

Министерство образования Российской Федерации

Новомосковский институт (филиал)
Государственного образовательного учреждения высшего
профессионального образования «Российский химико-
технологический университет имени Д.И. Менделеева»

Котельников А.А., Маслова Н.В.,
Вент Д.П., Котельникова М.Г.

Лабораторный практикум по предмету “Компьютерные сети”

Учебно – методическое пособие

Новомосковск
2009

УДК 681.322
ББК 32.973
В676

Рецензенты:
Иванов Б.И.

Начальник отдела вычислительной
и множительной техники ОАО “НИАП”
Васильев С.В.
Управляющий сервис-центром «Ультра»

Составители: Котельников А.А., Маслова Н.В., Вент Д.П., Котельникова М.Г.

В676 Лабораторный практикум по предмету “Компьютерные сети”. Учебно-методическое пособие. / РХТУ им. Д.И.Менделеева, Новомосковский ин-т. Сост.: Котельников А.А., Маслова Н.В., Вент Д.П., Котельникова М.Г. Новомосковск, 2009. - 71 с.

В данное пособие предназначено для студентов всех форм обучения специальности 100101 «Сервис» специализации «Компьютерные сети» и специальности 220301 «Автоматизация технологических процессов и производств».

В практикум включены 6 лабораторных работ. Каждая лабораторная работа предваряется краткими теоретическими сведениями по основам сетевых технологий, закрепление которых осуществляется при выполнении соответствующей лабораторной работы

Табл. 3 Ил. 29 .Библиогр.: 19 назв.

УДК 681.322
ББК 32.973

© Новомосковский институт (филиал)
ГОУ ВПО Российского химико-технологического
Университета им. Д.И. Менделеева, 2009

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	5
Лабораторная работа № 1 Подготовка к подсоединению ПЭВМ к локальной компьютерной сети	7
1.1 Теоретические основы	7
1.2 Необходимое оборудование	18
1.3 Порядок выполнения работы	19
1.4 Типовые задания	19
1.5 Оформление отчета по работе	19
1.6 Вопросы к защите работы	20
Лабораторная работа № 2 Настройка программного обеспечения для работы ПЭВМ в локальной компьютерной сети	21
2.1 Теоретические основы	21
2.2 Необходимое оборудование	34
2.3 Порядок выполнения работы	34
2.4 Типовые задания	35
2.5 Оформление отчета по работе	35
2.6 Вопросы к защите работы	36
Лабораторная работа № 3 Проектирование локальной корпоративной компьютерной сети	37
3.1 Теоретические основы	37
3.2 Необходимое оборудование	40
3.3 Порядок выполнения работы	40
3.4 Типовые задания	40
3.5 Оформление отчета по работе	41
3.6 Вопросы к защите работы	41
Лабораторная работа № 4 Построение и исследование компьютерных сетей с помощью системы NETCRACKER	42
4.1 Теоретические основы	42
4.2 Необходимое оборудование	52
4.3 Порядок выполнения работы	52
4.4 Типовые задания	52
4.5 Оформление отчета по работе	53
4.6 Вопросы к защите работы	53

Лабораторная работа № 5 Изучение алгоритма маршрутизации OSPF (Open Shortest Path First)	54
5.1 Теоретические основы	54
5.2 Необходимое оборудование	59
5.3 Порядок выполнения работы	59
5.4 Типовые задания	59
5.5 Оформление отчета по работе	59
5.6 Вопросы к защите работы	59
Лабораторная работа № 6 Изучение и настройка аппаратной и программной поддержки сетевого обеспечения многопроцессорной вычислительной системы на основе LINUX-машин	61
6.1 Теоретические основы	61
6.2 Необходимое оборудование	68
6.3 Порядок выполнения работы	68
6.4 Типовые задания	68
6.5 Оформление отчета по работе	69
6.6 Вопросы к защите работы	69
СПИСОК ЛИТЕРАТУРЫ	70

ВВЕДЕНИЕ

Курс «Компьютерные сети» знакомит студентов с основными принципами построения, использования и сервиса компьютерных сетей. При изучении этого курса студенты получают следующие знания и навыки:

- узнают, что компьютерные сети, для чего они создаются и как работают приложения в сетях Майкрософт и в Интернете
- познакомятся с такими понятиями, как топология сети, сетевая архитектура, стек протоколов, сетевая операционная система и сетевое приложение
- научатся строить надежную сетевую кабельную инфраструктуру, объединять компьютеры в сеть с помощью различных устройств связи
- научатся настраивать протокол TCP/IP, управлять IP-маршрутизацией и налаживать работу сетевых операционных систем.

Данное пособие поможет студентам в выполнении лабораторных работ, в ходе которых студенты на практике смогут применить знания, полученные на теоретических занятиях.

Исходные знания, умения и навыки:

Чтобы успешно выполнить лабораторные студенты должны:

- иметь представление об устройстве компьютера (процессор, память, накопители и т.д.);
- уметь работать в среде ОС Windows, запускать различные программы (в том числе с использованием командной строки) и работать с ними;
- перед выполнением каждой лабораторной работы хорошо усвоить материалы соответствующей главы учебного пособия.

Техническое оснащение для проведения лабораторных работ:

- Компьютеры, удовлетворяющие минимальным требованиям для установки операционных систем Windows 2000 Professional / Server, Windows XP Professional или Windows Server 2003.
- Сетевой адаптер (PCI-совместимый стандарта 10/100/1000 Мб/с Ethernet).
- Дополнительный сетевой адаптер.

Сетевое обеспечение и оборудование:

- отрезок коаксиального кабеля RG-58 длиной не менее 1 м;
- набор BNC-коннекторов (как минимум два обычных, два T-коннектора и два терминатора);
- клещи для заделки BNC-коннекторов;
- два или более PCI-совместимых сетевых адаптера с BNC портами;
- гибридный 10-мегабитный концентратор с несколькими портами 10Base-T и одним портом 10Base-2;
- несколько отрезков кабеля типа «витая пара» категории 5;
- несколько коннекторов RJ-45;
- клещи для заделки коннекторов RJ-45;
- тестер для проверки качества заделки коннекторов;
- два или более 10/100-мегабитных концентратора или 10/100/1000-мегабитных коммутатора (можно использовать небольшие устройства с 5–8 портами).
- один или несколько сетевых адаптеров Wi-Fi, предназначенных для установки в разъем PCI или в порт USB
- беспроводная точка доступа Wi-Fi, подключаемую к сети Ethernet.

ЛАБОРАТОРНАЯ РАБОТА № 1

ПОДГОТОВКА К ПОДСОЕДИНЕНИЮ ПЭВМ К ЛОКАЛЬНОЙ КОМПЬЮТЕРНОЙ СЕТИ

Цель работы – приобретение практических знаний и навыков в выборе и установке сетевых адаптеров, монтажу и разделке сетевого кабеля, физическому присоединению ЭВМ к кабельной системе при создании локальной компьютерной сети по технологии Ethernet.

1.1 Теоретические основы.

Сетевой стандарт Ethernet был разработан в 1975 г. в исследовательском центре корпорации Xerox, после чего доработан совместно DEC, Intel и XEROX (отсюда сокращение DIX) и впервые опубликован как “Blue Book Standart” для Ethernet I в 1980 г. Этот стандарт получил дальнейшее развитие и в 1985 г. вышел новый – Ethernet II (известный также как DIX).

