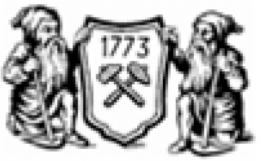
ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ



ПЕРВОЕ ВЫСШЕЕ ТЕХНИЧЕСКОЕ УЧЕБНОЕ ЗАВЕДЕНИЕ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ МИНЕРАЛЬНО-СЫРЬЕВОЙ УНИВЕРСИТЕТ «ГОРНЫЙ»»

Кафедра «Системного анализа и управления»

«Информационные сети и телекоммуникации»

методические указания и контрольные задания
для студентов очной и заочной формы обучения по направлению
220201.62 «Управление и информатика в технических системах»
220201.65 «Управление и информатика в технических системах»
220201.68 «Управление и информатика в технических системах»

Санкт - Петербург
2012 г.

УДК 681.5

Информационные сети и телекоммуникации: методические указания и контрольные задания для студентов очной и заочной формы обучения по направлению 220201.62, 220201.65, 220201.68 «Управление и информатика в технических системах» / Сост. А.Л. Ляшенко, Ю.В. Ильюшин Санкт-Петербургский горный ин-т, СПб., 20012. 45 с.

Утверждено:

Зав. каф. САиУ, _____ Д.А. Первухин

Рецензенты:
Першин И.М. (СКФУ) д.т.н., профессор;
Чернышев А.Б. (ПГГТУ) д.т.н., доцент;

В методических указаниях содержатся краткие теоретические сведения и курс лабораторных занятий по дисциплине «Информационные сети и телекоммуникации». Лабораторные работы, контрольные задания для студентов заочной формы обучения разработаны с учетом УМК для направлений высшего профессионального образования, бакалавриатата и магистратуры.

Содержание

Лабораторная работа №1. Основные понятия теории сети. Настройка DNS и TCP/IP

Цели:

Изучение настроек компьютера под ос windows для работы в сети Интернет.

- Использование пиктограмм и меню windows control panel для определения сетевых установок
- Использование утилит ipconfig, arp для определения сетевых установок компьютера
- Определение имени компьютера (computer name) и доменного имени (domain name)
- Определение производителя сетевого адаптера и его драйвера
- Идентификация используемого протокола сетевого уровня
- Определение сетевых ip-адресов
- Определение маски и ip-адреса шлюза по умолчанию (default gateway)
- Определение того, используется ли domain name system (dns) , dynamic host configuration protocol (dhcp) и windows internet name service (wins).
- Определение локального адреса компьютера (mac)
- Использование менеджера устройств windows для определения правильности работы сетевого адаптера.

Теоретические основы.

1. Каждый компьютер в сети интернет имеет адреса трех уровней:
   - Локальный адрес узла, определяемый технологией, с помощью которой построена отдельная сеть, в которую входит данный узел. Для узлов, входящих в локальные сети - это мас-адрес сетевого адаптера (network interface card- nic) или порта маршрутизатора, например, 11-a0-17-3d-bc-01. Эти адреса назначаются производителями оборудования и являются уникальными адресами, так как управляются централизовано. Для всех существующих технологий локальных сетей мас-адрес имеет формат 6 байтов: старшие 3 байта - идентификатор фирмы производителя, а младшие 3 байта назначаются уникальным образом самим производителем. Для узлов, входящих в глобальные сети, такие как x.25 или frame relay, локальный адрес назначается администратором глобальной сети.
   - Ip-адрес, состоящий из 4 байт, например, 109.26.17.100. Этот адрес используется на сетевом уровне. Он назначается администратором во время конфигурирования компьютеров и маршрутизаторов. Ip-адрес состоит из двух частей: номера сети и номера узла. Номер сети может быть выбран администратором произвольно, либо назначен по рекомендации специального подразделения internet (network information center, nic), если сеть должна работать как составная часть internet. Обычно провайдеры услуг internet получают диапазоны адресов у подразделений nic, а затем распределяют их между своими абонентами.
   - Номер узла в протоколе ip назначается независимо от локального адреса узла. Деление ip-адреса на поле номера сети и номера узла - гибкое, и граница между этими полями может устанавливаться весьма произвольно. Узел может входить в несколько ip-сетей. В этом случае узел должен иметь несколько ip-адресов, по числу сетевых связей. Таким образом ip-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение.
   - Символьный идентификатор-имя, например, serv1.ibm.com. Этот адрес назначается администратором и состоит из нескольких частей, например, имени машины, имени организации, имени домена. Такой адрес, называемый также dns-именем, используется на прикладном уровне, например, в протоколах ftp или telnet.
2. Ip-адрес состоит из двух логических частей - номера сети и номера узла в сети. Какая часть адреса относится к номеру сети, а какая к номеру узла, определяется значениями первых битов адреса:
   - Если адрес начинается с 0, то сеть относят к классу a, и номер сети занимает один байт, остальные 3 байта интерпретируются как номер узла в сети. Сети класса a имеют номера в диапазоне от 1 до 126. (номер 0 не используется, а номер 127 зарезервирован для специальных целей, о чем будет сказано ниже.) В сетях класса a количество узлов должно быть больше 216 , но не превышать 224.

— Если первые два бита адреса равны 10, то сеть относится к классу в и является сетью средних размеров с числом узлов 28 - 216. В сетях класса в под адрес сети и под адрес узла отводится по 16 битов, то есть по 2 байта.
— Если адрес начинается с последовательности 110, то это сеть класса с с числом узлов не больше 28. Под адрес сети отводится 24 бита, а под адрес узла - 8 битов.
— Если адрес начинается с последовательности 1110, то он является адресом класса d и обозначает особый, групповой адрес - multicast. Если в пакете в качестве адреса назначения указан адрес класса d, то такой пакет должны получить все узлы, которым присвоен данный адрес.
— Если адрес начинается с последовательности 11110, то это адрес класса e, он зарезервирован для будущих применений.

3. Количество централизовано выделенных администратору номеров сетей иногда недостаточно для того, чтобы структурировать сеть надлежащим образом, например, разместить все слабо взаимодействующие компьютеры по разным сетям. Одним из способов решения данной проблемы является использование так называемых *масок*, которые позволяют разделять одну сеть на несколько сетей. Маска - это число, двоичная запись которого содержит единицы в тех разрядах, которые должны интерпретироваться как номер сети. В масках, которые использует администратор для увеличения числа сетей, количество единиц в последовательности, определяющей границу номера сети, не обязательно должно быть кратным 8, чтобы повторять деление адреса на байты. Пусть, например, маска имеет значение 255.255.192.0 (11111111 11111111 11000000 00000000). И пусть сеть имеет номер 129.44.0.0 (10000001 00101100 00000000 00000000), из которого видно, что она относится к классу в. После наложения маски на этот адрес число разрядов, интерпретируемых как номер сети, увеличилось с 16 до 18, то есть администратор получил возможность использовать вместо одного, централизованно заданного ему номера сети, четыре:

129.44.0.0 (10000001 00101100 00000000 00000000)
129.44.64.0 (10000001 00101100 01000000 00000000)
129.44.128.0 (10000001 00101100 10000000 00000000)
129.44.192.0 (10000001 00101100 11000000 00000000)

Например, ip-адрес 129.44.141.15 (10000001 00101100 10001101 00001111), который по стандартам ip задает номер сети 129.44.0.0 и номер узла 0.0.141.15, теперь, при использовании маски, будет интерпретироваться как пара: 129.44.128.0 - номер сети, 0.0. 13.15 - номер узла. Таким образом, установив новое значение маски, можно заставить маршрутизатор по-другому интерпретировать ip-адрес. При этом два дополнительных последних бита номера сети часто интерпретируются как номера подсетей.

4. В интернете ip-адрес узла, то есть адрес компьютера или порта маршрутизатора (шлюза в интернет), назначается произвольно администратором сети и прямо не связан с его локальным адресом.. Локальный адрес используется только в пределах локальной сети. При обмене данными между маршрутизатором и узлом этой сети. Маршрутизатор, получив пакет для узла одной из сетей, непосредственно подключенных к его портам, должен для передачи пакета сформировать кадр в соответствии с требованиями принятой в этой сети технологии и указать в нем локальный адрес узла, например его мас-адрес. В пришедшем пакете этот адрес не указан, поэтому перед маршрутизатором встает задача поиска его по известному ip-адресу, который указан в пакете в качестве адреса назначения. Аналогичную задачу решает конечный узел, когда он хочет отправить пакет в удаленную сеть через маршрутизатор, подключенный к той же локальной сети, что и данный узел.

Для определения локального адреса по ip-адресу используется протокол разрешения адреса *address resolution protocol, arp*. Для решения обратной задачи используется протокол *reversed arp*.

5. Ip-адреса могут назначаться администратором сети вручную. Протокол *dynamic host configuration protocol (dhcp)* был разработан для того, чтобы освободить администратора от этих проблем. Основным назначением dhcp является динамическое назначение ip-адресов.

Средства/подготовка.

Для выполнения работы необходим компьютер под ос windows, в составе локальной сети (соответственно имеющий сетевой адаптер) и подключенный к сети интернет.

В ос windows используются:

- Утилиты ipconfig, arp
- Системные настройки windows папок «настройки/панель управления/сеть», «настройки/панель управления/менеджер устройств».

Ход работы.

Включите компьютер и введите свое имя и пароль для доступа к операционной системе.

Шаг 1 – определите сетевые настройки вашего компьютера с помощью программ «панель управления»/ «сеть и удаленный доступ к сети» и «мой компьютер»/ «свойства».

Вы должны узнать имя компьютера, имя домена, тип сетевого клиента, тип используемого протокола сетевого уровня, информацию относительно nic.

Заполните следующую таблицу 1.1

Табл. 1.1 – Таблица для заполнения значений

Имя netbios	
Имя в домене nt (win 2000)	
Тип сетевого клиента	
Имя установленного драйвера nic.	
Установленный сетевой протокол	

Шаг 2 – определите установки протокола tcp/ip, такие, как ip-адрес, dhcp и dns.

Заполните следующую таблицу 1.2.

Табл. 1.2 – Таблица для заполнения значений

Параметр	Вид информации	Значение
Ip адрес.	Каким образом компьютер получает свой ip адрес	
Ip адрес.	Ip адрес компьютера	
Ip адрес.	Маска подсети	
Шлюз	Шлюз по умолчанию	
Конф. Dns	Dns разрешен?	
Конф. Dns	Ip адрес сервера dns	
Конф. Wins	Wins разрешен?	
Конф. Wins	Ip адрес сервера wins	

Шаг 3 – используйте программу «панель управления/система/оборудование.диспетчер устройств/сетевые платы» для проверки работоспособности и определения параметров nic.

Заполните следующую таблицу 1.3

Табл. 1.3 – Таблица для заполнения значений

Изготовитель nic	
Nic работает без ошибок?	
Дата драйвера nic	
Список файлов драйвера	

Шаг 4 – использование утилиты ipconfig.exe

Изучите опции команды ipconfig, выполнив команду ipconfig /?

Задача:

запустите программу ipconfig.exe /all из окна cmd (окна dos).

Заполните следующую таблицу 1.4.

Табл. 1.4 – Таблица для заполнения показателей

Ip адрес компьютера	
Маска подсети	

Мас адрес компьютера	
Шлюз по умолчанию (default gateway)	
Сервер dhcp	
Ip адрес сервера dns	
Ip адрес сервера wins	

Шаг 5 – использование утилиты arp.exe
Введите команду arp /? И изучите ее формат, операнды и параметры.

Задача:
выполните команды arp -a, arp –a inet_addr, arp –a eth_addr и запишите результаты.

Шаг 7 – отчет по работе
Покажите результаты работы преподавателю.
Отправьте файл с результатами преподавателю по электронной почте. Укажите в нем ваше имя, группу, время и место выполнения работы.

Задание к лабораторной работе.
выполнить анализ прохождения ip-пакетов до адресов с помощью команд ping, tracert, netstat.
Команда ping служит для принудительного вызова ответа конкретной машины в сети, позволяет проверить функционирование основных узлов (элементов) сети. В выполнении команды ping принимает участие система маршрутизации, схемы разрешения адресов (firewall) и сетевые шлюзы, поэтому для получения положительного результата сеть должна быть в рабочем состоянии. Формат команды:

```
Ping [-t] [-a] [-n число] [-l размер] [-f] [-i ttl] [-v tos] [-r число] [-s число]
    [[-j списокузлов] | [-k списокузлов]] [-w таймаут] списокрассылки
Параметры:
-t              отправка пакетов на указанный узел до команды прерывания.
                для вывода статистики и продолжения нажмите
                (ctrl)+(break), для прекращения - (ctrl)+(c).
-a              определение адресов по именам узлов.
-n число        число отправляемых запросов.
-l размер       размер буфера отправки.
-f              установка флага, запрещающего фрагментацию пакета.
-i ttl          задание срока жизни пакета (поле "time to live").
-v tos          задание типа службы (поле "type of service").
-r число        запись маршрута для указанного числа переходов.
-s число        штамп времени для указанного числа переходов.
-j списокузлов  свободный выбор маршрута по списку узлов.
-k списокузлов  жесткий выбор маршрута по списку узлов.
-w таймаут      таймаут каждого ответа в миллисекундах.
```

пример выполнения команды ping :

```
Ping www.dgtu.donetsk.ua
Обмен пакетами с w3.donntu.edu [194.44.183.9] по 32 байт:
Ответ от 194.44.183.9: число байт=32 время<1мс  ttl=254
Ответ от 194.44.183.9: число байт=32 время<10мс ttl=254
Ответ от 194.44.183.9: число байт=32 время<10мс ttl=254
Ответ от 194.44.183.9: число байт=32 время<10мс ttl=254
Статистика ping для 194.44.183.9:
   пакетов: отправлено = 4, получено = 4, потеряно = 0 (0% потерь),
Приблизительное время передачи и приема:
   наименьшее = 0мс, наибольшее = 1мс, среднее = 0мс
```

```
Ping k16c5
Обмен пакетами с k16c5  [192.168.18.250] по 32 байт:
Ответ от 192.168.18.250: число байт=32 время<1мс  ttl=128
```

```
Ответ от 192.168.18.250: число байт=32 время<10мс ttl=128
Ответ от 192.168.18.250: число байт=32 время<10мс ttl=128
Ответ от 192.168.18.250: число байт=32 время<10мс ttl=128
Статистика ping для 192.168.18.250:
   пакетов: отправлено = 4, получено = 4, потеряно = 0 (0% потерь),
Приблизительное время передачи и приема:
   наименьшее = 0мс, наибольшее = 1мс, среднее = 0мс
```

```
Ping uran.donetsk.ua
Обмен пакетами с uran. Donetsk.ua[194.44.183.1] по 32 байт:
Ответ от 194.44.183.1: число байт=32 время=7мс ttl=252
Ответ от 194.44.183.1: число байт=32 время=4мс ttl=252
Ответ от 194.44.183.1: число байт=32 время=4мс ttl=252
Ответ от 194.44.183.1: число байт=32 время=4мс ttl=252
Статистика ping для 194.44.183.1:
   пакетов: отправлено = 4, получено = 4, потеряно = 0 (0% потерь),
Приблизительное время передачи и приема:
   наименьшее = 4мс, наибольшее = 7мс, среднее = 4мс
```

перейти на начало лабораторной работы # 2

Команда tracert позволяет определить последовательность шлюзов, через которую проходит ip-пакет на пути до пункта своего назначения. Возвращаемой информацией команды является список машин, начиная с первого шлюза и заканчивая пунктом назначения. Формат команды :

```
Tracert [-d] [-h максчисло] [-j списокузлов] [-w интервал] имя
Параметры:
   -d              без разрешения в имена узлов.
   -h максчисло    максимальное число прыжков при поиске узла.
   -j списокузлов  свободный выбор маршрута по списку узлов.
   -w интервал     интервал ожидания каждого ответа в миллисекундах.
```

пример выполнения команды tracert :

```
Tracert www.dgtu.donetsk.ua
Трассировка маршрута к www.dgtu.donetsk.ua [194.44.183.9]
С максимальным числом переходов 30:
  1   <10 мс   <10 мс   <10 мс  gw0if2.lan4.donntu [194.168.18.254]
  2    4 мс     1 мс    10 мс  dgtu.donetsk.ua [194.44.183.9]
Трассировка завершена.
```

```
Tracert uran.donetsk.ua
Трассировка маршрута к uran.donetsk.ua [194.44.183.1]
С максимальным числом переходов 30:
  1   <10 мс   <10 мс   <10 мс  gw0if2.lan4.donntu [192.168.18.254]
  2    4 мс     4 мс    5 мс  uran.donetsk.ua [194.44.183.1]
Трассировка завершена.
```

```
Tracert cs.dgtu.donetsk.ua
Трассировка маршрута к www.dgtu.donetsk.ua [194.44.183.213]
С максимальным числом переходов 30:
  1   <10 мс   <1 мс   <10 мс  gw0if2.lan4.donntu [194.168.18.254]
  2    1 мс     1 мс    1 мс    dgtu.donetsk.ua [194.44.183.213]
Трассировка завершена.
```

перейти на начало лабораторной работы # 2

Еще одним средством изучения состояния сети есть команда netstat. Она пердназначена для отображения статистики протокола и текущего сетевого подключения tcp/ip. Формат команды :

```
Netstat [-a] [-e] [-n] [-s] [-p имя] [-r] [интервал]
  -a          отображение всех подключений и ожидающих портов.
              (подключения со стороны сервера обычно не отображаются).
  -e          отображение статистики ethernet. Этот ключ может
```

```
                применяться вместе с ключом -s.
        -n          отображение адресов и номеров портов в числовом формате.
        -p имя      отображение подключений для протокола "имя": tcp или udp.
                    используется вместе с ключом -s для отображения статистики
                    по протоколам. Допустимые значения "имя": tcp, udp или ip.
        -r          отображение содержимого таблицы маршрутов.
        -s          отображение статистики по протоколам. По умолчанию выводятся
                    данные для tcp, udp и ip. Ключ -p позволяет указать
                    подмножество выводящихся данных.
        интервал    повторный вывод статистических данных через указанный
                    интервал в секундах. Для прекращения вывода данных
                    нажмите клавиши ctrl+c. Если параметр не задан, сведения
                    о текущей конфигурации выводятся один раз.
```

пример выполнения команды netstat :

```
Netstat
Активные подключения

  имя   локальный адрес      внешний адрес                 состояние
  tcp   k16c4:1074           host12.list.ru:9999           syn_sent
  tcp   k16c4:nbsession      dhcp_dyn_ip_242.lan4.donntu:1030   established
  tcp   k16c4:nbsession      dhcp_dyn_ip_243.lan4.donntu:1073   established
  tcp   k16c4:nbsession      dhcp_dyn_ip_243.lan4.donntu:1144   established
  tcp   k16c4:nbsession      dhcp_dyn_ip_250.lan4.donntu:1026   established
```

```
Netstat -a
Активные подключения
  имя   локальный адрес    внешний адрес        состояние
  tcp   k16c3:1033         k16c3:0              listening
  tcp   k16c3:1025         k16c3:0              listening
  tcp   k16c3:1026         k16c3:0              listening
  tcp   k16c3:137          k16c3:0              listening
  tcp   k16c3:138          k16c3:0              listening
  tcp   k16c3:nbsession    k16c3:0              listening
  tcp   k16c3:1025         *:*
  tcp   k16c3:1026         *:*
  tcp   k16c3:nbname       *:*
  tcp   k16c3:nbdatagram   *:*
```

```
Netstat -e
Статистика интерфейса
                     получено          отправлено

Байт                    262891             23508
Одноадресные пакеты        155               137
Многоадресные пакеты      1405                55
Отброшено                    0                 0
Ошибки                       0                 0
Неизвестный протокол       302
```

Лабораторная работа № 2. Команды диагностики сети

Цель работы:

— Изучить принципы работы простейших средств мониторинга сети.
— Получить навыки решения задач, связанных с мониторингом сети.

