при равновероятном алфавите источника, т.е. для любого сообщения  $a_i$  из алфавита  $A$  справедливо равенство  $p(a_i) = p = 1/L$ . При этом выражение (2.2) сводится к виду:

$$H_{\max}(A) = \log_2 \left[ \frac{\text{бит}}{\text{сообщение}} \right], \quad (2.3)$$

Среднее количество информации, выдаваемое источником в единицу времени, называют **производительностью источника** и определяют по формуле:

$$H^*(A) = \frac{H(A)}{T}, \left[ \frac{\text{бит}}{\text{сообщение}} \right], \quad (2.4)$$

где  $T$  – среднее время, отводимое на передачу одного сообщения.

Выражение для средней длительности сообщения имеет вид:

$$T = \tau_{cp} = \bar{\tau} = \sum_{i=1}^L \tau_i \cdot p(a_i), \quad (2.5)$$

здесь  $\tau_i$  – время передачи  $i$  – го сообщения заданного алфавита.

Для характеристики алфавита источника сообщений представляет интерес сравнение энтропии  $H(A)$  с максимально возможной при данном алфавите энтропией  $H_{\max}(A)$ . Такой характеристикой выступает понятие избыточности алфавита в данном источнике сообщений:

$$r(A) = 1 - \frac{H(A)}{H_{\max}(A)} \quad (2.6)$$

В частности, если учесть зависимость вероятностей появления букв от ранее предшествовавших букв, то смысл избыточности алфавита заключается в следующем. Предположим, что на выходе системы связи появилось сообщение «распреде...», то с вероятностью, близкой к единице, мы можем ожидать появления следующим сообщением «...ние...». Кроме того, если контекст говорит о математической направленности текста, то вероятно далее будет идти слово «вероятностей». Как видим, налицо избыточность русского алфавита.

Необходимо отметить, процесс преобразования множества  $A$  в алфавит источника  $X$  (часто это преобразование называют укрупнением алфавита) не приводит к увеличению избыточности алфавита. Доказательство этого утверждение можно найти в [1].

### 3 Задача статистического кодирования

Для того, чтобы сформулировать задачу статистического кодирования приведем пример из [2].

**Пример 1.** Пусть  $A = \{a_1, a_2, a_3\}$ . Некоторые возможные коды для букв алфавита  $A$  представлены в таблице 2.

Таблица 2 - Возможные коды для простейшего алфавита

$A$	$Ko$ $\partial I$	КОД 2	КОД 3
$a_1$	1	01	0
$a_2$	0	10	10
$a_3$	01	11	111

Код 1 не является однозначно декодируемым кодом, так как комбинация  $a_2$  является начальной частью комбинации  $a_3$ . Для доказательства этого рассмотрим, например, двоичную последовательность 0101. Она может быть декодирована одним из сообщений:

- $a_2, a_1, a_2, a_1$ ;
- $a_3, a_3$ ;
- $a_3, a_2, a_1$ .

Код 2 декодируется однозначно, поскольку все кодовые комбинации этого кода имеют равные длины и различны.

Код 3 также однозначно декодируемый, поскольку никакая его кодовая комбинация не является началом (префиксом) другого кодового слова.

*Итак, из представленного примера нетрудно сформулировать задачу статистического кодирования. Основное положение ее заключается в однозначности декодирования кодовой последовательности, т.е. совершенно правильно восстановить переданное сообщение из принятой последовательности кодовых символов «1» и «0». С другой стороны необходимо кодировать сообщение минимальным количеством двоичных символов, что обеспечит в отведенное время передать большее число сообщений. Задача статистического кодирования сводится к отысканию таких кодов.*

#### 4 Геометрическая интерпретация кода

*Существует несколько методов геометрического отображения кода, но наиболее наглядным является **кодоевое дерево** множества кодовых слов. Данная интерпретация позаимствована из теории графов, и вид кодового дерева представляет собой ничто иное, как граф кода.*