На основе стандарта Ethernet DIX был разработан стандарт IEEE 802.3, одобренный в 1985 году для стандартизации комитетом по LAN IEEE (*Institute of Electrical and Electronics Engineers*). В зависимости от вида физической среды передачи данных стандарт IEEE 802.3 имеет модификации (число 10 в начале каждой обозначает скорость передачи данных 10 Мбит/сек):

- 10Base-5 (коаксиальный кабель диаметром 0,5 дюйма –*толстый коаксиал* с волновым сопротивлением 50 ом; максимальная длина сегмента сети без повторителей 500 м, считается бесперспективным).
- 10Base-2 (коаксиальный кабель диаметром 0,25 дюйма –*тонкий коаксиал*, волновое сопротивление 50 ом; максимальная длина сегмента сети без повторителей 185 м, считается бесперспективным).
- 10Base-T (кабель на основе неэкранированной витой пары – UTP, *Unshielded Twisted Pair*; физическая топология – звезда с концентратором в центре, максимальное расстояние между концентратором и конечным узлом – до 100 м).
- 10Base-F (двухволоконный волоконно-оптический кабель, топология сети аналогична 10Base-T; варианты: FOIRL допускает расстояние до 1000 м, 10Base-FL и 10Base-FB – до 2000 м).

В 1995 г. принят стандарт Fast Ethernet (IEEE 802.3u), в 1998 г. – Gigabit Ethernet (IEEE 802.3z), в 2002 г. - 10 Gigabit Ethernet (IEEE 802.3ae). Ethernet и Fast Ethernet применяют один и тот же метод раз-

деления среды передачи данных CSMA/CD (*Carrier Sense Multiple Access with Collision Detection*, метод коллективного доступа с опознаванием несущей и обнаружением коллизий).

Кабель UTP является наиболее дешевым (при обеспечении достаточной скорости передачи данных и простоте монтажа). UTP-кабели категории 1 применяются в основном для телефонной разводки, UTP категории 3 служат для передачи как голоса так и данных при невысокой производительности (диапазон часто до 16 МГц). Для высокоскоростных протоколов при передаче на большие расстояния могут применяться (более дорогие) кабели UTP категорий 6 и 7 (экран вокруг каждой пары и вокруг всех жил соответственно, рабочие частоты до 300 и 600 МГц).

В настоящее время при создании локальных компьютерных сетей практически всегда (для технологий Ethernet, Fast Ethernet и Gigabit Ethernet) применяют кабель UTP категории 5 (8 попарно скрученных медных жил, активное сопротивление не более 9,4 ом на 100 м, полное волновое сопротивление 100 ом на частоте 100÷120 МГц, затухание сигнала 0,8÷22 дБ на частотах от 64 кГц до 100 МГц). Каждый провод кабеля UTP маркирован цветом (синий и белый с синими полосками, оранжевый и белый с оранжевыми полосками, зеленый и белый с зелеными полосками, коричневый и белый с коричневыми полосками по скрученным парам соответственно), для UTP-кабеля применяются разъемы RJ-45 (рис.1.1).

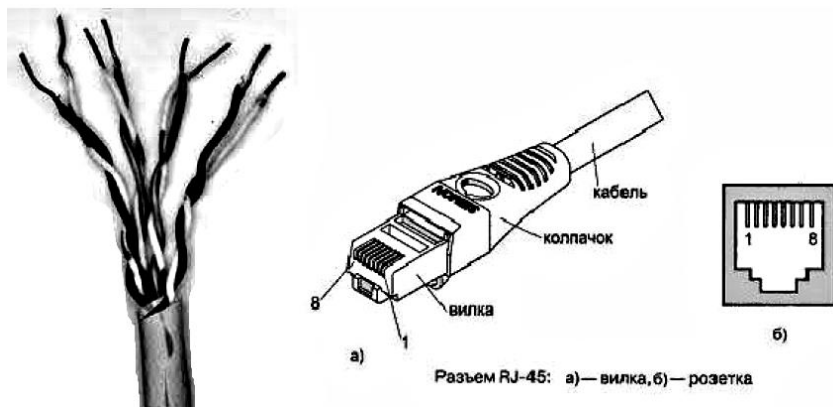


Рис.1.1 Кабель UTP категории 5 (слева) и разъем RJ-45, показаны вилка (plug) и розетка (jack).

Отрезок UTP-кабеля (обычно не более 5 метров) со смонтированными на его концах вилками RJ-45 называют Patch cordом. Вилки RJ-45 являются не-разборными, при необходимости кабель просто отрезают около вилки и монтируют новую.

Для технологии Ethernet используется топология “звезда” с концентратором в центре, причем определены порты типа MDI (*Medium Depended Interface, разъем сетевого адаптера*) и MDIX (*MDI crossing, разъем портов сетевого концентратора*), см. рис.1.2. При соединении MDI-MDIX (подключение конечных узлов сети к портам активного оборудования) используется ‘прямой’ кабель (рис.1.3а), при соединении MDI-MDI (непосредственное соединение адаптеров компьютеров, рис.1.2б) или MDIX-MDIX (соединение двух коммуникационных устройств) используют ‘перекрестный’ (*крассовый*) кабель (рис.1.3б, причем на рис.1.2 “перекрестный” кабель обозначен символом **x**).

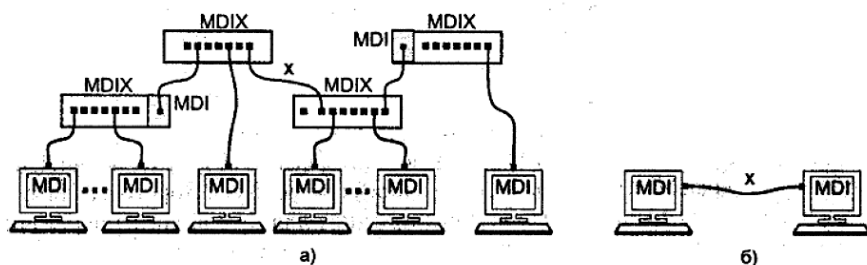


Рис.1.2 Сеть 10BaseT/100BaseTX.

а) – звезда, б) – непосредственное соединение двух компьютеров (двухточечное соединение)

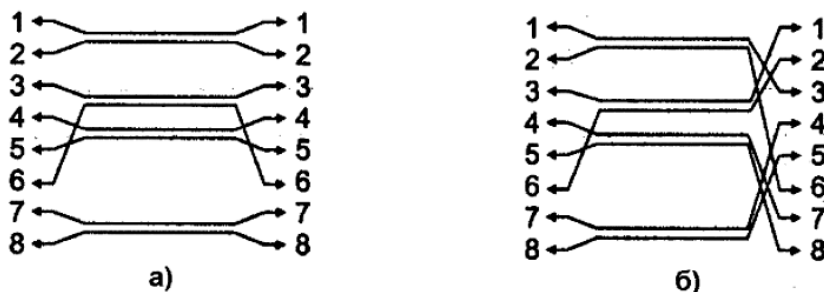


Рис.1.3 Интерфейсные кабели Ethernet.

а) – “прямой”, б) – “перекрестный” (крассовый)

В Gigabit Ethernet 1000BaseTX применяют только “прямые” кабели (в случае использования “перекрестного” кабеля скорость связи установится 100 Mbit/сек). Впрочем, большинство современных коммутаторов используют функцию автоопределения типа кабеля (MDI или MDIX), что почти исключает вероятность ошибочного подсоединения.