Теоретические основы.

1. Протокол icmp

Протокол icmp (интернет-протокол контрольных сообщений) стека протоколов tcp/ip предназначен для передачи между сетевыми устройствами сообщений об ошибках и контрольных сообщений при помощи ip-пакетов.

В протоколе icmp определены несколько типов сообщений, в том числе см. таблицу 2.1.

Табл. 2.1 – Несколько типов сообщений в протоколе

Destination unreachable	Time to live exceeded	Parameter problem
Source quench	Redirect	Echo
Echo reply	Timestamp	Timestamp reply
Information request	Information reply	Address request
Address reply		

Например, если маршрутизатор получает пакет, который он не может доставить по указанному в нем адресу, отправителю передается icmp-сообщение о недостижимости адреса (destination unreachable).

2. Ping: проверка соединения с определенным интерфейсом.

Программа ping использует протокол icmp.

Эта команда посылает пакет эхо-запроса на другой ip-адрес и ожидает ответа. Она чаще всего используется для того, чтобы посмотреть, «жив ли» другой компьютер. Ответ на запрос содержит также данные о том, как долго пакет путешествовал до адресата. Можно использовать команду ping с различными опциями: число посланных пакетов (от1 до 10), время жизни пакета (time to live –ttl, от 1 до 255ms), размер пакета (от 16 до 8192 байт), время ожидания (timeout, до 9999 ms) и разрешать или нет фрагментацию каждого пакета.

Формат команды в ос windows:

Ping [-t] [-a] [-n count] [-l size] [-f] [-i ttl] [-v tos]
[-r count] [-s count] [[-j host-list] | [-k host-list]]
[-w timeout] destination-list

Options:

-t	выполнение команды до прерывания (ctrl+c)
-a	разрешать адреса в имена
-n count	число отправляемых пакетов.
-l size	размер буфера отправки
-f	установить флаг "не фрагментировать".
-i ttl	установить время жизни.
-w timeout	время ожидания ответа в мс.
-v tos	задание типа службы (поле "type of service").
-r count	запись маршрута для указанного числа переходов.
-s count	штамп времени для указанного числа переходов.
-j host-list	свободный выбор маршрута по списку узлов.
-k host-list	жесткий выбор маршрута по списку узлов.
Destination-list	список рассылки.

3. Программа tracert. Определение промежуточных сетевых интерфейсов между хостами. Трассировка маршрута.

Программа трассировки маршрута использует протокол icmp.

Эта утилита очень похожа на ping, за исключением того, что она показывает все другие ip-адреса (интерфейсы), которые пакет проходит до своего места назначения. Дополнительно можно изменять различные опции, ассоциированные с trace route: максимальное число дозволяемых промежуточных узлов (maximum hops, от 1 до 255) и timeout (до 9999 ms).

Формат команды в ос windows:

Tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout] target_name

Options:

-d	не разрешать адреса в имена.

-h maximum_hops наибольшее число промежуточных узлов.
-j host-list трассировка через определенный список хостов
-w timeout время ожидания каждого ответа в мс.

4. Программа netstat. Сетевая статистика.

Программа **netstat** используется для просмотра активных соединений каждого протокола, таблиц маршрутизации, а так же детализирует статистику передачи данных.

Формат команды в ос windows:

Netstat [-a] [-e] [-n] [-s] [-p имя] [-r] [интервал]

-a отображение всех подключений и ожидающих портов.
(подключения со стороны сервера обычно не отображаются).

-e отображение статистики ethernet. Этот ключ может применяться вместе

С ключом -s.

-n отображение адресов и номеров портов в числовом формате.

-p имя отображение подключений для протокола "имя": tcp или udp.

Используется вместе с ключом -s для отображения статистики по протоколам. Допустимые значения "имя": tcp, udp или ip.

-r отображение содержимого таблицы маршрутов.

-s отображение статистики по протоколам. По умолчанию выводятся данные для tcp, udp и ip. Ключ -p позволяет указать подмножество выводящихся данных.

Интервал повторный вывод статистических данных через указанный интервал в секундах. Для прекращения вывода данных нажмите клавиши ctrl+c. Если параметр не задан, сведения о текущей конфигурации выводятся один раз.

Инструментарий.

В работе будут использоваться сети, соединенные с рабочей станцией линками разных типов. Первая сеть (194.85.33.0) доступна через спутниковый канал связи, проходящий через спутниковый ретранслятор на геостационарной орбите с пропускной способностью 512 кбит/с. Вторая сеть 217.23.64.0 доступна через наземный волоконно-оптический канал связи с пропускной способностью 2 мбит/с, третья сеть 212.194.38.0 входит в состав корпоративной сети вуза и доступна в локальной сети с пропускной способностью 10/100 мбит/с.

Ход работы.

Задание 1. Ping.

Выполните команду ping в командной строке с различными значениями параметров -t, -n, -l, -i,-w. Какие наблюдения и выводы вы сделали?

Ping www.sgu.ru
Ping www.microsoft.com
Ping www.sun.com
Ping 212.193.38.83

Выполните ping к тем же хостам с параметром –f, увеличивая параметр -l size. При каком значении размера перестают получаться ответы?

Задание 2. Tracert.

Выполните команду tracert в командной строке с различными значениями параметров. Какие наблюдения и выводы вы сделали?

Используйте, например,
Ping www.sgu.ru
Ping www.microsoft.com
Ping www.sun.com
Ping 212.193.38.83

Задание 3. Поисковые сервисы европейского и российского ip-регистров.

Определите, кому принадлежат сети 194.85.33.0, 217.23.64.0, 212.193.38.0. Для этого используйте поисковые аппараты http://www.ripe.net/db/whois/whois.html и http://www.ripn.net:8080/nic/whois/index.html. Пользуясь данными этих информационных систем,

попробуйте определите географическое расположение сетей. Попробуйте изобразить топологическую схему соединения этих сетей. Поместите ее на рис. 2.1.

Рис.2.1 –Для заполнения топологии сети

Задание 4. Использование программы ping для исследования параметров сети.

1. Приведите сравнительные результаты выполнения команд ping по адресам 194.85.33.29, 194.85.33.30, 217.23.64.2, 212.193.38.248, 212.193.35.10 по параметрам «время отклика», ttl в форме таблицы. Объясните полученные различия.
2. Соберите средние времена прохождения 10 пакетов на указанные адреса. Сравните с р4езультатами, полученными при использовании сервиса ping в интерфейсе looking glass на сайте http://noc.runnet.ru. Объясните полученные различия.
3. Соберите усредненные времена прохождения 10 пакетов увеличивающегося размера по указанным адресам. Начните с 64 байт и каждый раз удваивайте размер пакета. При каком размере пакета потери превышают 50 %. Как влияет время ожидания отклика на процент прохождения пакетов этого размера. При каком времени ожидания отклика потери для пакетов зафиксированного размера не возникают?
4. Представьте результаты измерений в форме таблиц, наилучшим образом проявляющим, по вашему мнению, обнаруженные зависимости.
5. Используя программу ping, оцените вклад разных сетевых участков, по которым проходит эхо-пакет между вашей рабочей станцией и интерфейсом 194.85.33.29.

Задание 5. Использование программы tracert для анализа соединений в сети.

1. Приведите сравнительные результаты выполнения команд tracert по адресам 194.85.33.29, 194.85.33.30, 217.23.64.2, 212.193.38.248, 212.193.35.10. Объясните полученные различия.
2. Выполните трассировку к адресу 212.193.38.248 и к адресу 217.23.64.2 со стороны сайта http://noc.runnet.ru. Приведите полученные результаты.
3. Используя данные, полученные в результате выполнения трассировки и отправки эхо-пакетов между интерфейсами 212.193.38.248 и 194.85.35.100, оцените вклад разных участков сетей, соединяющих эти интерфейсы, в среднее время прохождения пакетов между ними.
4. Используя полученную в ходе выполнения всех заданий информацию, уточните схему задания 1, изобразите на ней обнаруженные вами промежуточные интерфейсы и линки сети, объединяющей подсети 194.85.33.0, 217.23.64.0, 212.193.38.0.

Лабораторная работа № 3. Топология сети WWW

Цель работы:

Целью работы является изучение вопросов конфигурации сетей ethernet

Теоретические основы.

Наибольшее распространение среди локальных вычислительных сетей получила сеть ethernet (стандарт ieee 802.3). Стандарт определяет множественный доступ к моноканалу типа “шина” с обнаружением конфликтов и контролем передачи (по-русски мдкн/ок - метод доступа с контролем несущей и обнаружением коллизий (столкновений), по-английски csma/cd – carrier-sense multiple access/collision detection). Основные характеристики стандарта ieee 802.3 следующие: топология – “шина”, скорость передачи – 10 мбит/с, метод доступа - csma/cd, передача узкополосная (моноканал). Передача идет пакетами переменной длины. Предусмотрена индивидуальная, групповая и широковещательная адресация.

Помимо стандартной топологии типа “шина” применяются также топологии типа “пассивная звезда” и “дерево”. При этом предполагается использование репитеров и пассивных (репитерных) концентраторов, соединяющих между собой различные части (сегменты) сети (рис. 3.1). В качестве сегмента может выступать единичный абонент. Главное – чтобы в полученной в результате топологии не было замкнутых путей (петель). Фактически получается, что абоненты соединены все в ту же “шину”, так как сигнал от каждого из них распространяется сразу во все стороны и не возвращается назад.

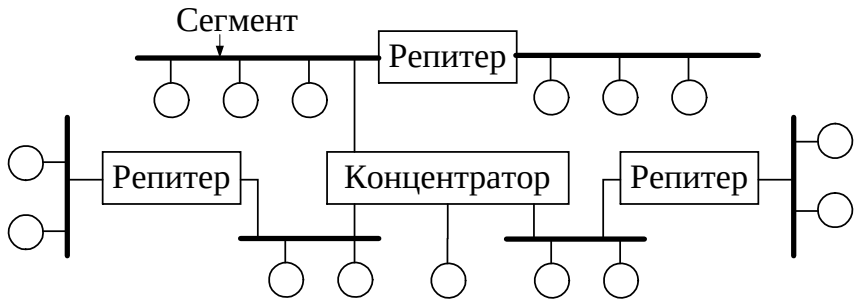


Рис. 3.1 – Сегменты сети.

Для сети ethernet стандарт определяет четыре основных типа среды передачи:

— 10base5 (“толстый” коаксиальный кабель);
— 10base2 (“тонкий” коаксиальный кабель);
— 10base-t (витая пара);
— 10base-f (оптоволоконный кабель).

Обозначение среды передачи включает в себя три элемента: цифра “10” означает скорость передачи 10 мбит/с, слово base означает передачу в основной полосе частот (т.е. Без модуляции высокочастотного сигнала), а последний элемент означает допустимую длину сегмента: “5” – 500 метров, “2” – 200 метров (точнее, 185 метров) или тип линии связи: “t” – витая пара (от английского “twisted-pair”, “f” – оптоволокно (от английского “fiber optic”).

Аппаратура 10base5 (“толстый” кабель)

Аппаратные средства 10base5 представлены на рис. 3.2, а схема подсоединения адаптера к “толстому” кабелю – на рис. 3.3. “толстый” коаксиальный кабель имеет диаметр 0,5 дюйма (около 1 см) и отличается высокой жесткостью, что приводит к большим трудностям монтажа аппаратуры. Волновое сопротивление “толстого” коаксиального кабеля – 50 ом. Максимальная длина сегмента – 500 метров (без репитеров). Широко распространены “толстые” кабели типа rg-8 и rg-11.

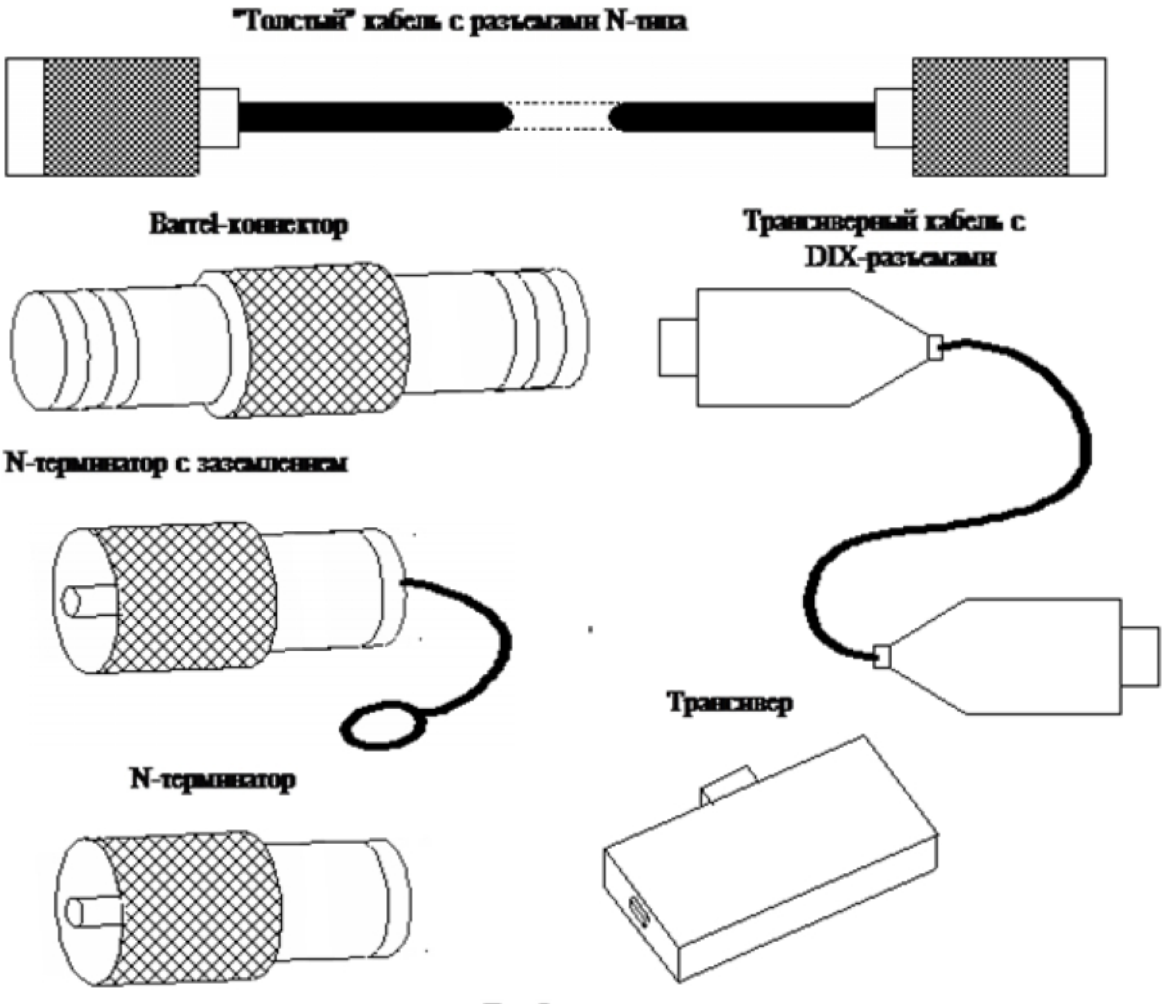


Рис.3.2 – Аппаратные средства 10base5.