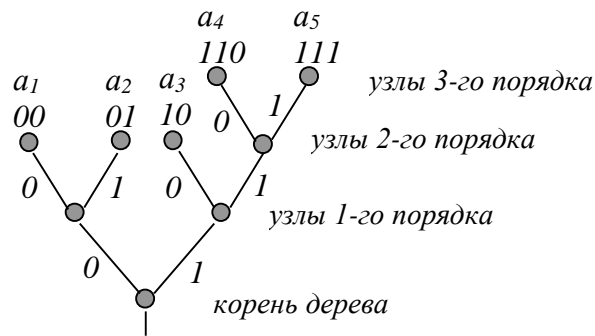


Рисунок 1 - Пример кодового дерева множества кодовых слов

Кодовое дерево (см. рисунок 1) включает в себя **узлы** и соединяющие отдельные узлы **ветви дерева** или, по аналогии, **ребра графа**. Построение дерева ведется от основного узла, называемого **корневой точкой** (**корень дерева**). Из этого узла, как впрочем, и из любого другого узла дерева, исходят ветви дерева, количество которых определяется основанием кода. Так, например, для кода с основанием  $t = 2$  из каждого узла дерева исходит две ветви. Каждая ветвь дерева соединяет узел, породивший ее, с узлом, следующим по иерархии. Если узел не порождает собой новые ветви дерева, то он является **концевым** и с ним сопоставляется код элементарного сообщения. Каждая ветвь дерева нумеруется символами применяемого кода (в двоичной системе этими символами выступают «1» и «0»). Продвижение по дереву от корневой точки до конечного узла определяет кодовую комбинацию отдельного элементарного сообщения, при этом код составляется по номерам ветвей (номер первой ветви определяет значение старшего  $i$ -го разряда комбинации, номер второй – значение  $(i-1)$ -го разряда, и т.д.). Сопоставление кодовых комбинаций с конечными узлами, равно как и нумерация ветвей, определяется некоторыми правилами, обеспечивающими однозначное декодирование кодовой последовательности.

**Теорема 1. Достаточное условие однозначного декодирования.** Для того чтобы таблица последовательностей кодовых знаков (кодовых комбинаций) являлась кодом, т.е. удовлетворяла условию однозначного декодирования, достаточно, чтобы ни одна из этих кодовых комбинаций не была началом (префиксом) других более длинных кодовых комбинаций этой таблицы.

Коды, удовлетворяющие теореме 1, называют **префиксными кодами**. Если кодовые комбинации сопоставляются лишь с конечными узлами кодового дерева, то дерево отображает **префиксный код**. Нумерация ветвей должна соблюдаться при разработке передающего и приемного устройства.

Вернемся к примеру 1.

Мы разобрались в однозначном декодировании. Рисунок 2 иллюстрирует кодовые деревья для кодов 2 и 3 таблицы 2. Необходимо разобраться, в каком из представленных кодов минимальным количеством символов передается большее количество элементарных сообщений.

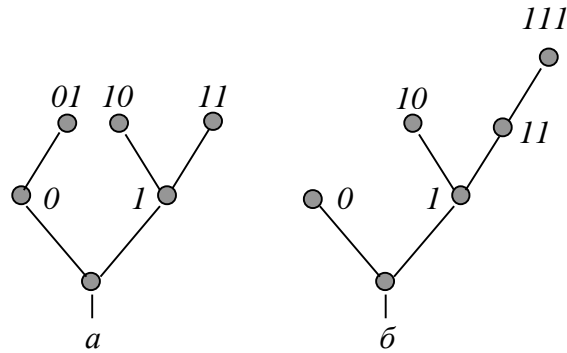


Рисунок 2 - Кодовые деревья для кода 2 (а) и кода 3 (б)

Среднее число двоичных символов  $\bar{l}$  на одно сообщение алфавита объемом  $L$ , для двоичных кодов, определяется выражением:

$$\bar{l} = \sum_{i=1}^L p(a_i) \cdot l_i, \quad (4.1)$$

где  $l_i$  - количество двоичных символов в кодовой комбинации  $i$ -го элементарного сообщения.