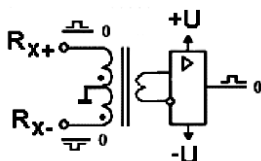
В 10- и 100-мегабитном Ethernet’е (10BaseT/100BaseTX) названия контактов содержат символы **TX** (*transmitter*, передатчик), **RX** (*receiver*, приемник) со знаками ‘+’ и ‘-’ и из 8 жил используется только половина (рис.1.3); для Gigabit Ethernet (1000BaseTX) используются все 8 медных жил (обмен данными по 4 парам жил в обоих направлениях одновременно), подсоединение соответствует табл.1.1.

Таблица 1.1

Разъем RJ-45 адаптера Ethernet.

Контакт	10BaseT/100BaseTX	1000BaseTX
1	TX+	BI_D1+
2	TX-	BI_D1-
3	RX+	BI_D2+
4	не подсоединен	BI_D3+
5	не подсоединен	BI_D3-
6	RX-	BI_D2-
7	не подсоединен	BI_D4+
8	не подсоединен	BI_D4-

Сигналы по каждой двухпроводной линии пере даются дифференциальным способом (с противоположной полярностью по линиям ‘+’ и ‘-’), причем входные и выходные цепи сетевых адаптеров имеют гальваническую развязку:



Кабель UTP соединяется с вилкой RJ-45 без применения пайки. При монтаже вилки RJ-45 на кабель UTP-5 удаляют внешнюю оболочку кабеля на длину полудюйма (12,5 мм, см. рис.1.4б); для удаления оболочки на специальном инструменте (рис.4а) имеется специальный нож и ограничитель длины удаляемой оболочки. Снимать изоляцию с жил не нужно, однако жилы следует расположить на плоскости в соответствии со схемой заделки (правое изображение из рис.1.4б и ниже следующие схемы).

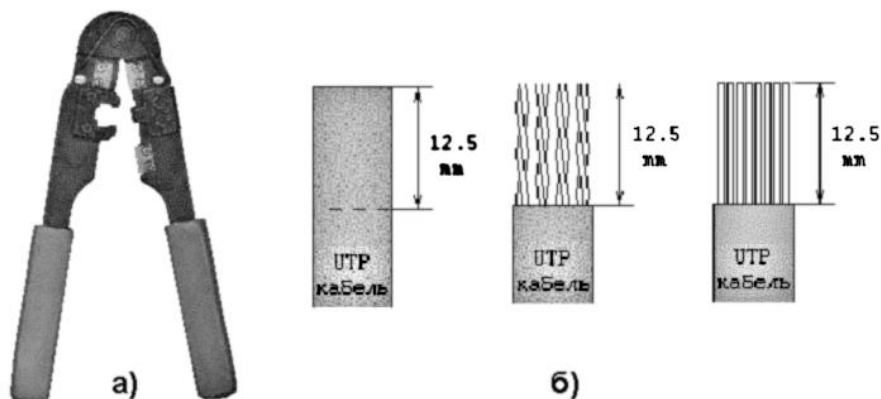
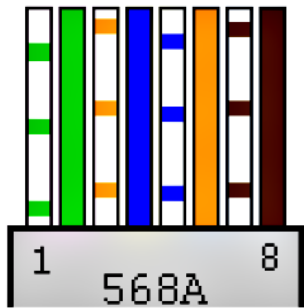
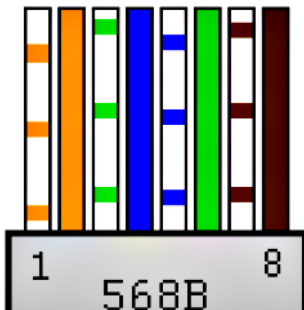



Рис.1.4 Обжимной инструмент для разделки UTP-кабеля (а) и последовательность снятия внешней оболочки с сетевого кабеля (б).

Варианты заделки проводов (разводка проводов витой пары) показаны в табл.1.2 (“прямой” кабель). В качестве схем заделки для 8-ми жильного кабеля равноценно можно использовать схему 568А или 568В (но одинаковую для данной сети, рекомендуется первая), для 4-х жильного кабеля используется схема согласно последнему из рисунков.

После описанного расположения жил на плоскости следует повернуть вилку контактами к себе (как на рис.1.5) и аккуратно надвинуть на кабель до упора, чтобы провода прошли под контактами. Вид вилки с кабелем внутри показан на рис.1.5в.

Варианты заделки проводов

Стандарт EIA/TIA-568A (8-ми жильный “прямой” кабель, схема 568A)		
Номера контактов	Цвет оболочки провода	
1	белый с зелеными полосками	
2	Зеленый	
3	белый с оранжевыми полосками	
4	Синий	
5	белый с синими полосками	
6	Оранжевый	
7	белый с коричневыми полосками	
8	Коричневый	
Стандарт EIA/TIA-568B, AT&T 258A (8-ми жильный “прямой” кабель, схема 568B)		
1	белый с оранжевыми полосками	
2	оранжевый	
3	белый с зелеными полосками	
4	Синий	
5	белый с синими полосками	
6	Зеленый	
7	белый с коричневыми полосками	
8	Коричневый	
Стандарт 10Base-T/100Base-TX (4-х жильный “прямой” кабель)		
1	белый с оранжевыми полосками	
2	оранжевый	
3	белый с зелеными полосками	
4	синий	

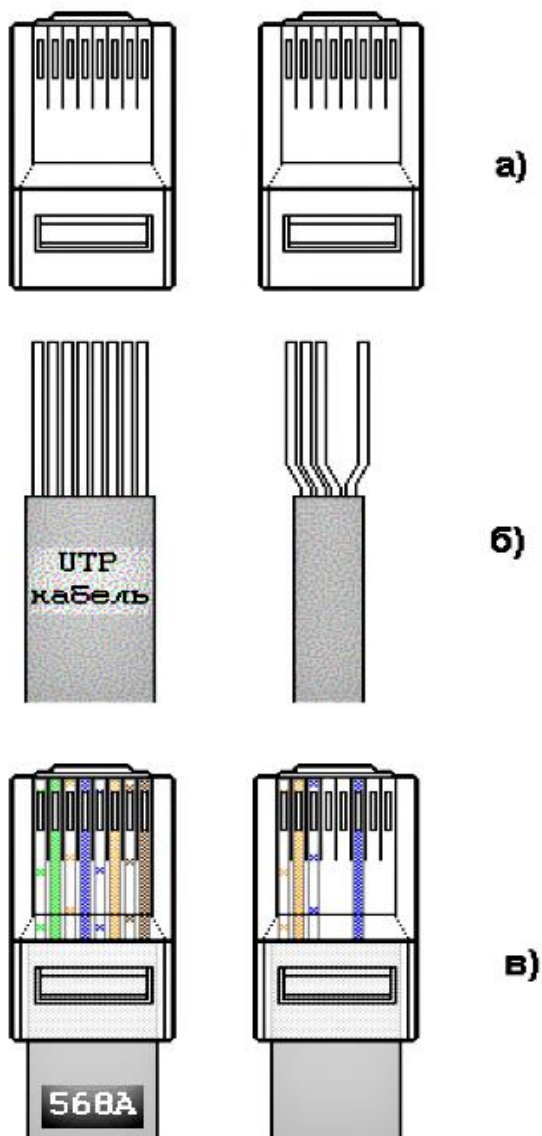


Рис.1.5 Расположение вилки RJ-45 контактами к себе(а), разделанного кабеля UTP (б) и вид вилки с кабелем внутри (слева – 8-ми жильный кабель, справа – 4-х жильный).