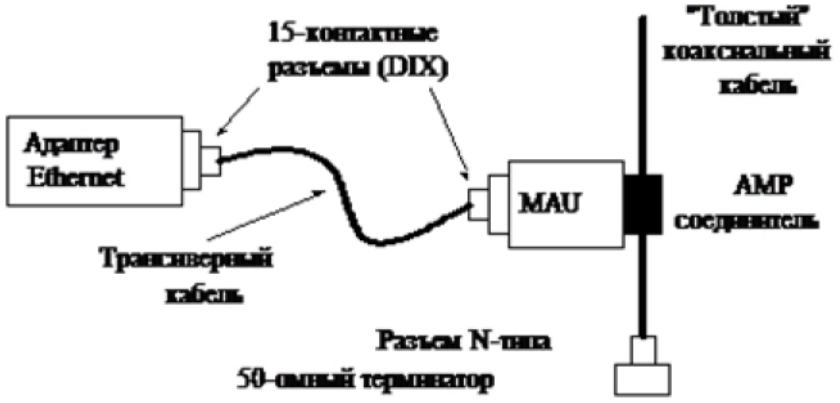


Рис.3.3 - Схема подсоединения адаптера к “толстому” кабелю.

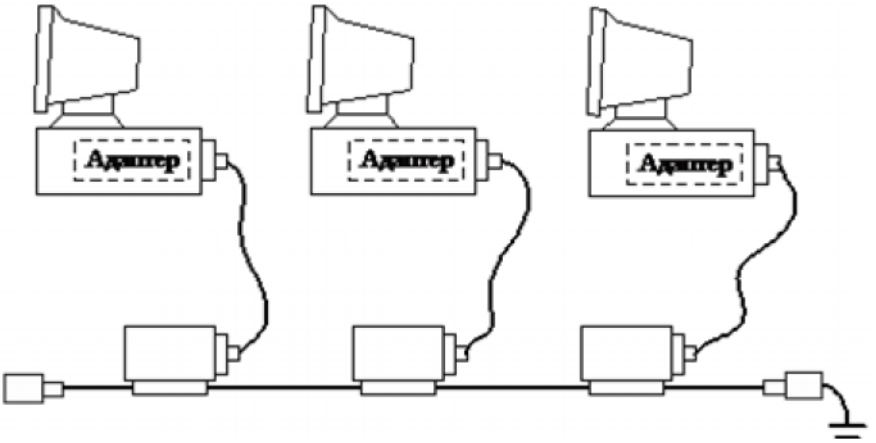


Рис. 3.4 – Пример соединения.

Для соединения кусков “толстого” коаксиального кабеля и присоединения к нему терминаторов используются разъемы n-типа. Два разъема n-типа соединяются с помощью barrel-коннекторов. На концах кабеля сегмента должны быть установлены 50-омные терминаторы n-типа, один из которых надо заземлить. Для присоединения трансиверов к “толстому” кабелю чаще всего используют amp соединитель (рис. 3.4).

Непосредственно на кабеле размещается специальный трансивер (или mau – medium attachment unit), присоединяемый к сетевому адаптеру с помощью гибкого многопроводного трансиверного кабеля aui (диаметром около 1 см), состоящего из 4 витых пар, имеющего на концах 15-контактные разъемы (dix-разъемы типа “вилка”). Длина обычного трансиверного кабеля может достигать 50 м, а более тонкого и гибкого офисного варианта трансиверного кабеля – до 12,5 м. Трансивер питается от источника питания компьютера.

Трансивер (transmitter+receiver=transceiver – приемопередатчик) – это часть сетевого адаптера, которая выполняет следующие функции:

— Прием и передачу данных с кабеля на кабель;
— Определение коллизий на кабеле;
— Электрическая развязка между кабелем и остальной частью адаптера;
— Защита кабеля от некорректной работы адаптера.

Допускается подключение к одному сегменту не более 100 трансиверов, причем расстояние между подключениями трансиверов не должно быть меньше 2,5 м.

Схема соединения компьютеров сегмента сети на “толстом” кабеле показана на рис. 4.

Сетевой адаптер, работающий с “толстым” кабелем, должен иметь внешний 15-контактный aui-разъем (разъем dix типа “розетка”).

Стандарт разрешает использование в сети не более 4 репитеров (репитерных концентраторов) и, соответственно, не более 5 сегментов кабеля. Это дает максимальную длину сети 10base5 равную 2500 метров. Только 3 сегмента из 5 могут быть нагруженными, то есть такими, к которым подключаются компьютеры. Между нагруженными сегментами должны быть ненагруженные сегменты, так что максимальная конфигурация сети представляет собой два нагруженных крайних сегмента, которые соединяются ненагруженными сегментами еще с одним центральным нагруженным сегментом.

Правило применения репитеров (репитерных концентраторов) в сети ethernet 10base5 носит название “правило 5-4-3”: 5 сегментов, 4 репитера (репитерных концентратора), 3 нагруженных сегмента.

Каждый репитер (репитерный концентратор) подключается к сегменту одним своим трансивером, поэтому к нагруженным сегментам можно подключить не более 99 компьютеров. Максимальное количество компьютеров в сети 10base5 составляет 99·3=297 компьютеров.

Минимальный набор оборудования для односегментной сети на “толстом” кабеле включает в себя следующие элементы:

— Сетевые адаптеры (по числу объединяемых компьютеров);
— “толстый” кабель с разъемами n-типа на концах, общая длина которого достаточна для объединения всех компьютеров сети;
— Трансиверные кабели с 15-контактными разъемами на концах длиной от компьютера до

“толстого” кабеля (по количеству сетевых адаптеров);

— Трансиверы (по количеству сетевых адаптеров);
— Два barrel-коннектора n-типа для присоединения терминаторов на концах кабеля;
— Один n-терминатор без заземления;
— Один n-терминатор с заземлением.

Аппаратура 10base2 (“тонкий” кабель).

“Тонкий” коаксиальный кабель отличается от “толстого” меньшей толщиной - диаметр около 0,5 дюйма (5 мм), большей гибкостью, большим удобством монтажа, меньшей стоимостью. “тонкий” кабель имеет волновое сопротивление 50 ом и требует 50-омного оконечного согласования. Максимальная длина сегмента – 185 метров (без репитеров).

Самым большим недостатком “тонкого” кабеля является меньшая допустимая длина сегмента (до 185 м). Наиболее распространенные типы “тонкого” коаксиального кабеля – это rg-58 /u, rg-58 a/u, rg-58 c/u.

Аппаратные средства 10base2 представлены на рис.3.5, а схема подсоединения адаптера к “тонкому” кабелю – на рис. 3.6.

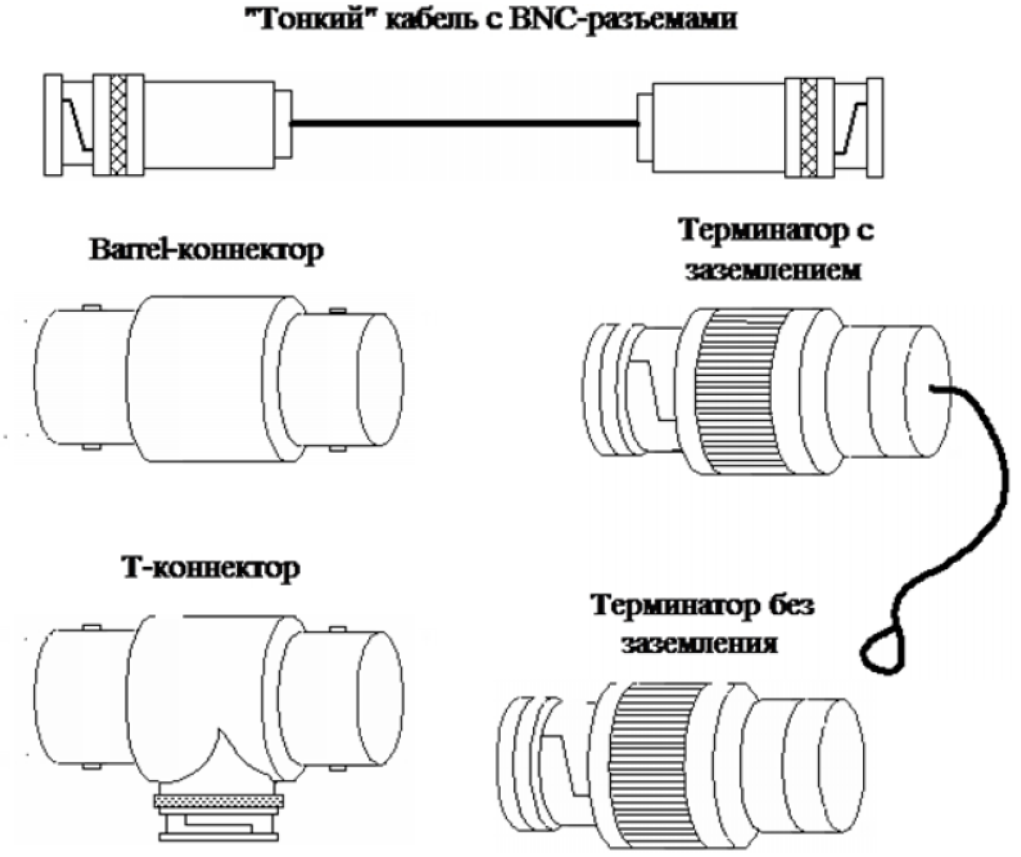


Рис. 3.5 - Аппаратные средства 10base2

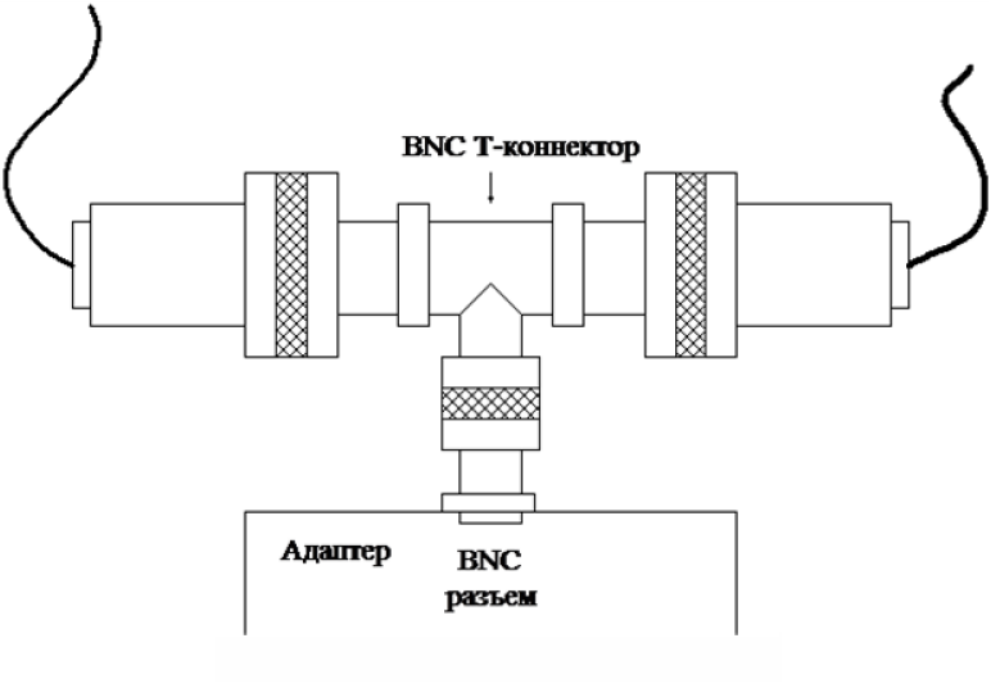


Рис. 3.6 - Схема подсоединения адаптера.

Если вся сеть выполняется на “тонком” кабеле, то, согласно стандарту, количество сегментов не должно превышать пяти (общая длина сети составит 925 м, потребуется четыре репитера). При этом на одном сегменте не должно быть больше 30 абонентов, включая репитеры, то есть общее число компьютеров в сети на базе “тонкого” кабеля не может быть больше (30-1)·3=87. Минимальное расстояние между компьютерами – 1 м.

Стандарт 10base2 предусматривает использование репитеров (репитерных концентраторов), применение которых также должно соответствовать “правилу 5-4-3”.

Минимальный набор оборудования для односегментной сети на “тонком” кабеле должен включать в себя следующие элементы:

- сетевые адаптеры (по числу объединяемых в сеть компьютеров);
- отрезки кабеля с bnc-разъемами на двух концах, общая длина которых достаточна для объединения всех компьютеров;
- Bnc t-коннекторы (по числу сетевых адаптеров);
- один bnc терминатор без заземления;
- один bnc терминатор с заземлением.

Аппаратура 10bas-t (витая пара).

В сети ethernet на базе витой пары (utp-кабели, unshielded twisted-pair cable) передача сигналов осуществляется по двум витым парам проводов, каждая из которых передается только в одну сторону (одна пара – передающая, другая – принимающая). Каждый из абонентов сети присоединяется кабелем к концентратору, использование которого обязательно.

Длина соединительного кабеля между адаптером и концентратором не должна превышать 100 м. Кабель используется гибкий, диаметром около 6 мм. Наиболее распространенный тип кабеля – телефонный кабель eia/tia категории 3.

Кабели присоединяются 8-контактными разъемами типа rj-45, в которых используются только четыре контакта. В концентраторах иногда применяются также 50-контактные разъемы типа telco.

В стандарте определено максимальное число концентраторов между двумя станциями сети, а именно 4. Это правило носит название “правило 4-х хабов”. При создании сети 10bas-t с большим числом станций концентраторы можно соединять друг с другом иерархическим способом, образуя древовидную структуру.

Петлевидное соединение концентраторов в стандарте 10base-t запрещено. Резервирование связей (создание параллельных каналов связи между важными концентраторами для резервирования связей на случай отказа порта, концентратора или кабеля) возможно только за счет перевода одной из параллельных связей в неактивное (заблокированное) состояние.

Общее количество компьютеров в сети 10base-t – 1024, максимальная длина сети (максимальное расстояние между двумя компьютерами сети) –

500 м.

Минимальный набор оборудования для сети на витой паре включает в себя следующие элементы:

- сетевые адаптеры (по числу объединяемых в сеть компьютеров), имеющие разъемы rj-45;
- Отрезки кабеля с разъемами rj-45 на концах (по числу объединяемых компьютеров);
- один концентратор, имеющий столько utp-портов, сколько необходимо объединить компьютеров.

Аппаратура 10base-fl (оптоволоконный кабель).

Применение оптоволоконного кабеля в ethernet помимо обеспечения полной гальванической развязки компьютеров сети, позволило увеличить длину сегмента и существенно повысить помехоустойчивость передачи.

Передача информации идет по двум оптоволоконным кабелям, передающим сигналы в разные стороны, показаны на рис. 3.7.

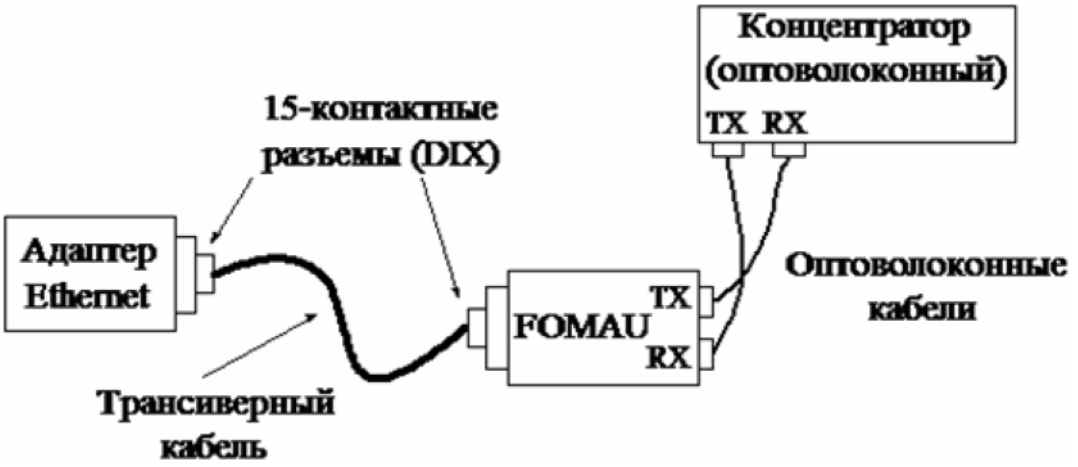


Рис. 3.7 – Пример передачи информации по двум оптоволоконным кабелям.

Стандарт 10base fl обеспечивает связь между двумя компьютерами, между двумя репитерами или между компьютером и репитером. Стандарт гарантирует длину оптоволоконной связи между репитерами (репитерными повторителями) до 1 км при общей длине сети не более 2500 м. Максимальное расстояние между компьютером и концентратором – 2000 м. Максимальное число репитеров (репитерных концентраторов) между любыми компьютерами сети – 4. Максимальная длина оптоволоконного кабеля 10base-fl, соединяющего репитерные концентраторы (репитеры) с компьютерами, не должно превышать 400 метров. Ко всем сегментам могут подключаться компьютеры.