Формула (4.1) представляет собой ничто иное, как математическое ожидание количества бит, приходящихся на одно элементарное сообщение алфавита источника.

Предположим, что для алфавита  $A$  в примере 1 вероятность появления сообщения  $a_1$  равна 0,2, сообщения  $a_2$  - 0,3, а сообщения  $a_3$  - 0,5. тогда среднее число двоичных символов  $\bar{l}$ , приходящихся на одно сообщение

$$\text{для кода 2 составляет } \bar{l} = 0,2 \cdot 2 + 0,3 \cdot 2 + 0,5 \cdot 2 = 2,0,$$

$$\text{а для кода 3 составляет } \bar{l} = 0,2 \cdot 1 + 0,3 \cdot 2 + 0,5 \cdot 3 = 2,3,$$

т.е. код 2 в среднем экономичнее кода 3.

Рассмотрим еще один код - код 4, который сообщению  $a_1$  ставит в соответствие 10; сообщению  $a_2$  - 11;  $a_3$  - 0. Среднее число двоичных символов, приходящихся на одно сообщение этого кода:

$$\bar{l} = 0,2 \cdot 2 + 0,3 \cdot 2 + 0,5 \cdot 1 = 1,5,$$

т.е. код 4 экономичнее кода 3, и кода 2. Существует ли код для заданного источника, при котором среднее число двоичных символов будет минимальным?

**Теорема 2. Теорема Шеннона об эффективном кодировании.** Исходную последовательность символов (букв) алфавита  $A$  можно эффективно закодировать так, что среднее количество двоичных знаков кодовой последовательности будет как угодно близко, но не менее энтропии исходной последовательности алфавита  $A$ , т.е.:

$$\bar{l} < H(A) + \varepsilon, \quad (4.2)$$

где  $\varepsilon$  - сколь угодно малая величина.

*Теорема Шеннона указывает на существование эффективных кодов, но не приводит алгоритмов их реализации. Поиском таких кодов занимались многие исследователи и, наряду с многочисленными алгоритмами, в 1952 году вышла статья, автором которой являлся Хаффмен, рассматривающая один из методов эффективного кодирования.*

## 5 Метод Хаффмена

Пусть сообщения входного алфавита  $A = \{a_1, a_2, \dots, a_k\}$  имеют соответственно вероятности  $p_1, p_2, \dots, p_k$ . Тогда алгоритм кодирования методом Хаффмена состоит в следующем:

1. Сообщения располагаются в столбец в порядке убывания вероятности их появления.
2. Два самых маловероятных сообщения  $a_{k-1}$  и  $a_k$  объединяем в одно сообщение  $b$ , которое имеет вероятность, равную сумме вероятностей сообщений  $a_{k-1}$  и  $a_k$ , т.е.  $p_{k-1} + p_k$ . В результате получим сообщения  $a_1, a_2, \dots, a_{k-2}, b$ , вероятности которых  $p_1, p_2, \dots, p_{k-2}, (p_{k-1} + p_k)$ . Полученные сообщения вновь располагаем в порядке убывания вероятностей.
3. Повторяем шаги 1 и 2 до тех пор, пока не получим единственное сообщение, вероятность которого равна 1.
4. Полученную последовательность действий отображаем в кодовое дерево. Для этого корневой точке присваиваем значение 1 (вероятность полного сообщения) и выбираем одну из ветвей исходящих из корневой точки. Она ведет к узлу, вероятность которого определяется одним из двух слагаемых, сумма которых составила результирующую вероятность 1. Из данного узла опять же исходят две ветви, заканчивающиеся узлами с вероятностями, сумма которых составила вероятность порождающего узла. Так, продвигаясь по дереву, в конце концов добираемся до конечного узла, вероятность которого определяется соответствующим данному узлу элементарным сообщением. Построение дерева продолжается до тех пор, пока не будут определены все конечные узлы, а дерево отображать весь заданный алфавит и соответствующие кодовые комбинации для каждого из сообщений источника.