Последним действием является обжим вилки. На обжимном инструменте имеется специальное гнездо, в которое вставляется вилка с проводами, после чего нажатием на ручки инструмента вилка обжимается (рис.1.6а). При этом контакты (стрелка 2 на рис.1.6) будут утоплены внутрь корпуса, прорежут изоляцию проводов и обеспечат надежный контакт с жил кабеля с контактами вилки. Фиксатор провода также должен быть утоплен в корпус (нажатие по стрелке 1 на рис.1.6).

В крайнем случае (если нет обжимного инструмента) можно обжать разъем RJ-45 тонкой отверткой (рис.1.7). При этом следует утопить все 8 шт. контактов (1) в корпус, а затем утопить и фиксатор провода (3). Полезно подложить что-либо под разъем, чтобы не сломать его фиксатор (2). Это не есть самый надежный способ монтажа, но приемлемый.

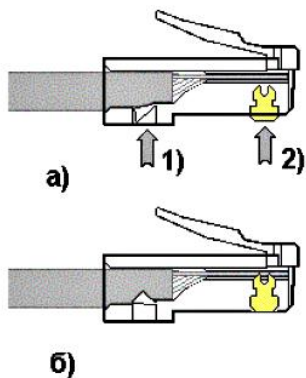


Рис.1.6 Обжим вилки.

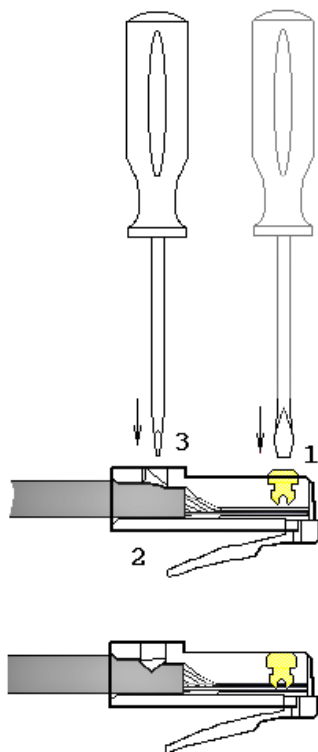
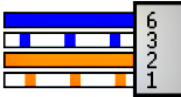

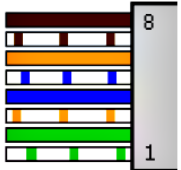
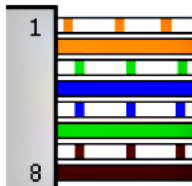
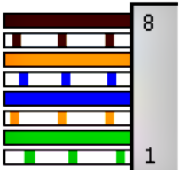
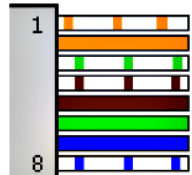


Рис.1.7 Обжим тонкой отверткой

Для непосредственного соединения двух компьютеров можно рекомендовать показанное в табл.1.3 соединение (“перекрестный” кабель). Приведен вариант 4-х жильного так называемого “нуль-хабного кабеля”.

Таблица 1.3

Соединение “перекрестный” кабель (“нуль-хабный кабель”)

“Cross-over” (“нуль-хабный”) кабель (4-х жильный UTP-кабель)				
	одна сторона	Цвет провода	вторая сторона	
	1	белый с оранжев. полосками	3	
	2	оранжевый	6	
	3	белый с синими полосками	1	
	6	синий	2	
“Нуль-хабный” кабель (8-ми жильный UTP-кабель) (первый вариант)				
	1	белый с зелеными полосками	3	
	2	зеленый	6	
	3	белый с оранжев. полосками	1	
	4	синий	4	
	5	белый с синими полосками	5	
	6	оранжевый	2	
	7	белый с коричн. полосками	7	
	8	коричневый	8	
“Нуль-хабный” кабель (8-ми жильный UTP-кабель) (второй вариант)				
	1	белый с зелеными полосками	3	
	2	зеленый	6	
	3	белый с оранжев. полосками	1	
	4	синий	7	
	5	белый с синими полосками	8	
	6	оранжевый	2	
	7	белый с коричн. полосками	4	
	8	коричневый	5	

При тщательном выполнении монтажа вилок RJ-45 достигается устойчивый контакт между жилами кабеля и контактами вилки. В редких случаях (выявляемых обычно уже на этапе настройки программного обеспечения поддержки сети) требуется проверка физического соединения портов (выполняется с помощью *кабельных тестеров* или просто омметром). В состав ПО сетевых карт некоторых производителей включены утилиты (напр., *Virtual Cable Tester* фирмы 3Com), позволяющие определить место неуверенного контакта в кабеле или разъеме (используется явление отражения сигнала в кабеле).

Сетевые кабели обычно соединяют сетевые карты или карты с сетевыми устройствами. На сетевых картах располагается ответная часть вилки RJ-45 – гнездо разъема RJ-45. Гнезда устанавливаются в сетевые карты, концентраторы, коммутаторы, трансиверы и другие устройства. Сам разъем представляет собой ряд из 8 пружинящих контактов и выемку для фиксатора вилки. Если смотреть на гнездо со стороны контактов (причем они располагаются сверху), то отсчет идет слева направо (рис.1.1, справа, б).

Розетка представляет собой гнездо (разъем) соединителя с каким-либо приспособлением для крепления кабеля и корпусом для удобства монтажа, обычно в комплекте поставляется и вилка. Внешняя розетка представляет собой небольшую пластмассовую коробочку, к которой прилагается шуруп и двухсторонняя наклейка для монтажа на стену. Такая розетка служит окончанием сетевого кабеля, обычно разводимого по стене помещения и помещенного в коробах. В розетках типа **KRONE** для монтажа кабеля UTP-5 используется специальная пластина с щелью, в которую заталкивается провод, при этом прорезается изоляция и жила кабеля входит в надежный контакт с пластиной (пайка не применяется). Для монтажа проводов имеется специальный инструмент, который помимо заталкивания проводов в щель обрезает лишние его куски. В любом случае настоятельно рекомендуется после тщательного замера длины кабеля оставить по $1 \div 1,5$ м с каждой стороны для монтажа и укладки части кабеля в непосредственной близости от компьютера (или иного сетевого устройства).

Сетевая карта или сетевой адаптер (NIC, *Network Interface Card*) - плата расширения, обычно вставляемая в разъем системной (материнской) платы (*main board*) компьютера; современные системные платы обычно имеют встроенную сетевую карту. На рис.1.8 показана сетевая карта шины данных PCI.

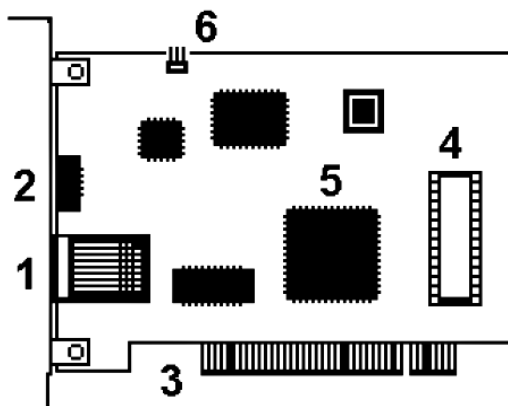


Рис.1.8 Сетевая карта шины данных PCI.

1 - разъем под витую пару (RJ-45), 2 – светодиодный индикатор активности сети, 3 – шина данных PCI, 4 - панелька под микросхему BootROM (для загрузки операционной системы компьютера не с локального диска, а с сервера сети), 5 - микросхема контроллера платы, 6 – коннектор подключения 3-х проводного кабеля к системной плате для “пробуждения” по сети (Remote Wake Up; для этого передается специальный кадр Magic Packet, при приеме которого ПЭВМ “просыпается”).