Аппаратура 10base-fl имеет сходство как с аппаратурой 10base5 (применяются внешние трансиверы соединенные с адаптером трансиверным кабелем), так и с аппаратурой 10base-t (применяются топологии типа “пассивная звезда” и два разнонаправленных кабеля). Схема соединения сетевого адаптера и концентратора показана на рис. 7. Минимальный набор оборудования для соединения оптоволоконным кабелем двух компьютеров включает в себя следующие элементы:

- Два сетевых адаптера с трансиверными разъемами;
- Два оптоволоконных трансивера (fomau);
- Два трансиверных кабеля;
- Два оптоволоконных кабеля с st-разъемами на концах.

Выбор конфигурации Ethernet.

Соблюдение многочисленных ограничений, установленных для различных стандартов физического уровня сетей ethernet, гарантирует корректную работу сети. Правила “5-4-3” для коаксиальных сетей и “4-х хабов” для сетей на основе витой пары и оптоволокна не только дают

гарантии работоспособности сети, но и оставляют большой “запас прочности” сети. Для сетей, состоящих из смешанных кабельных систем, на которые правила о количестве повторителей не рассчитаны, необходимо проводить дополнительные расчеты.

Чтобы сеть ethernet, состоящая из сегментов различной физической природы, работала корректно, необходимо выполнение четырех основных условий:

- Количество компьютеров в сети не более 1024;
- Максимальная длина каждого физического сегмента не более величины, определенной в соответствующем стандарте физического уровня;
- Время двойного оборота сигнала между двумя самыми удаленными друг от друга компьютерами сети не более 575 битовых интервала;
- Сокращение межкадрового интервала при прохождении последовательности кадров через все повторители должно быть не больше, чем 49 битовых интервала.

Соблюдение этих требований обеспечивает корректность работы сети даже в случаях, когда нарушаются простые правила конфигурирования, определяющие максимальное количество повторителей и общую длину сети в 2500 м.

Расчет времени двойного оборота сигнала.

Модель, применяемая для оценки конфигурации ethernet, основана на подсчете временных характеристик данной конфигурации. В ней применяется две системы расчетов: одна предполагает вычисление двойного (кругового) времени прохождения сигнала по сети, а другая – проверку допустимости получаемого (межкадрового) временного интервала. При этом расчеты в обеих системах расчетов ведутся для наихудшего случая.

При первой системе расчетов используются такие понятия, как “начальный сегмент”, “промежуточный сегмент” и “конечный сегмент”. Отметим, что промежуточных сегментов может быть несколько, а начальный и конечный сегменты при разных расчетах могут меняться местами. Для расчетов используются величины задержек, представленные в таблице 3.1.

Табл. 3.1 – Величины задержек для расчета.

Тип Сегмента ethernet	Макс. Длина, м	Начальный сегмент		Промежуточный сегмент		Конечный сегмент		Задержка на метр длины
		T0	Tm	T0	Tm	T0	Tm	T1
10base5	500	11,8	55,0	46,5	89,8	169,5	212,8	0,0866
10base2	185	11,8	30,8	46,5	65,5	169,5	188,5	0,1026
10base-t	100	15,3	26,6	42,0	53,3	165,0	176,3	0,1130
10base-fl	2000	12,3	212,3	33,5	233,5	156,5	356,5	0,1000
Foirl	1000	7,8	107,8	29,0	129,0	152,0	252,0	0,1000
Aui (> 2 м)	2+48=50	0	5,1	0	5,1	0	5,1	0,1026

Примечание.

Задержки даны в битовых интервалах. Расчет сводится к следующему:

1. В сети выделяется путь наибольшей длины;
2. Если длина сегмента не максимальна, то рассчитывается двойное (круговое) время прохождения в каждом сегменте выделенного пути по формуле: $ts=l \cdot t1+t0$, где l – длина сегмента в метрах (при этом надо учитывать тип сегмента: начальный, промежуточный или конечный);
3. Если длина сегмента максимальна, то из таблицы для него берется величина задержки tm;
4. Суммарная величина задержек всех сегментов выделенного пути не должна превышать 575 битовых интервалов;
5. Затем необходимо проделать те же действия для обратного направления выбранного пути (то есть, считая конечный сегмент начальным, и наоборот);
6. Если задержки в обоих случаях не превышают 575 битовых интервалов, то сеть работоспособна.

Если в выбранной вами конфигурации сети путь наибольшей длины не столь очевиден, то подобные расчеты необходимо произвести для всех путей, претендующих на наибольшую задержку сигнала. В любом случае двойное время прохождения в соответствии со стандартом недостаточно,

чтобы сделать окончательный вывод о работоспособности сети.

Расчет сокращения межкадрового интервала.

Чтобы признать конфигурацию сети корректной, нужно рассчитать также уменьшение межкадрового интервала репитерами (репитерными концентраторами).

Эта величина не должна быть меньше, чем 49 битовых интервалов. Для вычислений здесь также используются понятия начального сегмента и промежуточного сегмента (конечный сегмент не вносит вклада в сокращение межкадрового интервала, так как пакет доходит по нему до принимающего компьютера без прохождения репитеров и репитерных концентраторов).

Для расчета сокращения межкадрового интервала можно воспользоваться значениями максимальных величин уменьшения межкадрового интервала при прохождении репитеров (репитерных концентраторов) различных физических сред приведенными в таблице 3.2.

Табл. 3.2 – Разные физические среды

Тип сегмента	Начальный сегмент	Промежуточный сегмент
10base5	16	11
10base2	16	11
10base-t	10,5	8
10base-fl	10,5	8

Вычисления здесь очень простые. Суммируя величины сокращений межкадрового интервала для наибольшего пути в выбранной конфигурации и сравнивая сумму с предельной величиной в 49 битовых интервалов, мы можем сделать вывод о работоспособности сети.

Такие же вычисления проводятся и для обратного направления по этому же пути.

Порядок выполнения работы.

1. Ознакомиться с теоретической частью к лабораторной работе.
2. В соответствии с заданным вариантом спроектируйте локальную вычислительную сеть организации (приложение а).
3. Подготовьте спецификацию на оборудование и материалы спроектированной локальной вычислительной сети организации (приложение б).

Требования к отчету.

Отчет по лабораторной работе должен содержать:

1. титульный лист;
2. задание;
3. конфигурацию спроектированной сети;
4. программу расчетов, подтверждающих работоспособность сети (программа должна выполнять расчеты для любой конфигурации сети);
5. программу подготовки спецификации на оборудование и материалы (программа должна выполнять расчеты для любой конфигурации сети);
6. результаты проектирования показать преподавателю на экране монитора.

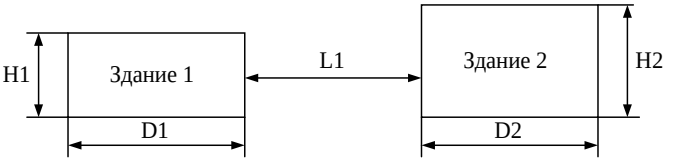
Контрольные вопросы.

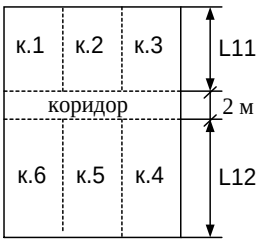
1. Среды передачи для сети ethernet?
2. Аппаратура 10base5?
3. Аппаратура 10base2?
4. Аппаратура 10base-t?
5. Аппаратура 10base-fl?
6. Выбор конфигурации ethernet?

Список литературы.

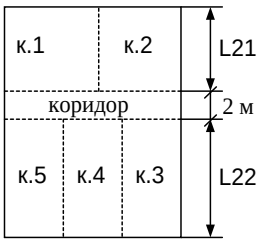
1. Новиков ю.в., карпенко д.г. Аппаратура локальных сетей: функции, выбор, разработка/ под общей редакцией ю.в. Новикова. – м., издательство эком, 1998. – 288с.: ил.
2. Компьютерные сети. Принципы, технологии, протоколы/ в.г. Олифер, н.а. Олифер. – спб: издательство “питер”, 2000. – 672 с.: ил.

Приложение а





Экспликация комнат на этаже



Экспликация комнат на этаже

Вари-ант	L1, М	H1, М	D1, М	L11,М	L12,М	H2, М	D2, М	L21,М	L22, М	Этажность здания 1	Этажность здания 2
1.	Max	9	60	15	30	8	150	30	15	3	2
2.	Max	6	75	20	25	12	120	25	20	2	3
3.	Max	9	90	25	20	8	90	20	25	3	2
4.	Max	6	120	30	15	12	60	15	30	2	3

Вариант	Здание	Этаж	Количество компьютеров					
			К.1	К.2	К.3	К.4	К.5	К.6
1.	1	1	1	2	1	2	1	3
		2	3	1	2	1	2	1
		3	1	3	1	2	1	2
	2	1	2	1	3	1	2	1
		2	2	3	1	2	2	-
2.	1	1	3	1	2	1	2	1
		2	1	3	1	2	1	2
	2	1	2	1	3	1	3	-
		2	2	3	1	2	2	-
		3	4	2	1	2	1	-
3.	1	1	3	1	2	1	2	1
		2	1	2	1	2	1	3
		3	2	1	2	1	3	1
	2	1	3	1	3	1	2	-
		2	1	2	1	2	4	
4.	1	1	1	3	1	2	1	2
		2	3	1	2	1	2	1
	2	1	3	1	2	3	1	-
		2	4	1	2	1	2	-
		3	3	3	1	2	1	-

Вариант	Здание	Этаж	Тип среды Передачи	Тип среды передачи между зданиями
1.	1	1	10base5	10base5
		2	10base2	
		3	10base-t	
	2	1	10base-fl	

		2	10base5	
2.	1	1	10base2	10base2
		2	10base-t	
	2	1	10base-fl	
		2	10base5	
		3	10base2	
3.	1	1	10base-t	10base-t
		2	10base-fl	
		3	10base5	
	2	1	10base2	
		2	10base-t	
4.	1	1	10base-fl	10base-fl
		2	10base5	
	2	1	10base2	
		2	10base-t	
		3	10base-fl	

Примечание.
Можно применять репитеры и репитерные концентраторы на 4, 8, 12 портов.

Приложение б

№№	Наименование	Единица Измерения	Количество
Оборудование			
1.	Репитер	Шт.	
2.	Репитерный концентратор на 4 порта	Шт.	
3.	Репитерный концентратор на 8 портов	Шт.	
4.	Репитерный концентратор на 12 портов		
5.			
6.			
7.			
Материалы			
1.	“толстый” коаксиальный кабель	М	
2.	“тонкий” коаксиальный кабель	М	
3.	Utp-кабель категории 3	М	
4.	Оптический кабель	М	
5.			
6.			
7.			

Лабораторная работа № 4. Конфигурации сети fast ethernet

Цель работы:
Целью работы является изучение вопросов конфигурации сетей fast ethernet

Теоретические основы.
Сеть fast ethernet – это составная часть стандарта ieee 802.3. Она представляет собой более быструю версию стандарта ethernet, использующую метод доступа csma/cd (carrier-sense multiple access/collision detection) - метод доступа с контролем несущей и обнаружением коллизий (столкновений) и работающий на скорости передачи 100 мбит/с. В fast ethernet сохранен формат кадра принятый в классической версии ethernet.

Основная топология сети fast ethernet – “пассивная звезда”. Fast ethernet требует обязательного применения концентраторов. Концентраторы могут объединяться между собой связными

сегментами, что позволяет строить сложные конфигурации. Стандарт определяет три типа среды передачи для fast ethernet:

- 100base-t4 (передача идет со скоростью 100 мбит/с в основной полосе частот по четырем витым парам электрических проводов);
- 100base-tx (передача идет со скоростью 100 мбит/с в основной полосе частот по двум витым парам электрических проводов);
- 100base-f4 (передача идет со скоростью 100 мбит/с в основной полосе частот по двум оптоволоконным кабелям).

Для присоединения сетевого адаптера к сетевому кабелю в сети fast ethernet иногда используются специальные трансиверы, ориентированные на какой-то один тип кабеля. В этом случае применяемый сетевой адаптер не зависит от типа среды передачи, что повышает гибкость системы. Трансивер при этом подключается к адаптеру трансиверным кабелем длиной 0,5 м, оснащенным 40-контактным разъемом. Однако гораздо чаще сетевой адаптер ориентируется изготовителем на какой-то один неизменяемый тип передачи, и трансивер при этом уже не требуется, так как сетевой кабель подключается непосредственно к адаптеру. Адаптер в данном случае оснащен соответствующим кабелю разъемом. Стандарт определяет два типа (класса) репитеров (концентраторов) для fast ethernet:

- репитеры класса i характеризуются тем, что они преобразуют приходящие по сегментам сигналы в цифровую форму прежде чем передавать их во все другие сегменты. Поэтому к ним можно подсоединять сегменты разных типов: 100base-tx, 100base-t4 и 100base-fx. Но процесс преобразования требует временной задержки, поэтому можно использовать только один репитер клас-са i в пределах одной зоны конфликта;
- репитеры класса ii непосредственно повторяют приходящие на них сигналы и передают их в другие сегменты без преобразования. Поэтому к ним можно подключаться только сегменты одного типа (например, 100base-tx) или сегменты, использующие одну систему сигналов (например, 100base-tx и 100base-fx). Задержка в репитерах класса ii меньше, чем в репитерах класса i, поэтому можно применять два таких репитера в пределах одной зоны конфликта.

Аппаратура 100base-tx.

Схема объединения компьютеров в сеть 100base-tx практически ничем не отличается от схемы 10base-t. Для присоединения неэкранированных кабелей, содержащих две витые пары (волновое сопротивление 100 ом) используются 8-контактные разъемы типа rj-45 категории 5. Длина кабеля не может превышать 100 метров. Также используется топология типа “пассивная звезда” с концентратором в центре. Только сетевые адаптеры должны быть fast ethernet, концентратор рассчитан на подключение сегментов 100base-tx, и кабель должен быть категории 5. Между адаптерами и сетевыми кабелями могут включаться трансиверы. Предельная длина 100 м в fast ethernet определяется заданными временными соотношениями обмена (ограничение на двойное время прохождения). Стандарт рекомендует ограничиваться длиной сегмента в 90 м, чтобы иметь 10% запас. Из восьми контактов разъема используется только 4 контакта: два для передачи и два для приема. Стандарт предусматривает также возможность применения экранированного сетевого кабеля с двумя витыми парами (волновое сопротивление – 150 ом). В этом случае применяется 9-контактный разъем d-типа.

Аппаратура 100base-t4.

Основное отличие аппаратуры 100base-t4 от 100base-tx состоит в том, что в качестве соединительных кабелей в ней используются неэкранированные кабели, содержащие четыре витые пары (кабели категории 3, 4 или 5). Схема объединения компьютеров в сеть ничем не отличается от 100base-tx. Длина кабелей не может превышать 100 м (стандарт рекомендует ограничиваться 90 м для 10 % запаса). Между адаптерами и кабелями в случае необходимости могут включаться трансиверы. Для подключения сетевого кабеля к адаптеру (трансиверу) используются 8-контактные разъемы типа rj-45, соответствующей категории. Обмен данными идет по одной передающей витой паре, по одной приемной витой паре и по двум двунаправленным витым парам с использованием дифференциальных сигналов.

Аппаратура 100base-fx.

Аппаратура 100base-fx очень близка к аппаратуре 10base-fl. Точно также здесь используется топология типа “пассивная звезда” с подключением компьютеров к концентратору с помощью двух разнонаправленных оптоволоконных кабелей. Между сетевыми адаптерами и кабелями возможно включение трансиверов. Оптоволоконные кабели подключаются к адаптеру (трансиверу) с помощью разъемов типа sc, st.

Максимальная длина кабеля между компьютером и концентратором составляет 412 метров, причем это ограничение определяется временными соотношениями.

Выбор конфигурации fast Ethernet.

Для определения работоспособности сети fast ethernet стандарт ieee 802.3 предлагает две модели, называемые transmission system model 1 и transmission system model 2. При этом первая модель основана на несложных правилах, а вторая использует систему расчетов. В соответствии с первой моделью, при выборе конфигурации надо руководствоваться следующими принципами:

— сегменты, выполненные на электрических кабелях (витая пара), не должны быть длиннее 100 м;
— сегменты, выполненные на оптоволоконных кабелях, не должны быть длиннее 412 м;
— если используются трансиверы, то трансиверные кабели не должны быть длиннее 50 см.

При выполнении этих правил надо руководствоваться таблицей 4.1, определяющей максимальные размеры (в метрах) зоны конфликта (т.е. Максимальное расстояние между абонентами сети, не разделенными коммутаторами). При этом в двух последних столбцах таблицы, относящихся к случаю использования смешанных сред передачи (как витых пар, так и оптоволоконных кабелей), предполагается, что длина витой пары составляет 100 м, применяется только один оптоволоконный кабель. Первая строка относится к соединению двух компьютеров без применения репитера. Нереализуемые ситуации отмечены в таблице прочерками.

Табл. 4.1 – Значения определяющие максимальные размеры зоны конфликта

Тип репитера (концентратора)	Витая пара	Оптоволоконный кабель	T4 и fx	Tx и fx
Без репитера (два абонента)	100	412	-	-
Один репитер Класса i	200	272	231	260,8
Один репитер Класса ii	200	320	-	308,8
Два репитера Класса ii	205	228	-	216,2

Вторая модель основана на вычислениях суммарного двойного времени прохождения сигнала по сети.

Для расчетов в соответствии со второй моделью сначала надо выделить в сети путь с максимальным двойным временем прохождения и максимальным числом репитеров (концентраторов) между компьютерами. Если таких путей несколько, то расчет должен производиться для каждого из них. Расчет в данном случае ведется на основании таблицы 4.2.