Для наглядного понимания метода кодирования рассмотрим пример.

В таблице 3 приведен алфавит источника, состоящий из пяти сообщений различной вероятности.

**Таблица 3 - Произвольный алфавит источника сообщений**

$A$	$P(A)$
$a_1$	0,36
$a_2$	0,1
$a_3$	0,29
$a_4$	0,17
$a_5$	0,08

Рисунок 3 иллюстрирует алгоритм объединения сообщений источника в единое сообщение, вероятность которого равна 1. На рисунке 4 приведено дерево, отображающее код Хаффмена для выбранного источника.

**Внимание!** Поскольку нумерация ветвей дерева носит условный характер, то ради исключения недоразумений при выполнении домашнего задания придерживайтесь нумерации по аналогии с приведенным примером. Т.е. ветви идущей к узлу с большей вероятностью присваивайте номер «1», а идущей к узлу с меньшей вероятностью – «0».

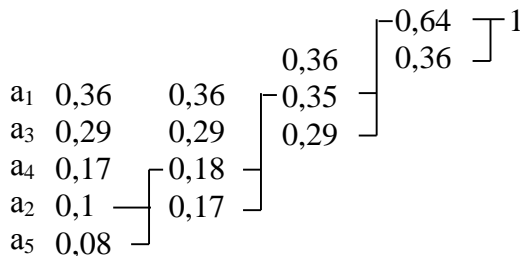


Рисунок 3 - Алгоритм кодирования по методу Хаффмена

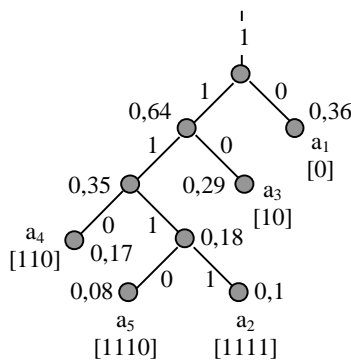


Рисунок 4 - Кодовое дерево простейшего алфавита

Анализ полученного кода показывает, что сообщение с максимальной вероятностью кодируется наименьшим количеством разрядов в двоичной системе (сообщение a<sub>1</sub> кодируется одним символом), напротив, сообщение с минимальной вероятностью кодируется наибольшим количеством разрядов (сообщения a<sub>2</sub> и a<sub>5</sub> кодируются четырьмя символами). В этом проявляется эффективность кода по сравнению с равномерным двоичным кодом, поскольку в среднем длительность передачи сообщения сокращается. Для количественной оценки эффективности неравномерных кодов, а полученный код как мы видим неравномерный, вводится коэффициент статистического сжатия, который показывает, во сколько раз отличается среднее число двоичных символов на сообщение при эффективном кодировании от среднего числа двоичных символов при равномерном кодировании. В математическом представлении это выглядит следующим образом:

$$K_{cc} = \frac{l_{p.k.}}{\bar{l}}, \tag{5.1}$$

где  $l_{p.k.}$  - средняя длина кодовой комбинации при равномерном кодировании.

Для представленного примера коэффициент статистического сжатия равен 1,38. Это говорит о том, что использование кодирования по методу Хаффмена для данного алфавита позволяет в 1,38 раза сократить число двоичных символов на сообщение в сравнении с равномерным кодированием. Нетрудно сделать выводы по времени передачи сообщения при упомянутых способах кодирования.

Эффективность кода в целом можно рассматривать и по теории Шеннона. При этом оценивается как близко среднее количество символов на одно сообщение к энтропии источника, т.е.:

$$K_{оэ} = \frac{H(A)}{\bar{l}}. \quad (5.2)$$

Приведенная количественная оценка называется коэффициентом относительной эффективности.