Для определения точки назначения пакетов в сети Ethernet используется MAC (*Media Control Access*)-адрес. Это уникальный серийный номер, присваиваемый каждому сетевому устройству Ethernet для идентификации его в сети. MAC-адрес присваивается адаптеру его производителем, но может быть программно изменен. В обычном режиме работы сетевые адаптеры просматривают весь проходящий сетевой трафик и ищут в каждом пакете свой MAC-адрес. Если такой находится, то устройство (адаптер) обрабатывает этот пакет. MAC-адрес имеет длину 6 байт (48 бит) и обычно записывается в шестнадцатичном виде, например, **12:34:56:78:90:AB** (двоеточия между байтами делают число более читабельным).

Каждый производитель присваивает адреса из принадлежащего ему диапазона адресов. Первые три байта адреса определяют производителя, например.:

- 00000C - Cisco
- 00000E - Fujitsu

- 00001D - Cabletron
- 00004C - NEC Corporation
- 000061 - Gateway Communications
- 000062 - Honeywell
- 0080C8 - D-Link
- 00A024 - 3Com
- 00C049 - US Robotics

Обычно все поддерживающие высшие скорости обмена данными сетевые адаптеры работают и на меньших скоростях (если комплементарное устройство не поддерживает данной скорости, но совместимо по стандарту Ethernet). Позволяет это *протокол согласования режимов* (autonegotiation, процесс основан на обмене специальными служебными импульсами), выполняемый каждый раз при установлении соединения после физического подключения (при инициализации портов) и позволяющий выбрать наиболее эффективный из режимов, доступных обоим портам.

Для обеспечения корректной работы каждой сетевой платы необходимо определить для нее *адрес ввода-вывода* (In/Out port) и *номер прерывания* (IRQ). Конфигурирование сетевой платы заключается в настройке ее на свободные адрес и прерывание, которые затем будут использоваться операционной системой. Адрес (In/Out port) и прерывание (IRQ) для каждой сетевой платы должно быть отличным от других устройств компьютера. Современные сетевые карты поддерживают технологию Plug-and-Play и автоматически выполняют эту операцию, при ручной установке полезной оказывается входящая в состав NT-совместимых версий Windows утилита **WINMSD.EXE** ("*Сведения о системе*"). Программная поддержка сетевых карт обеспечивается *драйверами*, для Windows возникновение проблем с драйверами маловероятно.

1.2 Необходимое оборудование

- IBM PC-совместимая ЭВМ
- сетевая карта (для шины данных PCI) производительностью 10÷100 Mbit/сек с разъемом RJ-45
- кабель UTP категории 5
- вилки RJ-45
- обжимной инструмент.

1.3 Порядок проведения работы.

Монтаж предполагает разделку концов кабеля в, обжим вилки RJ-45, проверка контакта осуществляется кабельным тестером или омметром.

Сетевая карта устанавливается в свободный разъем на системной плате ПЭВМ (если на последней не имеется встроенного сетевого адаптера), при включении ПЭВМ конфигурируется технологией Plug-end-Play. Используемые адреса ввода-вывода (In/Out port) и номер прерывания (IRQ) фиксируются. Сетевой кабель подсоединяется к заданным устройствам, комплексная проверка сетевого соединения проводится с использованием специализированного программного обеспечения (см. описание работы 2 данного практикума).

1.4 Типовые задания.

1. Смонтировать 4-х жильный UTP-кабель для соединения ПЭВМ с сетевым устройством (концентратором, коммутатором).
2. Смонтировать 8-ми жильный UTP-кабель для соединения ПЭВМ с сетевым устройством (концентратором, коммутатором).
3. Смонтировать 4-х жильный UTP-кабель для непосредственного соединения двух ПЭВМ.
4. Смонтировать 8-ми жильный UTP-кабель для непосредственного соединения двух ПЭВМ.

1.5 Оформление отчета по работе.

В отчете указываются параметры выполняемого задания (соединяемые сетевые устройства, тип кабеля, число жил) и выбранные студентом схемы соединения. Приводятся эскизы зачистки оболочки кабеля, последовательность расположения жил при монтаже вилки RJ-45, обжимного инструмента. Указывается модель сетевой карты (при возможности определить MAC-адрес и по нему фирму-изготовителя) и поддерживаемые скорости обмена данными, тип шины данных. При возникновении проблем с контактом в кабеле следует привести схему проверки надежности контакта.

1.6 Вопросы к защите работы

1. Какие сетевые кабели использует технология Ethernet? Что такое кабель UTP? В чем его достоинства и недостатки?
2. Что такое сетевые устройства MDI и MDIX? Для соединения каких устройств необходим “перекрестный” (кроссированный) кабель?
3. Почему при монтаже вилки RJ-45 на кабель нет необходимости снимать изоляцию с отдельных жил кабеля?
4. Что такое “нуль-хабный” кабель и для каких целей он применяется?
5. Каким образом однозначно идентифицируются сетевые адаптеры? С какой целью введена возможность изменения MAC-адреса?
6. В чем суть протокола согласования режимов работы сетевых портов?
7. В чем заключается процесс конфигурирование сетевой платы? Какие параметры при этом настраиваются?

ЛАБОРАТОРНАЯ РАБОТА № 2.

НАСТРОЙКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ РАБОТЫ ПЭВМ В ЛОКАЛЬНОЙ КОМПЬЮТЕРНОЙ СЕТИ

Цель работы – приобретение практических знаний и навыков в настройке программного обеспечения (ПО) ПЭВМ для обеспечения функционирования в составе локальной компьютерной сети.

2.1 Теоретические основы.

подавляющее большинство IBM PC- совместимых ЭВМ работают в локальных сетях, использующих стек протоколов TCP/IP. В данной работе рассматриваются вопросы, связанные с установкой, конфигурированием и проверкой функционирования таких сетей. Дополнительные сведения по данной теме можно получить по адресам:

- http://ixbt.com/comm/lan_faq.html
- <http://www.vgts.ru/doc/tcpip.html>
- <http://www.citforum.ru/nets/ip/contents.shtml>
- <http://www.3com.com/nsc/501302.html>

При объединении ПЭВМ исключительно внутри организации (без выхода в InterNet) можно использовать абсолютно любые допустимые IP-адреса. В RFC 1597 (Address Allocation for Private Internets, March 1994) для частных (private) и внутрикорпоративных сетей с выходом в InterNet рекомендуется использовать IP-адреса из следующих диапазонов (роутеры большинства InterNet-провайдеров эти адреса не маршрутизируют, что добавляет адресам безопасность):

- **10.0.0.0 ÷ 10.255.255.255** - 1 сеть класса А
- **172.16.0.0 ÷ 172.31.255.255** - 16 сетей класса В
- **192.168.0.0 ÷ 192.168.255.255** - 256 сетей класса С

При этом адреса **192.168.x.0** и **192.168.x.255** (подобные **192.168.0.0** и **192.160.0.255**, **192.168.1.0** и **192.168.1.255** и так далее) не применяются.

Например, в сети используются два компьютера. Каждому из них (а точнее, сетевой карте каждого) необходимо назначить индивидуальный IP-адрес. Для первой ПЭВМ назначен IP-адрес **192.168.0.1** и маска подсети (subnet mask) **255.255.255.0**, второй IP-адрес **192.168.0.2** и аналогичная сетевая маска. При подключении третьей ПЭВМ целе-

сообразно определить IP-адрес **192.168.0.3** и такую же сетевую маску (далее в круглых скобках). Для остальных соответственно **192.168.0.4 (255.255.255.0)** для четвертого, **192.168.0.5 (255.255.255.0)** - для пятого и так далее.