Табл.4.2 – Пример расчета

Тип сегмента	Задержка на метр (битовый интервал)	Максимальная задержка (битовый интервал)
Два абонента tx/fx	-	100
Два абонента t4	-	138
Один абонент t4 И один tx/fx	-	127
Сегмент на кабеле категории 3	1,14	114 (100 м)
Сегмент на кабеле категории 4	1,14	114 (100 м)
Сегмент на кабеле категории 5	1,112	111,2 (100 м)
Экранированная витая пара	1,112	111,2 (100 м)
Оптоволоконный кабель	1,0	412 (412 м)
Репитер (концентратор) Класса i	-	140
Репитер (концентратор) Класса ii с портами tx/fx	-	92

Репитер (концентратор) Класса ii с портами t4	-	67

Для вычисления полного двойного (кругового) времени прохождения для сегмента сети необходимо умножить длину сегмента на величину задержки на метр, взятую из второго столбца таблицы 2. Если сегмент имеет максимально возможную длину, то можно взять величину максимальной задержки для данного сегмента из третьего столбца таблицы. Затем задержки сегментов, входящих в путь максимальной длины, надо просуммировать и прибавить к этой сумме величину задержки для двух абонентов (три верхние строчки таблицы) и величины задержек для всех репитеров (концентраторов), входящих в данный путь. Суммарная задержка должна быть меньше, чем 512 битовых интервалов.

Задержки в кабеле могут отличаться от тех, которые приведены в таблице 2. Для более точного расчета следует использовать временные характеристики конкретного кабеля, применяемого в сети. Производители кабелей иногда указывают величину задержки на метр длины, а иногда – скорость распространения сигнала относительно скорости света (или nvp – nominal velocity of propagation). Связанны эти две величины формулой: $tз=1/(3\cdot10^{10}\cdot nvp)$, где tз - величина задержки на метр кабеля. Например, если nvp=0,4 (40%) от скорости света, то задержка tз будет равна 8,34 нс/м или 0,834 битовых интервала. Для вычисления двойного (кругового) времени прохождения нужно удвоенное значение tз умножить на длину кабеля. В таблице 4.3 даны величины nvp для некоторых типов кабелей.

Табл. 4.3 – Величины для некоторых типов кабелей

Фирма	Марка	Категория	Nvp
At&t	1010	3	0,67
At&t	1041	4	0,70
At&t	1061	5	0,70
At&t	2010	3	0,70
At&t	2041	4	0,75
At&t	2061	5	0,75
Belden	1229a	3	0,69
Belden	1455a	4	0,72
Belden	1583a	5	0,72
Belden	1245a2	3	0,69
Belden	1457a	4	0,75
Belden	1585a	5	0,75

Для некоторых репитеров и концентраторов изготовители указывают меньшие величины задержек, чем приведенные в таблице 2, что также надо учитывать при выборе конфигурации сети.

Порядок выполнения работы.

1. Ознакомиться с теоретической частью к лабораторной работе.
2. В соответствии с заданным вариантом спроектируйте локальную вычислительную сеть организации (приложение а).
3. Подготовьте спецификацию на оборудование и материалы спроектированной локальной вычислительной сети организации (приложение б).

Требования к отчету.

Отчет по лабораторной работе должен содержать:

1. титульный лист;
2. задание;
3. конфигурацию спроектированной сети;
4. программу расчетов, подтверждающих работоспособность сети (программа должна выполнять расчеты для любой конфигурации сети);
5. программу подготовки спецификации на оборудование и материалы (программа должна выполнять расчеты для любой конфигурации сети);
6. результаты проектирования показать преподавателю на экране монитора.

Контрольные вопросы.

1. Среды передачи для сети fast ethernet?
2. Аппаратура 100base-t4?
3. Аппаратура 100base-tx?
4. Аппаратура 100base-fx?
5. Выбор конфигурации fast ethernet (первая модель)?
6. Выбор конфигурации fast ethernet (вторая модель)?

Список литературы.

1. Новиков ю.в., карпенко д.г. Аппаратура локальных сетей: функции, выбор, разработка/ под общей редакцией ю.в. Новикова. – м., издательство эком, 1998. – 288с.: ил.
2. Компьютерные сети. Принципы, технологии, протоколы/ в.г. Олифер, н.а. Олифер. – спб: издательство “питер”, 2000. – 672 с.: ил.
3. Куин л., рассел р. Fast ethernet. –к.: издательская группа bnv, 1998. – 448 с.

Приложение а

Вариант	L1, M	H1, M	D1, M	L11,M	L12,M	H2,M	D2,M	L21,M	L22,M	Этаж. здания 1	Этаж. здания 2
1.	Max	9	60	15	30	8	150	30	15	3	2
2.	Max	6	75	20	25	12	120	25	20	2	3
3.	Max	9	90	25	20	8	90	20	25	3	2
4.	Max	6	120	30	15	12	60	15	30	2	3

Вариант	Здание	Этаж	Количество компьютеров					
			K.1	K.2	K.3	K.4	K.5	K.6
1.	1	1	1	2	1	2	1	3
		2	3	1	2	1	2	1
		3	1	3	1	2	1	2
	2	1	2	1	3	1	2	1
		2	2	3	1	2	2	-
2.	1	1	3	1	2	1	2	1
		2	1	3	1	2	1	2
	2	1	2	1	3	1	3	-
		2	2	3	1	2	2	-
		3	4	2	1	2	1	-
3.	1	1	3	1	2	1	2	1
		2	1	2	1	2	1	3
		3	2	1	2	1	3	1
	2	1	3	1	3	1	2	-
		2	1	2	1	2	4	
4.	1	1	1	3	1	2	1	2
		2	3	1	2	1	2	1
	2	1	3	1	2	3	1	-
		2	4	1	2	1	2	-
		3	3	3	1	2	1	-

Вариант	Здание	Этаж	Тип среды Передачи	Тип среды передачи между зданиями
1.	1	1	100base-t4 (кабель at&t 1010)	100base-fx
		2	100base-tx (кабель at&t 1061)	
		3	100base-fx	

	2	1	100base-t4 (кабель at&t 1041)	
		2	100base-tx (кабель at&t 2061)	
2.	1	1	100base-tx (кабель belden 1583a)	100base-t4 (кабель belden 1229a)
		2	100base-fx	
	2	1	100base-tx (кабель belden 1585a)	
		2	100base-fx	
		3	100base-t4 (кабель belden 1455a)	
3.	1	1	100base-fx	100base-tx (кабель at&t 2061)
		2	100base-t4 (кабель at&t 2041)	
		3	100base-tx (кабель at&t 1061)	
	2	1	100base-fx	
		2	100base-t4 (кабель at&t 2061)	
4.	1	1	100base-t4 (кабель belden 1455a)	100base-fx
		2	100base-tx (кабель belden 1583a)	
	2	1	100base-fx	
		2	100base-tx (кабель belden 1585a)	
		3	100base-fx	

Примечание.
Можно применять концентраторы класса i, класса ii на 8, 12, 16, 24 порта.

Приложение б

№	Наименование	Единица Измерения	Количество
Оборудование			
1.	Концентратор i класса на 8 портов	Шт.	
2.	Концентратор i класса на 12 портов	Шт.	
3.	Концентратор i класса на 16 портов	Шт.	
4.	Концентратор i класса на 24 порта	Шт.	
5.	Концентратор ii класса на 8 портов	Шт.	
6.	Концентратор ii класса на 12 портов	Шт.	
7.	Концентратор ii класса на 16 портов	Шт.	
8.	Концентратор ii класса на 24 порта	Шт.	
Материалы			
1.	Utp-кабель категории 3	М	
2.	Utp-кабель категории 4	М	
3.	Utp-кабель категории 5	М	
4.	Stp-кабель категории 5	М	
5.	Оптический кабель	М	
6.	Вилка rj-45	Шт.	
7.	Розетка rj-45	Шт.	

Лабораторная работа № 5. Router. Управление потоками данных в локальной сети.

Теоретические основы.

Маршрут (route) – список элементов сети связи (узлов коммутации ук, линий связи, каналов связи), начинающийся с узла-источника (уи) и заканчивающийся узлом-получателем (уп).

1. Маршруты между ук №1 и ук №4 на сети, изображенной на рисунке 5.1, будут иметь следующую запись:

$M^1_{1,4}$ = {ук № 1, ук № 4};

$M^2_{1,4}$ = {ук № 1, ук № 2, ук № 4};

$M^3_{1,4}$ = {ук № 1, ук № 2, ук № 3, ук № 4}.

В данном примере ук № 1 является исходящим, ук № 4 – входящим, а ук № 2, 3 – транзитными.

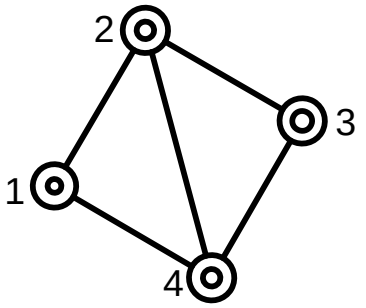


Рисунок 5.1 – Пример структуры сети связи

Маршрутизация (routing) – процедура, определяющая оптимальный по заданным параметрам маршрут на сети связи между узлами коммутации.

Для реализации маршрутизации на сети в каждом транзитном ук (ук № j), начиная с уи, формируется таблица *маршрутизации*, которая представляет собой матрицу размерностью (s – 1) x h_j

$$M^{(j)} = \left\| m_{i,v}^{(j)} \right\|_{(S-1),Hj} = \left(\overline{m_1^{(j)}}, ..., \overline{m_i^{(j)}}, ..., \overline{m_{j-1}^{(j)}}, \overline{m_{j+1}^{(j)}}, ..., \overline{m_S^{(j)}} \right), \qquad (5.1)$$

$$\overline{m_i^{(j)}} = \left(m_1^{(j)}, ..., m_{iv}^{(j)}, ..., m_{iHj}^{(j)} \right);\ v = \overline{1,H};\ i, j = \overline{1,S};\ i \neq j, \qquad (5.2)$$

Где s – количество ук в сети; h_j – количество исходящих линий связи (лс) из j-го ук.

матрица $m^{(j)}$ содержит информацию о предпочтительности выбора исходящей лс из j-го ук при поиске маршрута к i-му узлу (уп).

первый элемент $m_{i1}{}^{(j)}$ вектор-строки (5.2) указывает номер исходящей лс из j-го ук к смежному ук, которую предпочтительнее выбрать для организации маршрута к i-му ук (уп).

второй элемент (5.2) указывает номер следующей исходящей лс из j-го ук к другому смежному ук, которая менее предпочтительна для организации искомого маршрута. И так до hj-го элемента вектор-строки (5.2).

в данном случае: $m_{i1}{}^{(j)}$ – исходящая лс h_j-го выбора.

2. Построим таблицу маршрутизации для ук № 2 (рисунок 5.1). Соответствующие строки матрицы m(2) будут иметь следующий вид:

$$\overline{m_1^{(2)}} = (1,4,3);\ \overline{m_3^{(2)}} = (3,4,1);\ \overline{m_4^{(2)}} = (4,1,3). \qquad (5.3)$$

при поиске маршрута от ук № 2 к ук № 1 необходимо обратиться к вектор-строке $\overline{m_1^{(2)}} = (1,4,3)$. исходящая лс к ук № 1 является более предпочтительной, так как она ведет непосредственно к искомому ук, следовательно, является исходящей лс первого выбора. Соответственно, исходящие лс к ук №4 и 3 являются исходящими лс второго и третьего выбора. Для того, чтобы была возможность определять маршруты между любой парой ук необходимо построить таблицы маршрутизации в каждом узле сети. Совокупность таблиц маршрутизации для всех ук называется *планом распределения информации* (при) на сети связи.

3. Зададим при на сети, изображенной на рисунке 5.1:

$$M^{(1)} = \begin{array}{c|cc} 2 & 2 & 4 \\ 3 & 2 & 4 \\ 4 & 4 & 2 \end{array}; \quad M^{(2)} = \begin{array}{c|ccc} 1 & 1 & 4 & 3 \\ 3 & 3 & 4 & 1 \\ 4 & 4 & 1 & 3 \end{array};$$

$$M^{(3)} = \begin{array}{c|cc} 1 & 4 & 2 \\ 2 & 2 & 4 \\ 4 & 4 & 2 \end{array}; \quad M^{(4)} = \begin{array}{c|ccc} 1 & 1 & 2 & 3 \\ 2 & 2 & 3 & 1 \\ 3 & 3 & 2 & 1 \end{array}.$$

в данном примере формирование при осуществлялось по минимальному количеству транзитных ук в искомом маршруте. Возможны ситуации, когда формирование при осуществляется и по другим критериям:

- Надежность элементов сети связи;
- Время задержки передачи информации в элементах сети;
- Скорость передачи информации и прочие.
- Данные параметры являются случайными величинами и зависят от многих причин:
- Вида и интенсивности трафика пользователей сети;
- Условий окружающей среды при эксплуатации оборудования сети;
- Технического состояния оборудования сети и других причин.

Поэтому в процессе эксплуатации сетей связи могут возникнуть ситуации, при которых необходимо скорректировать таблицы маршрутизации и тем самым переформировать.

Если в процессе эксплуатации сетей связи происходит автоматическое переформирование при (без участия администрации сети), то такой при называют *динамическим*. Иначе формирование при будет *статическим*.

При позволяет определить маршруты между любой парой узлов на сети связи. Для этого необходимо во всех транзитных ук, начиная с уи, обращаясь к таблице маршрутизации, выбрать вектор-строку, номер которой совпадает с номером уп. В данной вектор-строке необходимо выбрать исходящую лс первого выбора. Если исходящая лс первого выбора оказалась недоступной (занятость передачей другой информации или неисправность аппаратуры), то следует выбрать исходящую лс второго выбора. В случае недоступности исходящей лс второго выбора необходимо выбрать следующую по предпочтительности исходящую лс. Данная процедура продолжается во всех узлах, участвующих в формировании искомого маршрута, пока не будет определен маршрут между заданной парой узлов.

В случае недоступности всех исходящих лс в данном узле потребуется либо вернуться на предыдущий ук и выбрать менее предпочтительную исходящую лс, либо дать отказ на невозможность организации искомого маршрута между заданной парой узлов.

Таким образом, анализируя процедуры, участвующие в формировании маршрутов можно заключить, что **маршрутизация состоит из двух этапов:**

1. Формирование при на сети связи.
2. Выбор исходящих лс в ук при поиске маршрута между уи и уп.

Протоколы, реализующие формирование и коррекцию при (формирование таблиц маршрутизации), часто называют протоколами маршрутизации. Протоколы, отвечающие за выбор исходящих лс в ук (формирование таблиц коммутации), – протоколами сигнализации.

Маршрутизация и модель вос.

В модели вос функции маршрутизации возложены на третий – сетевой уровень (network layer). Данный уровень удобно представить в виде подуровней (рисунок 5.2). На третьем, верхнем подуровне производится формирование.

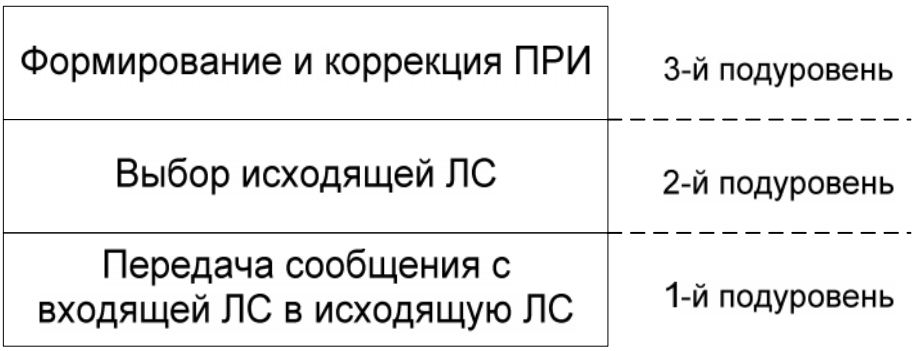


Рис. 5.2 – подуровни сетевого уровня модели ВОС.

Администрацией при проектировании или модификации сети связи. Частота коррекции при зависит от многих факторов:

- Использование статических или динамических методов маршрутизации;
- Набора статистики (за определенный период времени) о состоянии элементов сети связи (неисправность и перегруженность);
- Степени централизации устройств управления сетью связи (централизованные, децентрализованные или комбинированные методы управления);
- Возможности администрации влиять на процесс управления сетью связи;
- Наличие постоянных (не коммутируемых) соединений между пользователями сети связи.

Сформированные таблицы маршрутизации для каждого ук передаются на второй подуровень.

На втором подуровне решается задача определения и выбора (в каждом транзитном ук, начиная с уи) исходящих лс. Вызывающий пользователь сети инициирует пакет вызова на установление соединения с вызываемым пользователем. Пакет вызова, проходя через узлы коммутации, обращается к таблицам маршрутизации, которые сформированы на третьем подуровне. Затем пакет вызова определяет исходящие лс. Таким образом, в каждом транзитном ук, начиная с уи, формируются *таблицы коммутации*. В таблице коммутации указываются конкретные исходящие лс, участвующие в формировании маршрута между вызывающим и вызываемым пользователями.

Сформированные таблицы коммутации передаются на первый подуровень. В данном подуровне в соответствии с таблицами коммутации происходит передача сообщения по маршруту, сформированному на втором подуровне.

Методы формирования плана распределения информации на сети связи (таблиц маршрутизации).