## 6 Передача сообщения в канале с помехами

Что происходит при декодировании кодовой последовательности, если при ее передаче в реальном канале связи произошла ошибка, обусловленная наличием различного рода помех? Рассмотрим это на примере.

Предположим, что в канале передается последовательность сообщений  $\{a_2, a_5, a_4, a_3\}$ , вероятности которых соответствуют таблице 3. Этой последовательности сообщений соответствует кодовая последовательность  $\{1111\ 1110\ 110\ 10\}$ , полученная кодированием по методу Хаффмена. Вследствие наличия помех во втором символе кодовой последовательности произошла инверсия и на выход канала поступила последовательность

$1\bar{0}11111011010.$

*Согласно применяемому нами коду приведенная кодовая последовательность должна восприниматься следующим образом:*

$10\ 1111\ 10\ 110\ 10.$

Декодер преобразует эту кодовую последовательность в следующую последовательность сообщений:

$a_3, a_2, a_3, a_4, a_3.$

Возникновение ошибки в одном символе кодовой последовательности привело к ошибочному декодированию восьми символов. Эффект, при котором ошибка в одном сообщении ведет к возникновению ошибок в других сообщениях передаваемой последовательности, называется треком ошибок, и он является существенным недостатком рассмотренного метода кодирования.

## ОПИСАНИЕ ЛАБОРАТОРНОГО МАКЕТА

Настоящая работа выполняется на персональном компьютере и эмулирует работу системы связи, построенной на равномерном и эффективном кодировании. Функциональная схема лабораторного макета представлена на рисунке 5.

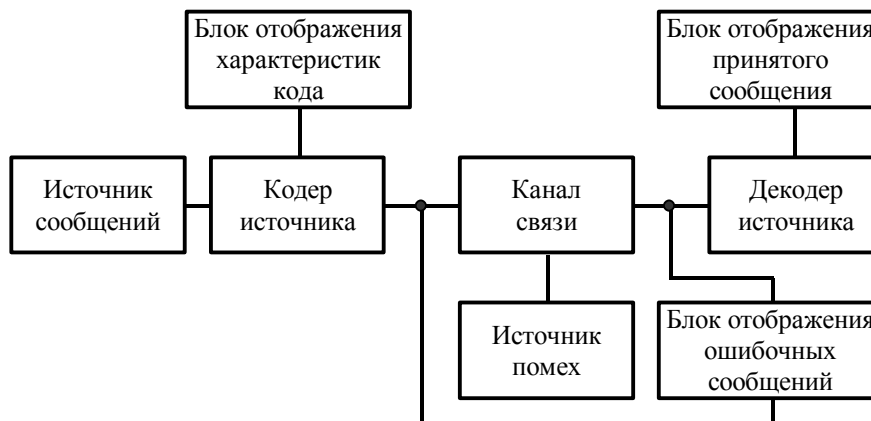


Рисунок 1 - Функциональная схема моделируемой системы связи

### 1 Назначение эмулируемых блоков

В лабораторном макете выделяются следующие блоки:

- источник сообщения – блок, позволяющий программным путем выбрать необходимый источник сообщения с простейшим алфавитом в соответствии с заданным вариантом из таблицы 1;

- кодер источника – эмулируется кодер, способный работать в двух режимах, а именно равномерном и эффективном кодировании; позволяет обрабатывать простейший алфавит и закодировать произвольное осмысленное сообщение с отображением кодовой последовательности; оценивает количество двоичных символов в закодированной последовательности;

- блок отображения характеристик кода – предназначен для анализа характеристик кода (равномерного или эффективного) в соответствии с заданным источником сообщений (таблица 1); отображает кодовую последовательность для каждого сообщения алфавита источника и основные характеристики:  $\bar{l}$ ,  $H(x)$ ,  $K_{03}$ ,  $K_{сс}$ ;