Нижеописанная настройка клиентского сетевого ПО будет конкретизирована в применении к русскоязычному варианту ОС Windows'XP и для случая, когда соответствующее ПО еще не было установлено.

Первым этапом является определение типа сетевой карты (карт) данной ПЭВМ, причем современные карты поддерживают технологию Plug-and-Play и автоматически (без вмешательства пользователя) устанавливают адреса ввода-вывода (In/Out port) и номер прерывания (IRQ). На этом этапе реален запрос устанавливающей системы заново вставить в устройство чтения лазерных дисков дистрибутива Windows или иного носителя с драйверами данного сетевого адаптера (если соответствующий драйвер отсутствует на диск-дистрибутиве Windows).

Собственно настройка сети начинается последовательным выбором вариантов меню (рис.2.1):

**Пуск→Программы →Стандартные→Связь→
→Мастер настройки сети.**

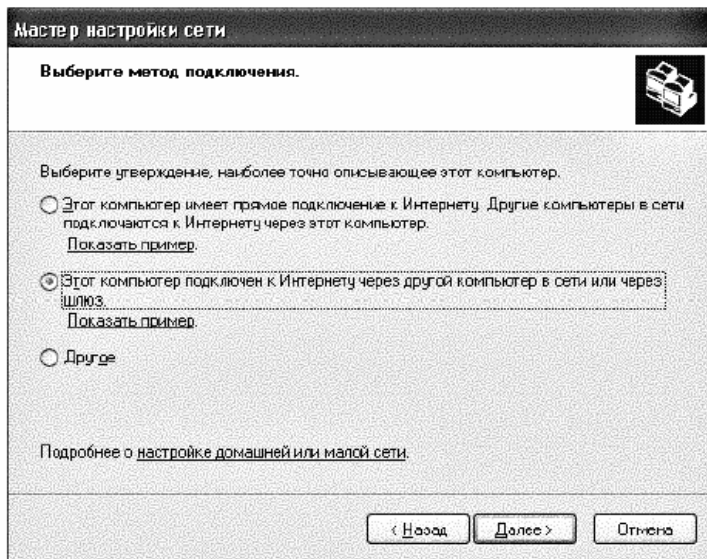


Рис.2.1 Окно мастера настройки сети

В окне мастера настройки сети выбирается метод использования сети (например, прямое подключение к InterNet'у или подключение через другой компьютер или через шлюз – на рис.2.1 показан выбор именно последнего варианта настройки). Для перехода к очередному окну служит кнопка “Далее”.

На рис.2.2 представлен встроенный в системную плату ПЭВМ сетевой адаптер 3Com Gigabit 3C940 (вообще говоря, сетевых адаптеров может быть несколько). Далее необходимо ввести текстовое описание и имя компьютера и, наконец, название рабочей группы ЭВМ (все компьютеры данной сети должны иметь именно это имя рабочей группы); кириллические символы использовать при этом рискованно. На следующем этапе можно создать диск настройки сети (для быстрого восстановления настроек после критических ситуаций) или завершить работу мастера настройки сети; для вступления в силу настроек необходимо перезагрузить ПЭВМ.

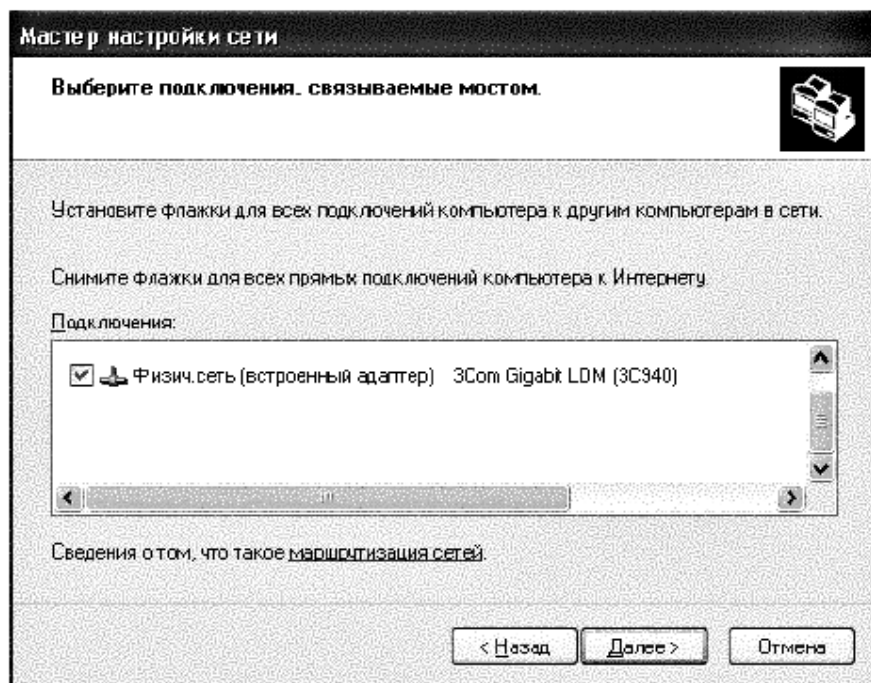


Рис.2.2 Варианты сетевых подключений для выбора

Далее следует выбрать сетевые протоколы. Для перехода к окну выбору протоколов можно использовать последовательность:

**Пуск→Программы→Стандартные→Связь→
→Сетевые подключения**

Щелкнуть правой кнопкой мыши на символе:



и выбрать вариант “Свойства” из выпадающего списка (рис.2.3).

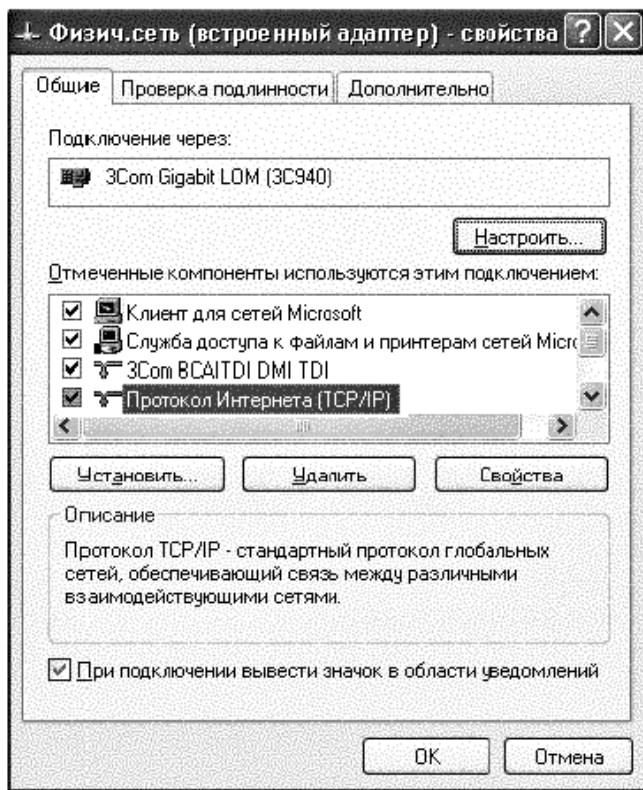


Рис.2.3 Список свойств выбранного подключения

По умолчанию в окне выбора протоколов устанавливаются “Клиент для сетей Microsoft”, “Служба доступа к файлам и принтерам сетей Microsoft” и “Протокол Интернета (TCP/IP)”. Обычно стека протоколов TCP/IP достаточно для работы, только в некоторых случаях требуется выбрать IPX/SPX (стек протоколов для сетей NetWare разработки фирмы Novell) или NetBEUI (транспортный протокол локальной сети, созданный для работы совместно с сетевым интерфейсом NetBIOS фирмы Microsoft Corp).