Метод рельефов. Суть данного метода состоит в следующем. Пусть i – произвольный ук сети связи. I–рельефом называется процедура присвоения значения числовой функции каждой лс. I–рельеф строится следующим образом. Из i-го ук по всем исходящим лс передается число 1. Все ук, в которые поступило число 1, передают по всем исходящим лс, кроме тех лс, по которым поступила 1, число 2. Далее ук, на которые поступило число 2, передают по лс, кроме тех, по которым поступила 2, число 3 и т.д., до тех пор, пока все лс не будут пронумерованы. Говорят, что лс имеет n высоту, если она обозначена числом n в i–рельефе.

Указанным способом формируется рельеф из каждого ук сети связи. В итоге получается, что каждая лс имеет s высот. В результате лс с минимальной высотой является исходящей лс первого выбора. Лс с большими высотами, соответственно, являются исходящими лс второго, третьего и т.д. Выбора.

4. Построим рельеф на сети относительно ук a (рисунок 5.3). Ук a по исходящим лс ав, ac, ad передает число 1 и присваивает им это значение. Узлы b, c и d передают по лс вg, вс, ci, ск, cd и dk в узлы g, i и к число 2. В свою очередь, узлы g, i, н и к передают по лс gl, gi, il, im, ih, hk и ко число 3. Перечисленным лс присваивается число 3. Ук l, м, н, о, в свою очередь, передают по лс lm, mn, нм, но и on число 4 и им присваивается число 4. Таким образом, на сети строится a-рельеф (рисунок 5.4).

Чтобы найти кратчайший маршрут от произвольного ук к узлу a достаточно в каждом ук выбирать исходящую лс с меньшим весом. Например, кратчайший маршрут от ук n до ук a будет следующий: $\mu^1_{N,A} = \{NO, OK, KD, DA\}$ или $\mu^2_{N,A} = \{NM, MI, IC, CA\}$, $\mu^3_{N,A} = \{NO, OK, KC, CA\}$.

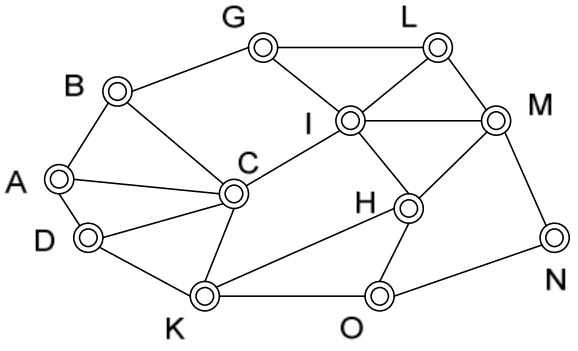
Недостатком данного метода является необходимость передачи информации при формировании рельефов между всеми узлами.

Игровой метод формирует при по накопленной ранее статистике установления соединения между заданной парой ук. Перед началом функционирования на сети устанавливается начальный при в виде набора таблиц маршрутизации (5.1). Каждому значению $m_{iv}^{(j)}$ присваивается некоторый весовой коэффициент $p_{i1}^{(j)}$. Причем, $p_i^{(j)} = \left(p_{i1}^{(j)},..., p_{iv}^{(j)},..., p_{iH_j}^{(j)}\right);\ v = \overline{1,H};\ i, j = \overline{1,S};\ i \neq j$ нормируется $\sum_{v=1}^{H} p_{iv}^{(j)} = 1.$

В результате формируется матрица весовых коэффициентов

$$P^{(j)} = \left\| p_{i,v}^{(j)} \right\|_{(S-1),H_j} = \left(\overline{p_1^{(j)}},..., \overline{p_i^{(j)}},..., \overline{p_{j-1}^{(j)}}, \overline{p_{j+1}^{(j)}},..., \overline{p_S^{(j)}}\right), \quad (5.4)$$

Где $\overline{p_i^{(j)}} = \left(p_{i1}^{(j)},..., p_{iv}^{(j)},..., p_{iH_j}^{(j)}\right);\ v = \overline{1,H};\ i, j = \overline{1,S};\ i \neq j.$ (5.5)



Рису. 5.3 – Формирование методом рельефов.

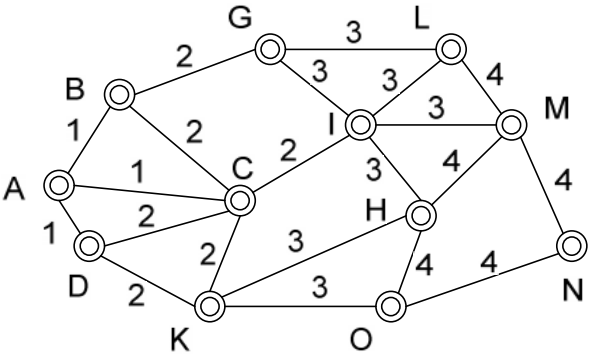


Рис. 5.4 – Формирование а–рельефа.

Определение маршрута и формирование при на сети игровым методом осуществляется следующим образом. Во всех транзитных ук, начиная с уи, при поиске маршрута к i-му уп происходит обращение к i-м строкам матриц маршрутизации (5.4). В i-х строках (5.5) определяется максимальный весовой коэффициент $p_{iv}^{(j)}$. Тем самым выбирается v-я исходящая лс из j-го ук при организации маршрута к i-му ук. В результате данных действий маршрут между заданной парой ук будет либо определен, либо данной заявке на определение маршрута будет дан отказ. В первом случае все лс, входящие в данный маршрут, поощряются. Весовые коэффициенты $p_{iv}^{(j)}$ данных исходящих лс увеличиваются. Во втором случае, когда маршрут не определен, исходящие лс, участвующие в данном поиске, штрафуются. Весовые коэффициенты $p_{iv}^{(j)}$ данных исходящих лс уменьшаются. В обоих случаях строки $\overline{p_i^{(j)}};\ i, j = \overline{1,S};\ i \neq j,$ элементы, которых были изменены (поощрены или оштрафованы), нормируется.

Критерием оптимальности является результат организации маршрутов.

5. Покажем формирование при игровом методом для сети, изображенной на рисунке 5.1. Будем считать, что начальный при задан в виде таблиц маршрутизации примера 5.3. Весовые коэффициенты (5.4) для узлов сети имеют следующий вид:

$$P^{(1)} = \left(\overline{p_2^{(1)}}, \overline{p_3^{(1)}}, \overline{p_4^{(1)}}\right) = \begin{array}{c|cc} & 2 & 4 \\ 2 & 0,7 & 0,3 \\ 3 & 0,5 & 0,5 \\ 4 & 0,3 & 0,7 \end{array},$$

$$P^{(2)} = \left(\overline{p_1^{(2)}}, \overline{p_3^{(2)}}, \overline{p_4^{(2)}}\right) = \begin{array}{c|ccc} & 1 & 3 & 4 \\ 1 & 0,6 & 0,1 & 0,3 \\ 3 & 0,2 & 0,6 & 0,2 \\ 4 & 0,2 & 0,2 & 0,6 \end{array},$$

$$P^{(3)} = \left(\overline{p_1^{(3)}}, \overline{p_2^{(3)}}, \overline{p_4^{(3)}}\right) = \begin{array}{c|cc} & 2 & 4 \\ 1 & 0,5 & 0,5 \\ 2 & 0,7 & 0,3 \\ 4 & 0,3 & 0,7 \end{array},$$

$$P^{(4)} = \left(\overline{p_1^{(4)}}, \overline{p_2^{(4)}}, \overline{p_3^{(4)}}\right) = \begin{array}{c|ccc} & 1 & 2 & 3 \\ 1 & 0,7 & 0,2 & 0,1 \\ 2 & 0,2 & 0,6 & 0,2 \\ 3 & 0,1 & 0,2 & 0,7 \end{array}.$$

Допустим, что необходимо определить маршрут между уи №2 и уп №1, при условии, что количество транзитных ук не должно превышать одного. В уи № 2 из таблицы весовых коэффициентов $p^{(2)}$ выбираем вектор-строку $\overline{p_1^{(2)}} = (0,6; 0,1; 0,3)$. исходящей лс первого выбора является лс к ук № 1. Предположим, что данная лс в настоящий момент времени недоступна. Так как $p_{1,4}^{(2)} > p_{1,3}^{(2)}$, то исходящей лс второго выбора является лс к ук № 4. Допустим, что исходящая лс из ук № 2 к ук № 4 в данный момент времени доступна. Следовательно, данная лс участвует, в организации искомого маршрута. В ук № 4 в соответствии с $\overline{p_1^{(4)}} = (0,7; 0,2; 0,1)$ выбираем исходящую лс к ук № 1. Допустим, она доступна. Следовательно, маршрут между уи и уп $\mu_{2,1} = \{2, 4, 1\}$ организован. Лс, участвующие в данной процедуре, поощряются. Соответствующие весовые коэффициенты $p_{1,4}^{(2)}, p_{1,1}^{(4)}$ увеличиваются (предположим, что на 0,2), а вектора $\overline{P_1^{(2)}}, \overline{P_1^{(4)}}$ нормируются. В результате получаем новые числовые значения: $\overline{P_1^{(2)}} = (0,5; 0,08; 0,42)$; $\overline{P_1^{(4)}} = (0,75; 0,17; 0,08)$.

Если ситуация поиска маршрута между заданной парой ук повторится, то вектора $\overline{P_1^{(2)}}, \overline{P_1^{(4)}}$ изменятся и примут следующий вид: $\overline{P_1^{(2)}} = (0,42; 0,07; 0,51)$; $\overline{P_1^{(4)}} = (0,79; 0,14; 0,07)$. анализируя ситуацию с вектором $\overline{P_1^{(2)}}$, видно, что исходящая лс к ук № 4 из ук № 2 при поиске маршрута к ук № 1 приняла значение первого выбора, так как ее весовой коэффициент $P_{1,4}^{(2)}$ стал максимальным из всех возможных в данном векторе.

Матрицы весовых коэффициентов ук № 2 и 4 примут следующий вид:

$$P^{(2)} = \begin{array}{c|ccc} & 1 & 3 & 4 \\ 1 & 0,42 & 0,07 & 0,51 \\ 3 & 0,2 & 0,6 & 0,2 \\ 4 & 0,2 & 0,2 & 0,6 \end{array}; \quad P^{(4)} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ 1 & 0,79 & 0,14 & 0,07 \\ 2 & 0,2 & 0,6 & 0,2 \\ 3 & 0,1 & 0,2 & 0,7 \end{array}.$$

Если рассматривать весовые коэффициенты $P_{iv}^{(j)}$ как вероятности выбора соответствующих исходящих лс $m_{iv}^{(j)}$, то можно предположить, что игровой метод решает задачу глобальной оптимизации сети связи по критерию – вероятность установления соединения между парами уи и уп.

Отсутствие необходимости передачи служебной информации при формировании при на сети является несомненным достоинством игрового метода. Однако данный метод обладает инерционностью. Действительно, при выходе элементов сети связи из строя потребуется некоторый период времени для переформирования при на сети.

Логический метод состоит в процедуре, выполняемой в каждом транзитном ук, начиная от уи, позволяющей определить исходящую лс, максимально близкой к геометрическому направлению на уп.

Сеть связи вкладывается в прямоугольную систему координат. Каждому узлу сети присваивается собственный адрес (x, y) (рисунок 5.5).

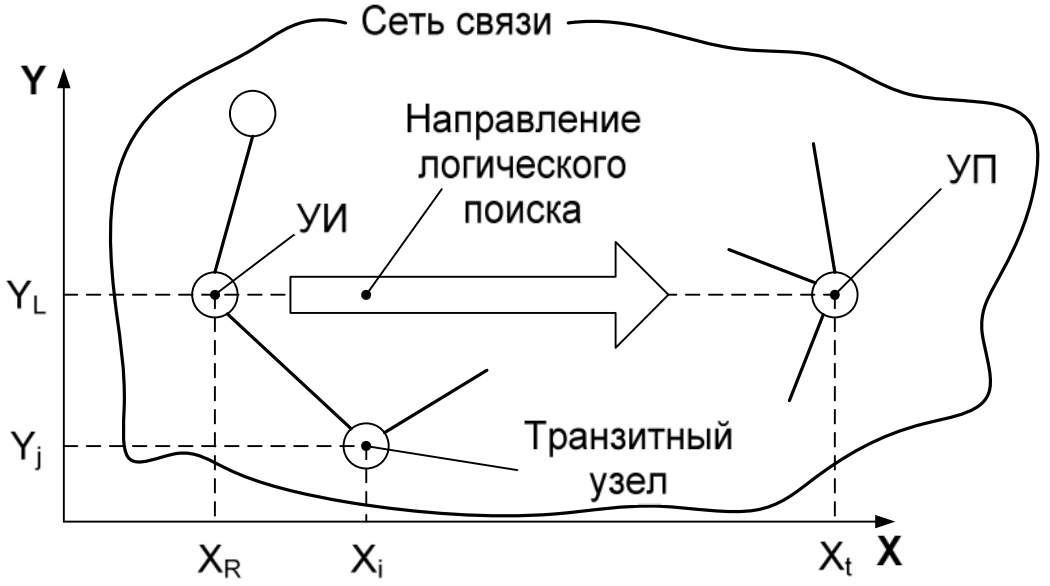


Рис. 5.5 – Поиск маршрута логическим методом

В каждом транзитном ук (x_i, y_j), начиная с уи (x_r, y_l), производится анализ адреса уп сопоставлением его с собственным. В результате вычисляется геометрическое направление из данного узла на уп. Затем определяется та лс, которая имеет наибольшее совпадение с ранее рассчитанным геометрическим направлением на уп. Если ближайшая по направлению исходящая лс не доступна, то подбирается очередная по предпочтительности исходящая лс.

6. На рисунке 5.6 представлена сеть связи, в которой уи и уп, соответственно, имеют координаты {1, 2} и {10, 2}. Из уи определяем геометрическое направление на уп (указано пунктиром). С данным направлением совпадает исходящая лс к узлу с координатами {4, 2}. В ук {4, 2} выбираем исходящую лс к ук с координатами {7, 3}, так как она имеет наименьший угол отклонения от геометрического направления на уп. В ук {7, 3} подобным образом выбираем лс к ук {8, 2}. В ук {8, 2} выбираем лс к ук {10, 2}.

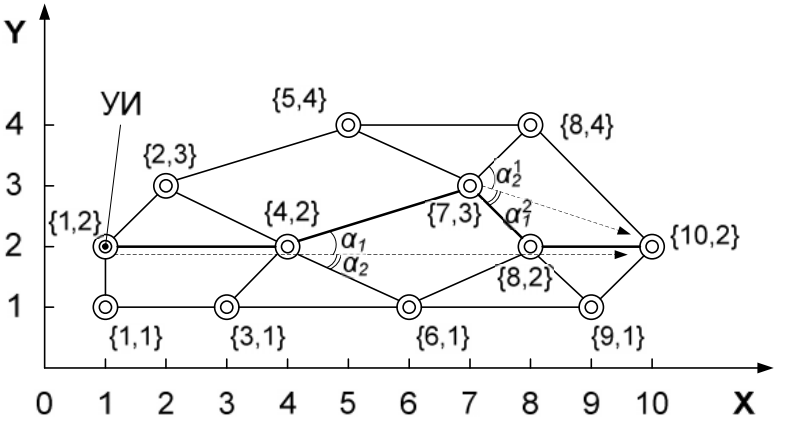


Рис. 5.6– Пример формирования при логическим методе

Таким образом: $\mu(\{1,2\};\{10,2\})=(\{1,2\},\{4,2\},\{7,3\},\{8,2\},\{10,2\})$. достоинством данного метода является простота и отсутствие необходимости передачи служебной информации по сети. В то же время логический метод не является динамическим и не решает задачу глобальной оптимизации при.

Логически-игровой метод формирования при является обобщением логического и игрового методов. По аналогии с логическим методом сеть связи вкладывается в прямоугольную систему координат, в соответствии с которой каждому узлу сети присваивается собственный адрес (x, y). В каждом ук j имеется матрица $P^{0(j)}=\left\|p_{i,v}^{0(j)}\right\|_{S^0,(H_j+3)}$; $v=\overline{1,H_j}$; $i,j=\overline{1,S^0}$; $i\neq j$, которая имеет следующий вид:

Табл.5.1 – Матрица значений весовых коэффициентов

№ уп	Координаты уп		Значения весовых коэффициентов исходящих лс к смежным ук с координатами							
	X	Y	Xqj	Yqj	…	Xvj	Yvj	…	Xhj	Yhj
1					…			…		
…	…	…	…	…	…	…	…	…	…	…
I					…			…		
…	…	…	…	…	…	…	…	…	…	…
J-1					…			…		
J+1					…			…		
…	…	…	…	…	…	…	…	…	…	…
S					…			…		
…	…	…	…	…	…	…	…	…	…	…
S0					…			…		

И содержит s^0 строк. Учитывая, что возможно увеличение числа ук на сети, то s^0 выбирают таким, чтобы $s^0>s$. Количество столбцов матрицы $p^{0(j)}$ для ук под номером j равно: $(н_j+3)$, где $н_j$ – число исходящих лс из j-го узла; три столбца отводится для номеров уп, представленных в общепризнанной нумерации и прямоугольной системе координат (x, y).

На момент ввода узла в эксплуатацию матрица содержит только информацию о смежных номерах ук с данными выраженных в прямоугольной системе координат: (x_{qj}, y_{qj}), ..., (x_{vj}, y_{vj}), ..., (x_{hj}, y_{hj}). По мере функционирования сети связи матрица $p^{0(j)}$ заполняется и корректируется.