- канал связи – не несет программной нагрузки и недоступен пользователю;

- источник помех – отображает передаваемую кодовую последовательность и позволяет вносить в нее ошибку любой кратности;

- декодер источника – эмулирует работу декодера двух режимов (декодирование равномерного или эффективного кода);

- блок отображения принятого сообщения – в поле отображения содержит текст, полученный после декодирования принятой кодовой последовательности;

- блок отображения ошибочных сообщений – анализирует кодовые последовательности на входе и выходе канала связи и сообщает о достоверности или ложности передаваемого кода; ошибочные символы (в однозначном соответствии их определенному символу алфавита) копирует в поле отображения ошибочных сообщений.

Кроме описанных блоков основного макета в программе имеется окно «Тест-опрос», позволяющее проверить усвоение студентом теоретического материала, а также окно «Домашнее задание», направленное на контроль выполнения студентом домашнего задания.

## 2 Работа с программой

Работу с программой можно разделить на три основных этапа – короткий тест-опрос из десяти вопросов для контроля теоретической подготовки пользователя к выполнению лабораторной работы, оценка выполнения студентом домашнего задания, непосредственное выполнение лабораторной работы.

Первые два этапа не должны вызывать вопросов связанных с работой приложений, поскольку интерфейсы данных приложений построены по классической схеме – все наглядно и доступно для пользователя, привыкшего работать с операционной системой типа Windows 95 и более поздних.

Основным является окно лабораторного макета, обеспечивающее связь с блоками макета, описание которых приводится в пункте 1 настоящей главы. Связь с модальными окнами (интерфейсами блоков) осуществляется через системное меню основного окна опцией «действия», или всплывающим меню, вызов которого осуществляется щелчком правой кнопки мыши в поле макета основного окна. Предполагается, работа со всеми приложениями программы не вызовет особых затруднений.

# ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Перед проведением лабораторной работы ознакомьтесь с настоящим руководством по использованию программного продукта и назначением эмулируемых блоков.

## 1 Допуск к работе

*Если на используемой Вами машине не активизирована программа лабораторного макета, то загрузите ее из указанного преподавателем каталога. Если Вы видите окно «Тест» выполните следующие действия:*

- прочтите сообщение и выберите соответствующую Вашей подготовленности кнопку; если Вы выбрали кнопку «ОК» программа предлагает Вам вопросы, время на раздумье 1,5 мин., выбранный Вами ответ отметьте щелчком левой кнопки мыши;
- ознакомьтесь с резюме, которая дала машина на Ваш ответ, и нажмите кнопку «Далее»; программа представит Вам следующий вопрос;
- проделайте описанные действия для всех десяти вопросов;
- если Вы прошли удачно тест, нажмите кнопку «В добрый путь» и программа представит Вам окно «Домашнее задание»; Если Вы не справились с тестом – нажмите кнопку «Возврат в теорию», или... проконсультируйтесь с преподавателем;
- в раскрывающемся списке окна «Домашнее задание» выберите номер Вашего домашнего задания и нажмите кнопку «Подтверждение ввода»;

- если в поле текста написан номер, не соответствующий Вашему домашнему варианту, воспользуйтесь кнопкой «Изменить» на данном этапе, поскольку в последствии она будет не доступна;
- введите последовательно код сообщений, которые вы получили при выполнении домашнего задания, а также численные значения основных характеристик кода и источника сообщений ( $\bar{l}$ ,  $H(x)$ ,  $k_{оз}$ ,  $k_{сс}$ ); после каждого ввода пользуйтесь кнопкой «Подтверждение»; последовательность ввода должна соответствовать запросу программы (строка слева от окна ввода); при вводе численных значений отделяйте дробную часть числа от целой точкой, а не запятой;
- при успешном завершении теста программа активизирует окно основного макета и Вы приступаете к выполнению лабораторной работы, в противном случае Вам необходима консультация с преподавателем.