Нажатие кнопки “Свойства” при подсвеченной строке с протоколом TCP/IP дает возможность настроить необходимые статические IP-адреса стека TCP/IP (рис.2.4). В качестве DNS-сервера можно указать, например **194.190.241.65** (ресурс <http://ch.informica.ru>). Если ПЭВМ оснащена несколькими сетевыми картами, процедуру настройки IP-адресов следует повторить для каждого контроллера.

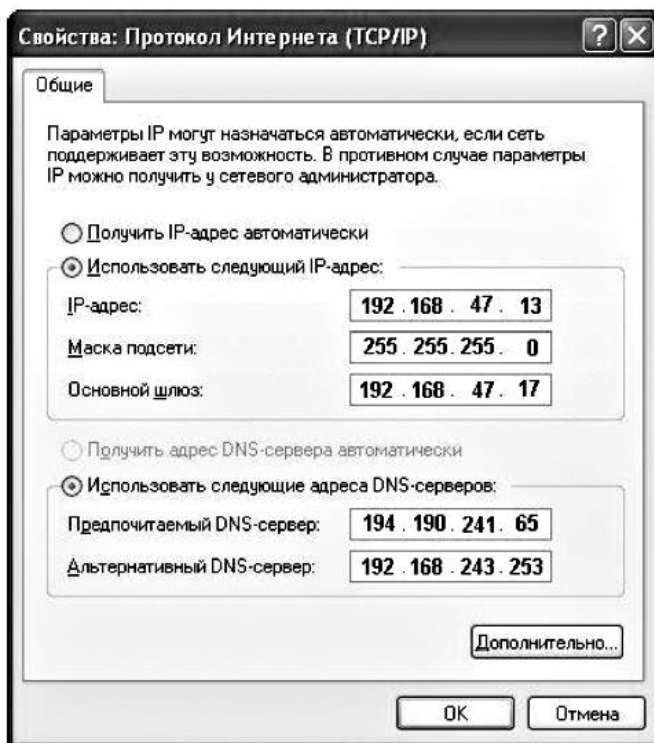


Рис.2.4 Настройка статического IP-адреса стека TCP/IP.

Через кнопку “Дополнительно” доступна настройка протокола **DHCP** (Dynamic Host Configuration Protocol) и службы **WINS** (Windows Internet Naming Service), позволяющей определить IP-адрес ПЭВМ по заданному NetBIOS-имени и др.

При необходимости можно изменить имя данного компьютера и рабочую группу, в который он включен (рис.2.5). Возможно также включение компьютера в домен (сообщество ЭВМ, обладающее сходным уровнем защиты и привилегиями).

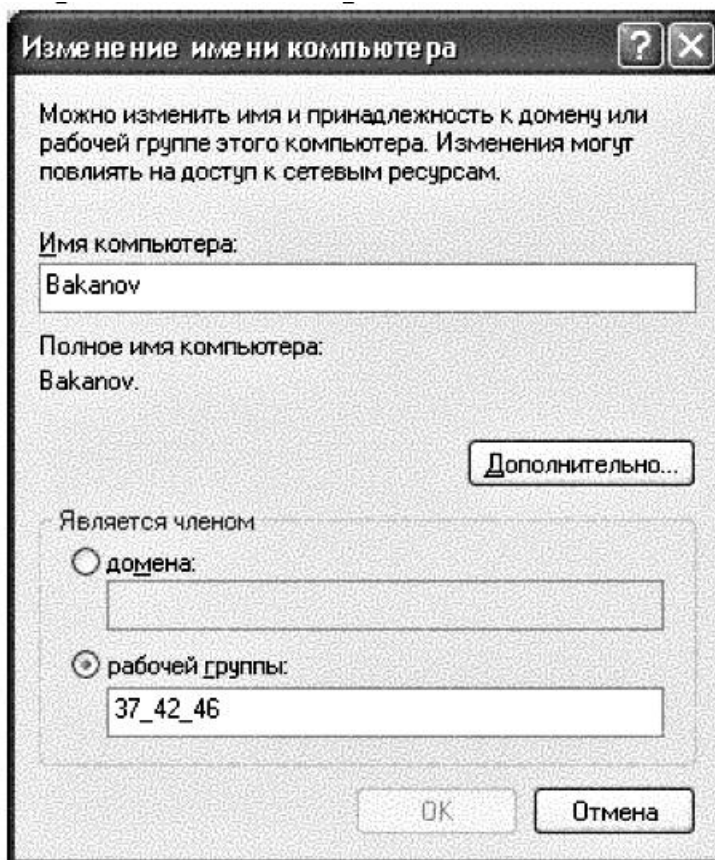


Рис.2.5 Окно изменения имени компьютера.

Для полного вхождения в силу сетевых настроек придется перезагрузить ПЭВМ.

При использовании броузера для работы с сетью InterNet следует настроить IP-адреса также и для него. Для броузера Microsoft Internet Explorer настройка производится в окне **“Настройка локальной сети”** (доступ к окну из главного меню броузера можно получить через последовательность: **Сервис→Свойства обозревателя→ Подключения→Настройка локальной сети**). При использовании прокси-сервера в локальной сети и назначении статического IP-адреса надо указать его и номер порта (**192.168.47.17** и **3128** соответственно рис.2.6), без использования прокси устанавливается только IP данной машины (при этом стандартным является использование порта 80). В большинстве случаев полезно установить флажок **“Не использовать прокси для локальных адресов”**.

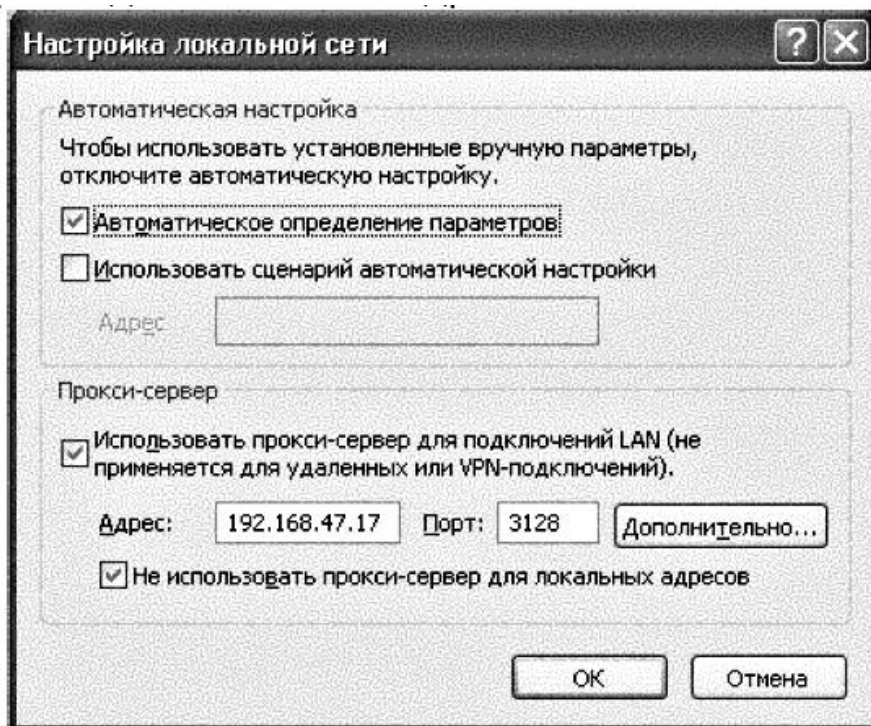


Рис.2.6 Использование прокси – сервера.