Определение исходящих лс осуществляется логическим методом, а заполнение и корректировка матрицы $p^{0(j)}$ осуществляется игровым методом.

Выбор исходящих лс (формирование таблиц коммутации).

Последовательный выбор исходящих лс состоит в том, что в каждом ук, начиная с уи, осуществляется выбор только одной исходящей лс. В результате на сети будет формироваться один маршрут, состоящий из последовательного наращивания коммутационных участков из уи к уп.

В зависимости от характера распространения на сети процесса поиска маршрута выделим три основных класса последовательных алгоритмов выбора исходящих лс: *градиентный*, *диффузный* и *градиентно-диффузный*.

Градиентный состоит в том, что в каждом транзитном ук, начиная с уи, в процессе выбора исходящей лс участвуют не все лс, а лишь часть (наиболее предпочтительные). Если в одном из ук исходящие лс, участвующие в выборе, не доступны, то данной заявке на формирование маршрута дается отказ.

В результате градиентного выбора маршрут будет формироваться вдоль геометрического направления с уи на уп (рисунок 5.7).

Выбор лс, при котором искомый маршрут формируется и в противоположную сторону от уп, будем называть *диффузным*.

Таким образом, диффузный выбор исходящих лс допускает возможность выбора любой доступной исходящей лс (рисунок 5.7).

Градиентно-диффузный метод является комбинацией первых двух.

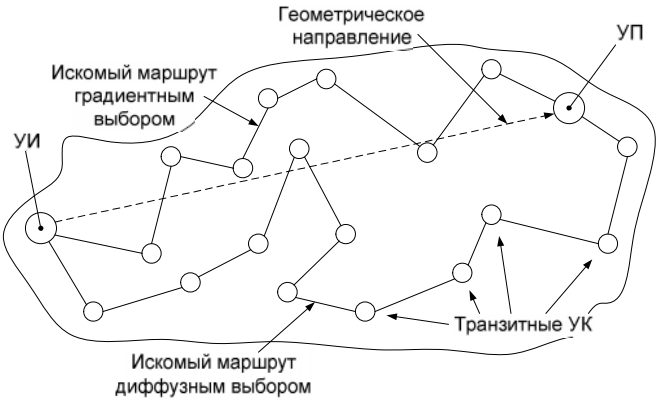


Рис. 5.7– Градиентный и диффузный выбор исходящих лс.

В свою очередь процедура выбора исходящих лс в каждом ук может быть детерминированной и вероятностной. В первом случае выбор исходящих лс осуществляется однозначно по максимальному значению одного из элементов вектора (6.5). Во втором случае выбор исходящей лс производится в результате случайного розыгрыша. При этом исходящие лс. Имеющие большее значения $p_{iv}^{(j)}$, получают большую вероятность выбора.

Возможен и комбинированный способ выбора исходящих лс, который содержит как вероятностную, так детерминированную компоненты.

Учитывая перечисленные градации, можно указать множество вариантов последовательных алгоритмов выбора исходящих лс в ук (например, «диффузный, вероятностный» или «градиентно-диффузный, детерминированный»).

Параллельный выбор исходящих лс. Отличительная особенность алгоритмов с параллельным выбором исходящих лс состоит в том, что поиск маршрута между уи и уп осуществляется одновременно по всем исходящим лс в определенной зоне сети связи.

Если выбор ширины зоны, в которой осуществляется поиск маршрута, определяется однозначно, по заранее выбранным критериям, то такой выбор будем называть детерминированным. Если же выбор ширины зоны поиска маршрута осуществляется в результате случайного выбора, то в данном случае выбор будем называть вероятностным.

Классическим примером параллельного выбора исходящих лс с детерминированным выбором ширины зоны поиска маршрута, является алгоритм, получивший во многих публикациях название волновой, или лавинный. При поступлении заявки на организацию маршрута между парой узлов в уи формируется поисковая посылка, которая пересылается ко всем соседним с ним узлам. В соседних ук эта процедура повторяется. Таким образом, поисковая посылка попадает во все узлы сети, причем через время, равное времени его передачи по кратчайшему маршруту. Основным недостатком волнового метода маршрутизации является дополнительная нагрузка, которая создается передачей поисковой посылки во все стороны, в том числе и в противоположную сторону от уп.

Локально-волновой метод маршрутизации является обобщением волнового метода маршрутизации и логического способа получения при на сети связи. Локально-волновой метод маршрутизации в зависимости от организации выбора исходящей лс может быть отнесен к параллельным и параллельно-последовательным методам. В то же время, способ выбора зоны, в которой осуществляется поиск маршрута, в локально-волновом методе может быть вероятностным, детерминированным и комбинированным.

Локально-волновой метод маршрутизации состоит в том, что для нахождения оптимального маршрута в сети между парой узлов из уи организуется волновой поиск, но не во всех направлениях, а лишь в сторону уп. Волна поиска при этом распространяется в некоторой зоне (рисунок 5.8). Ширина и форма зоны в зависимости от приоритета абонента может устанавливаться в заданных пределах. На рисунке 5.8 показан локально-волновой поиск на сети от уи к уп в некоторый момент времени, соответствующий примерно половине пути

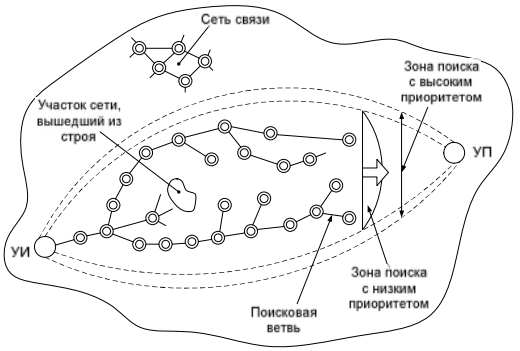


Рис. 5.8 – Поиск маршрута локально–волновым методом.

Между парой узлов. Из рисунка видно, что поисковая волна - это подвижная узкая зона, все узлы в пределах которой охвачены процессом волнового поиска. По мере продвижения к уп волна оставляет за собой лс, исходящие из уи. Чем выше приоритет абонента, тем больше возможностей он имеет для установления соединения. Таким образом, при данном методе в каждом узле определяются исходящие лс из данного узла к смежным узлам, наиболее близко совпадающие с геометрическим направлением на искомый узел. Выбранные исходящие лс располагаются в ряд по степени предпочтительности.

Количество подсоединенных лс, а следовательно, и ширина поисковой волны, определяется приоритетом вызывающего абонента. В частности, для абонентов низшей категории количество выбранных лс может не превышать одного, тогда поиск превращается в «чисто» последовательный.

На рисунке 5.9 приведена классификация методов маршрутизации на сети связи. Из рисунка следует, что существует множество вариантов реализации как последовательных, так и параллельных методов маршрутизации. Например: «вероятностный, диффузный с использованием динамического формирования при методом рельефов».

Рис. 5.9 – Классификация методов маршрутизации

Контрольные вопросы и задания

1. Дайте определение: маршрут; маршрутизация; таблица маршрутизации; план распределения информации на сети связи.
2. Назовите основные требования, предъявляемые к маршрутизации.
3. Назовите основные методы формирования при на сети связи.
4. Перечислите достоинства и недостатки метода рельефов.
5. Перечислите достоинства и недостатки игрового метода.
6. Перечислите достоинства и недостатки логического метода.
7. Перечислите достоинства и недостатки логически-игрового метода.
8. Перечислите достоинства и недостатки волнового метода.
9. В чем суть локально–волнового метода маршрутизации?
10. В соответствии со структурой сети, полученной от преподавателя, сформировать план

распределения информации, используя метод рельефов, игровой метод и логический метод. Сравните перечисленные методы по трудоемкости и эффективности.

Лабораторная работа № 6. Методы защиты от ошибок. Защита от ошибок в системах без обратной связи.

В системах без обратной связи (однонаправленных) для повышения верности приема используются следующие основные способы: многократная передача кодовых комбинаций; одновременная передача кодовой комбинации по нескольким параллельно работающим каналам; помехоустойчивое кодирование, т.е. Использование кодов, исправляющих ошибки (корректирующих кодов).

Многократная передача кодовых комбинаций является наиболее просто реализуемым способом повышения верности. Повторение кодовых комбинаций может осуществляться вручную и автоматически (без участия операторов). Пусть передается буква а, число повторений возьмем равным пяти. Если на приемном конце имеем абаас (буква а исказилась 2 раза, превратившись соответственно в б и с), то выносится решение о том, что передавалась буква а, поскольку в последовательности из пяти букв она встречалась наиболее часто. Если в принятой последовательности ни одна из букв не повторяется, то принятое сообщение ликвидируется (стирается).

Главный недостаток такого способа - существенное уменьшение скорости передачи. В нашем примере скорость передачи информации уменьшается в 5 раз по сравнению со случаем однократной передачи кодовых комбинаций.

Способ повышения верности, основанный на снижении скорости передачи, широко применяется в технике передачи дискретных сообщений. Так, в среднескоростных системах пд с частотной модуляцией предусмотрены две скорости - 600 и 1200 бод. Очевидно, что передача со скоростью 600 бод равносильна передаче 2 раза подряд единичных элементов длительностью 1/1200 мс.

При одновременной передаче кодовой комбинации по нескольким параллельным каналам (обычно число каналов нечетное) решение о том, какая кодовая комбинация передавалась, выносится методом голосования (т.е. Так же, как и при многократной передаче кодовых комбинаций). Иногда передача осуществляется по двум параллельным каналам, и информация выбирается из того канала, качество которого в момент приема кодовой комбинации было наилучшим. Для этого приемник должен располагать соответствующим устройством для оценки качества канала.

При передаче сообщений по n параллельным каналам скорость передачи информации не зависит от числа каналов. Однако при этом существенно возрастают (в n раз!) Расходы на аренду каналов.

Более эффективно используются дискретные каналы при применении корректирующих кодов. В однонаправленных системах это должны быть коды, исправляющие ошибки. Широкое распространение на практике получили двоичные корректирующие коды, т.е. Коды, при формировании которых используются только два типа элементов: 0 и 1. Только такие коды и будут рассматриваться в данной главе.

Построение корректирующих кодов.

Каждому символу исходного алфавита сообщений объема n_a поставим в соответствие n-элементную двоичную последовательность - кодовую комбинацию. Возможное (общее) число последовательностей длины n составляет $n_0=2^n$, причем должно соблюдаться условие $n_0>n_a$.

Если $n_0=n_a$, то все возможные последовательности n-элементного кода используются для передачи или, как говорят, являются разрешенными. Полученный таким образом код называется простым.

1. Для передачи сообщений, число которых равно восьми (na=8), используется трехэлементный код. Число кодовых комбинаций, которое можно при этом получить, n0=23=8. Из таблицы 7.1 видно, что комбинация под номером 0 отличается от комбинации 1 только в одной позиции. Следовательно, если при передаче комбинации 000 произойдет ошибка в третьем элементе, то получим комбинацию 001.

Степень различия комбинаций определяется *расстоянием хемминга d*. Это расстояние для любых двух кодовых комбинаций определяется числом несовпадающих в них разрядов. Например, две ниже написанные друг под другом комбинации не совпадают в двух разрядах:

$$\begin{array}{r} 0\,1\,0 \\ \oplus\ 1\,0\,0 \\ \hline 1\,0\,0 \end{array}$$

Поэтому расстояние хемминга *d*=2. Иначе его определяют как вес суммы по модулю два ($\oplus$ – условное обозначение суммы) этих кодовых комбинаций. *Весом w* кодовой комбинации называется число входящих в нее ненулевых элементов.

Табл. 6.1 – Кодовые комбинации трехэлементного кода

Номер комбинации	0	1	2	3	4	5	6	7
Вид комбинации	000	001	010	011	100	101	110	111

Перебрав все возможные пары кодовых комбинаций, можно найти *минимальное хеммингово расстояние*, которое принято называть кодовым и обозначать d_0. Для примера 6.1 кодовое расстояние d_0=1. Рассмотренный в примере код простой. Любая ошибка (даже одиночная!) При использовании такого кода приведет к тому, что переданная разрешённая кодовая комбинация перейдет в другую разрешенную. Таким образом, простой код не способен обнаруживать и тем более исправлять ошибки и имеет d_0=1.

Для того чтобы код мог обнаруживать ошибки, необходимо, чтобы соблюдалось неравенство $n_a<n_0$. При этом неиспользуемые n-элементные кодовые комбинации, число которых (n_0-n_a), будем называть *запрещенными*. Они определяют избыточность кода. Очевидно, что появление ошибки в кодовой комбинации будет обнаружено, если переданная разрешенная комбинация перейдет в одну из запрещенных. В качестве $n_p=n_a$ разрешенных кодовых комбинаций надо выбирать такие, которые максимально отличаются друг от друга.

2. Алфавит передаваемых сообщений na=2. Выберем из числа комбинаций, представленных в таблице 6.1, две. Очевидно, что ими должны быть комбинации 000, 111 или 001 и 110 и т.д. Кодовое расстояние d0=3. Ошибки кратности один или два превращают любую разрешенную кодовую комбинацию в запрещенную. Следовательно, максимальная кратность обнаруживаемых таким образом ошибок равна двум (to.ош=2).

Нетрудно догадаться, что минимальное кодовое расстояние d_0 и гарантированно обнаруживаемая кратность ошибок связаны соотношением $t_{о.ош}=d_0-1$.

Исправление ошибок возможно также только в том случае, если переданная разрешенная кодовая комбинация переходит в запрещенную. Вывод о том, какая кодовая комбинация передавалась, делается на основании сравнения принятой запрещенной комбинации со всеми разрешенными. Принятая комбинация отождествляется с той из разрешенных, на которую она больше всего похожа, т.е. С той, от которой она отличается меньшим числом элементов. Так, если в примере 6.2 при передаче кодовой комбинации 000 получим 001, то вынесем решение, что передавалась кодовая комбинация 000.

Связь между d_0 и кратностью исправляемых ошибок определяется выражением $t_{и.ош}=(d_0/2) - 1$ для четного d_0 и $t_{и.ош} =(d_0 - 1) / 2$ для нечетного d_0.

Итак, задача получения кода с заданной корректирующей способностью сводится к задаче выбора (путем перебора) из $n_0=2^n$ кодовых комбинаций n_a комбинаций с требуемым кодовым расстоянием d_0. Если n достаточно мало, то такой перебор не представляет особого труда. При больших n перебор может оказаться непосильным даже для современной эвм, поэтому на практике используют методы построения кодов, не требующие перебора с целью получения кода с заданным d_0 и отличающиеся невысокой сложностью реализации.

Классификация корректирующих кодов.

Помехоустойчивые или корректирующие коды (рисунок 6.1) делятся на блочные и непрерывные. К блочным относятся коды, в которых каждому символу алфавита сообщений соответствует блок (кодовая комбинация) из n(i) элементов, где i – номер сообщения. Если n(i)=n, т.е. Длина блока постоянна и не зависит от номера сообщения, то код называется *равномерным*. Такие коды чаще применяются на практике. Если длина блока зависит от номера сообщения, то блочный код называется неравномерным. Примером неравномерного кода служит код морзе. В непрерывных кодах передаваемая информационная последовательность не разделяется на блоки, а проверочные элементы (проверочные элементы в отличие от информационных, относящихся к исходной последовательности, служат для обнаружения и исправления ошибок и формируются по определенным правилам) размещаются в определенном порядке между информационными.

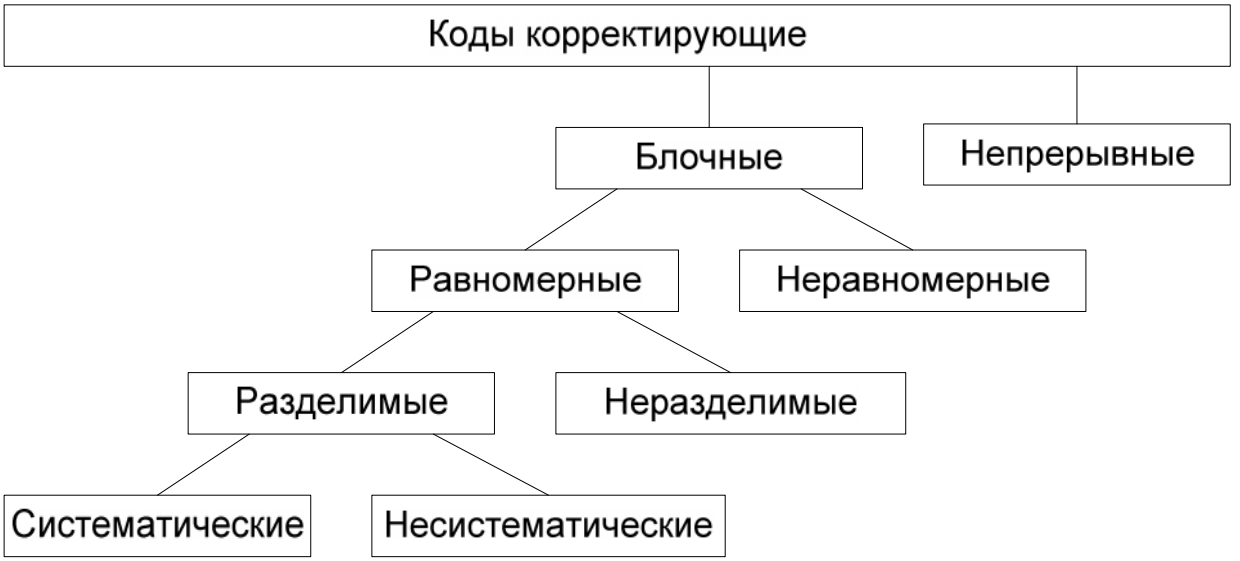


Рис. 6.1– Классификация корректирующих кодов

Равномерные блочные коды делятся на разделимые и неразделимые. В разделимых кодах элементы разделяются на информационные и проверочные, занимающие определенные места в кодовой комбинации, во-вторых, отсутствует деление элементов кодовых комбинаций на информационные и проверочные. К последним относится код с постоянным весом, например рекомендованный международным консультативным комитетом по телефонии и телеграфии (мкктт), семиэлементный телеграфный код № 3 с весом каждой кодовой комбинации, равным трем.