## 2 Практическая часть

*Перед выполнением работы откройте текстовый документ «отчет.doc». Все последующие действия с макетом отражаются в этом документе в виде таблиц и, обязательно, сохраняются под выбранным вами именем. После завершения работы преподаватель должен быть оповещен о местонахождении вашего отчета. Работа производится в следующем порядке.*

*2.1 Исследование характеристик кода при равномерном кодировании простейшего алфавита*

*Щелчком правой кнопки мыши в поле основного окна макета раскройте системное меню. В опции «Источник сообщения» выберете подменю «Простейший алфавит» и далее «Равномерное кодирование». Щелкните левой клавишей мыши по этой опции. Данные действия активизируют окно «Источник сообщений». Если в строке состояния написано «Равномерное кодирование», можно продолжать дальнейшее выполнение данного пункта. В противном случае закройте окно «источник сообщения» и повторите указанную последовательность действий.*

*Щелчком левой клавиши мыши выберете соответствующий Вашему домашнему заданию вариант и закройте окно «Источник сообщений» нажав левой клавишей мыши кнопку «Подтверждение».*

*Вызовите системное меню (правая клавиша мыши в поле макета) и выберете опцию «Характеристики кода» и далее «Отобразить». Произведите щелчок левой клавишей мыши по этой опции. Активизируется окно «Характеристики кода». Перепишите в таблицу 1 открытого Вами отчета отображаемый в окне «Характеристики кода» результат. При этом Вы заполните лишь по одной строке каждой части таблицы 1. оставшиеся строки предназначены для произвольно выбранного варианта. Поэтому...*

*Преподаватель обязан задать Вам произвольный номер варианта (отличный от вашего домашнего варианта). Вы же в свою очередь выполняете*



описанные выше действия, выбрав в окне «Источник сообщения» заданный вариант. На данном этапе таблица 1 в отчете должна быть полностью заполнена.

2.2 Исследование характеристик кода при эффективном кодировании простейшего алфавита

Проделайте последовательность действий аналогичную пункту 2.1 с той лишь разницей, что вызов окна «Источник сообщений» производится выбором опции «Эффективное кодирование» вместо прежде выбираемой опции «Равномерное кодирование». Контроль правильности выполнения – в строке состояния окна «Источник сообщения» должна иметься надпись «Эффективное кодирование».

Обработайте оба варианта и заполните таблицу 2 в отчете.

2.3 Исследование длительности кодовой последовательности при равномерном и эффективном кодировании простого сообщения

Вновь раскройте окно «Источник сообщения» как при равномерном кодировании. Выберите номер своего домашнего варианта и задайте с помощью раскрывающихся списков два наименее вероятных сообщения (вероятность сообщений даны в таблице 1 настоящего руководства). Закройте вызванное окно кнопкой «Подтверждение».

Посредством системного меню вызовите окно «Кодер источника» (щелчок левой клавиши мыши по соответствующей опции). В поле «Действия по работе с системой» щелчком левой клавиши мыши отметьте действие «Отобразить передаваемое сообщение». В текстовом окне «Исходная последовательность сообщений» должен появиться текст из пятикратного повторения заданных Вами сообщений. Теперь выберите действие «Отобразить код передаваемого сообщения» и наблюдайте в текстовом окне «Передаваемый по каналу код» закодированную последовательность Вашего сообщения.

Выберете действие «Оценить длительность кодовой последовательности», при этом в текстовой строке «Длительность кодовой комбинации для заданного сообщения равна:» отобразится количество бит необходимых для передачи заданного сообщения. Перепишите результат в таблицу 3 текстового документа «Отчет.doc», после чего выберете действие «Очистить дисплей».

Нажмите кнопку «Обратить источник сообщения» и в появившемся окне измените передаваемые сообщения на наиболее вероятные. Нажмите кнопку «Подтверждение» после чего проделайте те же действия, что и для наименее вероятных сообщений. В результате в таблице 3 должны быть заполнены две верхние строчки. Теперь нажмите кнопку «Вернуться к макету».

Повторите указанные действия с той лишь разницей, что первую активизацию окна «Источник сообщения» произвести через опцию «Эффективное кодирование». В результате всех действий таблица 3 в отчете должна быть полностью заполнена. Очистите дисплей кодера, но не закрывайте окна «Кодер источника сообщений».

#### 2.4 Декодирование блоковой последовательности содержащей ошибочные символы

Установите курсор в текстовом окне «Исходная последовательность сообщений» окна «Кодер источника сообщений» и введите произвольный текст с клавиатуры в режиме кириллицы. Поставьте флажок в окне «Кодер работает на равномерное кодирование» и выберите действие «Закодировать осмысленное сообщение». Нажмите кнопку «Перейти к декодеру».

Нажмите кнопку «Вкл./Выкл.» в окне «Декодер источника» после чего нажмите кнопку «Принять код». Поставьте флажок в окне «Декодировать как равномерный код» и нажмите кнопку «Декодировать». Теперь вызовите окно «Декодированный текст» нажатием кнопки «Отобразить текст» в окне декодера. Нажмите кнопку «Power» в появившемся окне и убедитесь в работоспособности системы, после чего закройте окно «Декодированный текст», а в окне «Декодер источника» нажмите кнопку «Вернуться к макету».

В системном меню макета выберите опцию «Кодер источника». При необходимости измените текст и внесите его в таблицу 4 отчета. Вновь закодируйте сообщение и нажмите кнопку «Вернуться к макету». В системном меню выберите опцию «Источник помех» и в ней опцию «Наличие помех». В появившемся окне нажмите кнопку «Внести ошибку». Во вновь открывшемся окне в строке ввода укажите номер символа, в котором Вы хотите произвести инверсию и нажмите кнопку «Внести ошибку». В таблице 4 отчета отметьте номер символа, в который Вы внесли ошибку. Если Вы желаете внести следующую ошибку, то нажмите кнопку «Повторить» и выполните те же действия, после чего нажмите кнопку «Подтверждение», а в таблице 4 укажите количество внесенных ошибок. В окне «Источник помех» нажмите кнопку «Вернуться к макету». С помощью системного меню вызовите окно «Декодер источника» и декодируйте текст как равномерно закодированный, выполнив действия описанные выше. Вызовите окно «Декодированный текст» и внесите в таблицу 4 принятый с ошибками текст. Закройте окно «Декодированный текст», а в окне «Декодер источника» нажмите кнопку «Вернуться к макету».

В системном меню выберите опцию «Отображение» и далее «Ошибки в коде». В появившемся окне нажмите кнопку «Начать проверку». После завершения проверки заполните таблицу 4 до конца и закройте окно «Проверка ошибочных символов» нажатием кнопки «Вернуться к макету».

### 2.5 Декодирование непрерывной последовательности содержащей ошибочные символы

*Выполните действия, аналогичные описанным в пункте 2.4 настоящего руководства с той лишь разницей, что в окне кодера необходимо снять флажок «Кодер работает на равномерное кодирование», а в окне декодера снять флажок «Декодировать как равномерный код». Задаваемый текст, а также количество и номера вносимых ошибок должны соответствовать тому тексту, тому количеству и тем номерам, которые Вы выбрали в пункте 2.4. В ходе выполнения Вами этого пункта должна быть заполнена таблица 5 в отчете.*

#### 2.6 Оформление отчета

Отчет должен содержать следующие сведения.

1. Цель работы
2. Результаты входного тестирования
3. Распечатка результирующих модульных окон.
4. Преподаватель может обозначить темы, на которые следует дать развернутый ответ.
5. Выводы по работе.

Выводы по работе должны включать:

3. Оценка уровня работы и ее полезности для изучения темы.
4. Замечания по усовершенствованию программы.