Примерами систематических кодов являются коды хемминга и циклические. Последние реализуются наиболее просто, что и привело к их широкому использованию в узо. Для систематического кода применяется обозначение (n,k) – код, где n – число элементов в комбинации; k – число информационных элементов.

Характерной особенностью этих кодов является также и то, что информационные и проверочные элементы связаны между собой зависимостями, описываемыми линейными уравнениями. Отсюда возникает и второе название систематических кодов – линейные.

Код хемминга.

Рассмотрим в качестве примера построение систематического кода с кодовым расстоянием $d_0=3$ (кода хемминга). Пусть число сообщений, которое необходимо передать, равно 16. Тогда необходимое число информационных элементов $k=\log_2 n_a=4$. Можно выписать все 16 кодовых комбинаций, включая нулевую (0000). Это один из возможных способов задания исходного (простого) кода. Другой способ заключается в выписывании только четырех кодовых комбинаций простого кода в виде матрицы, называемой единичной:

$$\begin{Vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{Vmatrix} \qquad (6.1)$$

Суммируя по модулю два в различном сочетании кодовые комбинации, входящие в единичную матрицу, можно получить 15 кодовых комбинаций, 16-я – нулевая. Кодовые комбинации, составляющие матрицу (6.1), линейно независимы. Можно было бы составить матрицу и из других кодовых комбинаций (лишь бы они были линейно независимыми). Ненулевые комбинации a_1, a_2, a_3, a_4 линейно независимые, если $q_1a_1 \oplus q_2a_2 \oplus q_3a_3 \oplus q_4a_4 \neq 0$, где $q_i \in \{0,1\}$ при условии, что хотя бы один из коэффициентов $q_i \neq 0$. Дополним каждую кодовую комбинацию в (6.1) проверочными элементами так, чтобы обеспечивалось $d_0=3$. Будем иметь в виду также тот факт, что к числу разрешенных комбинаций корректирующего кода принадлежит и комбинация 0000...0, называемая нулевой. Очевидно, что в числе добавляемых проверочных элементов должно быть не менее двух единиц. Тогда общее число единиц в каждой комбинации кода получим не меньше трех и комбинации, полученные нами, будут отличаться от нулевой, по крайней мере, в трех элементах. Добавим по две единицы к каждой строке матрицы (6.1):

$$\begin{Vmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{Vmatrix}, \qquad (6.2)$$

Складывая строки 1 и 2 матрицы (6.2) по модулю два

$$\begin{array}{r} 1\,0\,0\,0\,1\,1 \\ \oplus\ 0\,1\,0\,0\,1\,1 \\ \hline 1\,1\,0\,0\,0\,0 \end{array},$$

Видим, что они отличаются только в двух элементах, т.е. Заданное кодовое расстояние не обеспечивается. Дополним каждую строку проверочными элементами так, чтобы $d_0=3$. Тогда матрица примет вид

$$\begin{Vmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{Vmatrix}. \qquad (6.3)$$

Добавляемые проверочные элементы могут быть записаны и в другом порядке. Необходимо лишь обеспечить $d_0=3$.

Матрицу (6.3) называют производящей, или порождающей, матрицей кода (6,4), содержащего семь элементов, из которых четыре информационных. Обычно матрицу обозначают буквой **g** с индексом, указывающим, к какому коду она относится (в нашем случае $\mathbf{g}_{(7,4)}$). Производящая матрица состоит из двух матриц - единичной (размерности k·k) и $\mathbf{c}_{(r,k)}$, содержащей r столбцов и k строк. Суммируя в различном сочетании строки матрицы (6.3), получаем все (кроме нулевой) комбинации корректирующего кода с $d_0=3$.

Обозначим элементы комбинации полученного семиэлементного кода a_1, a_2, a_3, a_4, a_5, a_6, a_7, из которых a_1, a_2, a_3, a_4 – информационные и a_5, a_6, a_7 – проверочные. Последние могут быть получены путем суммирования по модулю два определенных информационных элементов. Разумеется, правило формирования проверочного элемента a_i для любой кодовой комбинации одинаково.

Найдем правило формирования элемента a_5, пользуясь матрицей (6.3). Из первой строки следует, что в суммировании должен обязательно участвовать элемент a_1 (только в этом случае $a_5=1$), из второй - что элемент a_3 в суммировании не должен участвовать, а из четвертой - что элемент a_4 должен участвовать в суммировании. Итак,

$$a_5 = a_1 \oplus a_2 \oplus a_4. \qquad (6.4)$$

уравнения для a_6 и a_7 по аналогии записываются в виде:

$$a_6 = a_1 \oplus a_3 \oplus a_4, \qquad (6.5)$$

$$a_7 = a_1 \oplus a_2 \oplus a_3. \qquad (6.6)$$

алгоритм формирования проверочных элементов a_5, a_6, a_7 может быть задан матрицей, называемой проверочной. Эта матрица содержит r строк и n столбцов. Применительно к сформированному нами коду (6,4) она имеет вид:

$$\mathbf{B}_{(6,4)} = \begin{Vmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{Vmatrix}.$$

единицы, расположенные на местах, соответствующих информационным элементам матрицы **н**(6,4), указывают на то, какие информационные элементы должны участвовать в формировании проверочного элемента. Единица на месте, соответствующем проверочному элементу, указывает, какой проверочный элемент получается при суммировании по модулю два информационных элементов. Так, из первой строки следует равенство

$$a_1 \oplus a_2 \oplus a_4 = a_5.$$

Процедура обнаружения ошибок основана на использовании проверок (12.4)-(12.6). Очевидно, что проверочные элементы, сформированные из принятых информационных, при отсутствии ошибок должны совпадать с принятыми проверочными.

3. Переданная кодовая комбинация имеет вид 1000111 (первая строка матрицы (2.3)). В результате действия помех на приемном конце имеем $a_1', a_2', a_3', a_4', a_5', a_6', a_7' = 1100111$. Произведем проверки (12.4)-(12.6):

$$a_1' \oplus a_2' \oplus a_4' = 1 \oplus 1 \oplus 0 = 0 = a_5^{\bullet}, \qquad (6.7)$$

$$a_1' \oplus a_3' \oplus a_4' = 1 \oplus 0 \oplus 0 = 1 = a_6^{\bullet}, \qquad (6.8)$$

$$a_1' \oplus a_2' \oplus a_3' = 1 \oplus 1 \oplus 0 = 0 = a_7^{\bullet}. \qquad (6.9)$$

В то же время $a_5' = 1,\ a_6' = 1,\ a_7' = 1,$ т.е. $a_5^{\bullet} \neq a_5',\ a_7^{\bullet} \neq a_7'$, что говорит о наличии ошибок в принятой кодовой комбинации. При отсутствии в принятой кодовой комбинации ошибок $a_5' \oplus a_5^{\bullet} = b_1 = 0,\ a_6' \oplus a_6^{\bullet} = b_2 = 0,\ a_7' \oplus a_7^{\bullet} = b_3 = 0.$

Комбинация $b_3b_2b_1$ называется синдромом (проверочным вектором). Равенство нулю всех элементов синдрома указывает на отсутствие ошибок или на то, что кодовая комбинация принята с ошибками, которые превратили ее в другую разрешенную. Последнее событие имеет существенно меньшую вероятность, чем первое.

Вид ненулевого синдрома определяется характером ошибок в кодовой комбинации. В нашем случае вид синдрома зависит от местоположения одиночной ошибки. В таблице 6.2 отражено соответствие между местоположением одиночной ошибки для кода, заданного матрицей (6.3), и видом синдрома.

Табл. 6.2 – Местоположение ошибки и вид синдрома

Номер элемента, в котором произошла ошибка	1	2	3	4	5	6	7
Вид синдрома	111	101	110	011	001	010	100

Таким образом, зная вид синдрома, можно определить место, где произошла ошибка, и исправить принятый элемент на противоположный.

4. Передавалась кодовая комбинация 1000111. Принята кодовая комбинация 0000111. Синдром имеет вид 111. В соответствии с таблицей 6.2 исказился первый элемент (a_1). Изменим первый элемент на противоположный:

$$\begin{array}{r} 0\,0\,0\,0\,1\,1\,1 \\ \oplus\ 1\,0\,0\,0\,0\,0\,0 \\ \hline 1\,0\,0\,0\,1\,1\,1 \end{array}.$$

Полученная в результате исправления ошибки кодовая комбинация совпадает с переданной.

Рассмотренный код (6,4) гарантированно обнаруживает двукратные ошибки, а исправляет только однократные ошибки.

Циклические коды.

В теории циклических кодов кодовые комбинации обычно представляются в виде полинома. Так, n-элементная кодовая комбинация записывается в виде

$$A(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + ... + a_1 x + a_0,$$

Где a_i={0,1}, причем a_i=0 соответствуют нулевым элементам комбинации, а a_i=1 – ненулевым. Например, комбинациям 1101 и 1010 соответствуют многочлены $A_1(x) = x^3 + x^2 + 1$ и $A_2(x) = x^3 + x.$

при формировании комбинаций циклического кода часто используют операции сложения многочленов и деления одного многочлена на другой. Так,

$$A_1(x) + A_2(x) = \left(x^3 + x^2 + 1\right) + \left(x^3 + x\right) = x^2 + x + 1,$$

Поскольку $x^3 + x^3 = x^3(1 \oplus 1) = 0.$ рассмотрим операцию деления на следующем примере:

$$\begin{array}{l|l} x^7+x^5+x^4+x^2+1 & x^3+x^2+1 \\ \hline x^7+x^6+x^4 & x^4+x^3+1 \\ \hline x^6+x^5+x^2 & \\ x^6+x^5+x^3 & \\ \hline x^3+x^2+1 & \\ x^3+x^2+1 & \\ \hline 0\ \ 0\ \ 0 & \end{array}$$

деление выполняется, как обычно, только вычитание заменяется суммированием по модулю два.

Разрешенные комбинации циклического кода обладают двумя очень важными отличительными признаками: циклический сдвиг разрешенной комбинации тоже приводит к разрешенной кодовой комбинации. Все разрешенные кодовые комбинации делятся без остатка на полином p(x), называемый *образующим*. Эти свойства используются при построении кодов, кодирующих и декодирующих устройств, а также при обнаружении и исправлении ошибок.

Найдем алгоритмы построения циклического кода, удовлетворяющего перечисленным выше условиям. Задан полином $P(x) = a_{r-1}x^r + a_{r-2}x^{r-1} + ... + 1,$ определяющий корректирующую способность кода, и задан исходный простой код, который требуется преобразовать в корректирующий циклический.

Обозначим многочлен, соответствующий комбинации простого кода, q(x). Возьмем произведение $q(x)x^r$ и разделим его на p(x). В результате получим многочлен g(x) и остаток r(x)/p(x):

$$\frac{Q(x)x^r}{P(x)} = G(x) + \frac{R(x)}{P(x)}. \qquad (6.10)$$

Умножим левую и правую части на p(x), тогда (6.10) перепишется в виде

$$Q(x)x^r = G(x)P(x) + R(x). \qquad (6.11)$$

перепишем равенство (6.11) в виде

$$G(x)P(x) = Q(x)x^r + R(x). \qquad (6.12)$$

левая часть (2.12) делится без остатка на p(x), значит, без остатка делится и правая часть. Из (6.12) вытекают два способа формирования комбинаций циклического кода: путем умножения многочлена g(x) на p(x) и путем деления $q(x)x^r$ на p(x) и приписывания к $q(x)x^r$ остатка от деления r(x).

5. Задан полином $g(x)=x^3+x$, соответствующий комбинации простого кода. Сформировать комбинацию циклического кода (7,4) с производящим полиномом $p(x)=x^3+x^2+1$. Можно получить комбинацию циклического кода в виде $g(x)p(x)=(x^3+x)(x^3+x^2+1)=x^6+x^5+x^4+x$. Однако в полученной

комбинации нельзя отделить информационные элементы от проверочных, и код получается неразделимым.

Перейдем ко второму способу, который чаще всего применяется на практике. Проделаем необходимые операции по получению комбинации циклического кода:

1) $G(x)x^r=(x^3+x)x^3=x^6+x^4$;

2)

$$
\begin{array}{l|l}
x^6+x^4 & x^3+x^2+1 \\
\oplus\ x^6+x^5+x^3 & \hline
x^3+x^2+1 \\
\hline
x^5+x^4+x^3 & \\
x^5+x^4+x^2 & \\
\hline
x^3+x^2 & \\
x^3+x^2+1 & \\
\hline
R(x)=1 &
\end{array}
,
$$

3) (x^6+x^4+1) – комбинация циклического кода, полученная методом деления на производящий полином. Она может быть переписана в виде 1010001. Первые четыре элемента – информационные, последние три - проверочные, т.е. Полученный код – разделимый.

Для обнаружения ошибок в принятой кодовой комбинации достаточно поделить ее на производящий полином. Если принятая комбинация разрешенная, то остаток от деления будет нулевым. Ненулевой остаток свидетельствует о том, что принятая комбинация содержит ошибки. По виду остатка (синдрома) можно в некоторых случаях также сделать вывод о характере ошибки и исправить ее.

Циклические коды достаточно просты в реализации, обладают высокой корректирующей способностью (способностью исправлять и обнаруживать ошибки) и поэтому рекомендованы мсэ-т для применения в аппаратуре пд. Согласно рекомендации v.41 в системах пд с ос рекомендуется применять код с производящим полиномом $p(x)=x^{16}+x^{12}+x^5+1$.

Эффективность применения корректирующих кодов.

Полезный эффект от применения корректирующих кодов заключается в повышении верности. Вероятность неправильного приема кодовой комбинации простого кода определяется как вероятность появления в кодовой комбинации хотя бы одной ошибки, т.е.

$$P_{ош}^{(\text{п})}=1-(1-p_{ош})^k,$$

Где $p_{ош}$ – вероятность неправильного приема единичного элемента; k – число элементов в комбинации простого кода. При применении систематических корректирующих кодов к исходной кодовой комбинации добавляются проверочные элементы, позволяющие исправлять или обнаруживать ошибки. Так, если код используется в режиме исправления ошибок и кратность исправляемых ошибок $t_{и.ош}$, то вероятность неправильного приема кодовой комбинации

$$P_{ош}^{(\text{и})}=\sum_{t=t_{и.ош}+1}^{n} C_n^t p_{ош}(1-p_{ош})^{n-t}.$$

В результате применения корректирующего кода в режиме исправления ошибок вероятность ошибки уменьшается в $k_и$ раз: $K_и=\frac{P_{ош}^{(\text{п})}}{P_{ош}^{(\text{и})}}>1.$ однако это достигается за счет увеличения затрат на реализацию системы и снижения скорости передачи информации. Если в системе с простым кодом скорость равна $c_п$, то в системе с корректирующим кодом скорость $c_к=c_п\cdot\gamma_1$, где $\gamma_1=k/n$ – коэффициент, характеризующий потери скорости вследствие введенной в код избыточности. Чем больше избыточность (меньше γ_1), тем меньше скорость передачи информации, т.е. Тем меньше в единицу времени передается полезной информации.

Качество реальных каналов во времени меняется, и если заданы требования на верность передачи, то необходимо ввести такую избыточность, которая обеспечивала бы заданную верность даже при самом плохом качестве канала. Напрашивается мысль о целесообразности изменения избыточности, вводимой в кодовую комбинацию, по мере изменения характеристик канала связи.

Системы, в которых меняется избыточность с изменением качества канала, относятся к числу адаптивных. Одним из типов адаптивных систем являются системы с обратной связью. В этих системах между приемником и передатчиком помимо основного (прямого) канала имеется вспомогательный (обратный).

Следует заметить, что системы без ос используются обычно только тогда, когда нельзя организовать канал обратной связи или когда предъявляются жесткие требования к времени задержки сообщения. Временем задержки кодовой комбинации называется время от момента выдачи ее первого элемента источником сообщений до момента получения последнего элемента комбинации получателем сообщений.

Контрольные вопросы и задания.

1. Какие методы защиты от ошибок вам известны?
2. Каково необходимое условие для того, чтобы код был способен обнаруживать ошибки?
3. В чем отличие понятий расстояние хемминга и кодовое расстояние?
4. Какова связь между кратностью обнаруживаемых и исправляемых ошибок и кодовым расстоянием?
5. Постройте производящую матрицу для кода, у которого $d_0=3$, $k=5$.
6. Запишите синдром для кода (7,4), для которого $a_5=a_1+a_2$, $a_6=a_1+a_3$, $a_7=a_1+a_2+a_3$, при появлении ошибки в a_3.
7. Запишите кодовую комбинацию циклического кода, у которого $p(x)=x^3+x+1$, а исходная кодовая комбинация имеет вид 1111.
8. Определите содержит или нет принятая кодовая комбинация циклического кода 1111000 ошибки, если $p(x)=x^3+x^2+1$.
9. Выполните задания п.5-8 в соответствии с данными, полученными от преподавателя.