Для подключения компьютера к сетевым ресурсам удаленных ЭВМ следует щелкнуть правой кнопкой мыши на ссылке “Сетевое окружение” на рабочем столе, из выпадающего меню выбрать вариант “Подключить сетевой диск...” и настроить необходимые параметры в окне “Подключение сетевого диска” (рис.2.7).

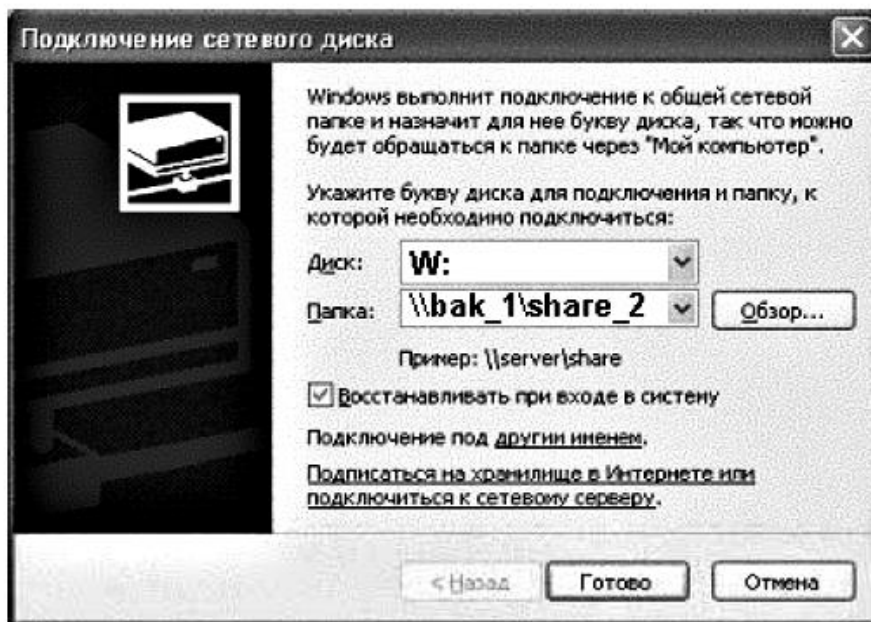


Рис.2.7 Подключение сетевого диска.

Формат описания ресурса известному шаблону:

\\имя_компьютера\имя_ресурса

с обязательным использованием двойного обратного слэша “\\” перед сетевым именем компьютера (доступ к конкретному ресурсу из списка может потребовать ввода имени и пароля). После настройки доступ к ресурсу будет возможен через список “Мой компьютер” (согласно рис.2.7 обращение к диску “W:” равносильно обращению к каталогу **share_2** компьютера **bak_1**).

При диагностировании и настройке сетевых соединений удобно пользоваться стандартными утилитами, применяющим интерфейс командной строки (нижеприведенные имена утилит даны в применении к ОС Windows, базированными на технологии NT). Указанные утилиты используют заданные при настройке сетевого ПО адреса DNS-серверов для разрешения (преобразования) доменных имен в IP-адреса.

Утилита **ping**.

Утилита **ping** тестирует сетевое соединение путем послыки ICMP-пакетов типа 8 (запрос эха), на которые получатель отвечает ICMP-пакетом типа 0 (эхо-ответ). С помощью этой утилиты удобно проверять наличие пути до заданного узла и определять временные характеристики этого пути. Утилите **ping** достаточно указать IP-адрес или DNS-имя, однако имеется ряд параметров, позволяющих более тонко управлять ее работой. Утилита **ping** выводит результат каждого запроса/ответа на отдельной строке, а перед завершением работы выдает статистику: минимальное, максимальное и среднее время передачи пакета, количество и долю потерянных пакетов. Фактически **ping** является “рабочей лошадкой” при тестировании сетевых соединений.

Общий формат использования утилиты (находящиеся в квадратных скобках параметры опциональны):

```
ping [-t] [-a] [-n число] [-l размер] [-f] [-i TTL] [-v TOS]  
[-r число] [-s число] [[-j списокУзлов] | [-k списокУзлов]]  
[-w таймаут] конечноеИмя
```

Для получения такой подсказки достаточно запустить **ping** без параметров, для вывода подсказки в файл **ping_test.txt** следует использовать **ping > ping_test.txt** (то же относится и к большинству иных утилит).

Параметр “-t” включает постоянную проверку связи до нажатия **Ctrl+C**. При нажатии **Ctrl+Break** выводится статистически накопленная информация и работа продолжается (обычно этот параметр используют, чтобы как можно быстрее узнать о наличии связи с заданным узлом).

Параметр “-a” требует определение IP-адреса по имени узла (по умолчанию не выполняется).

Параметр “-n <число>” позволяет задать количество запросов (по умолчанию **четыре** запроса).

Параметр “-l <число>” дает возможность задать размер пакета (по умолчанию размер пакета **64** байта).

Параметр “-f” позволяет установить в запросах флаг “не фрагментировать”. Используется в сочетании с параметром “-I” для обнаружения сетей с малым размером кадра, для передачи через которые IP-пакеты приходится фрагментировать.

Параметр “-i <число>” задает время жизни пакета (TTL), по умолчанию у ICMP-пакетов время жизни равно **255**.

Параметр “-r <число>” дает возможность получить маршрут, по которому передавались запрос и ответ (показать маршрутизацию). Числовой параметр может быть от **1** до **9** и определяет максимальное количество узлов, которые будут показаны в маршруте.

Параметр “-w <число>” позволяет задать время ожидания каждого пакета (в миллисекундах), по умолчанию это **1000** миллисекунд.

Утилита **tracert**

Утилита **tracert** дает возможность проследить маршрут пакетов до заданного узла и получить временные характеристики для каждого промежуточного маршрутизатора на этом пути. Эта утилита, как и ранее описанная **ping**, отправляет серию пакетов ICMP типа 8, но с разными значениям TTL: сначала отправляется три пакета с **TTL=1** (на эти пакеты ближайший маршрутизатор ответит пакетами ICMP типа **11** (истекло время передачи), из которых будет извлечен его адрес), затем с **TTL=2** (на эти пакеты ответит второй маршрутизатор) и так далее до тех пор, пока не будет достигнут заданный узел или значение TTL не превысит порог. Для каждого TTL утилита выводит по одной строке с адресом маршрутизатора (и, возможно, с его доменным именем - если удалось его разрешить) и тремя значениями времени, которое понадобилось для передачи пакета. Формальный синтаксис:

tracert [-d] [-h максЧисло] [-j списокУзлов] [-w интервал] имя.

Параметр “-d” позволяет (принудительно) не выполнять разрешение IP-адресов маршрутизаторов в доменные имена, это позволяет ускорить работу утилиты за счет отмены обращения к службе DNS.

Параметр “**-h <число>**” дает возможность задать порог, до которого будет расти TTL (по умолчанию - **30**).

Параметр “**-w <число>**” позволяет задать время ожидания каждого пакета (в миллисекундах), по умолчанию **1000** миллисекунд.

Утилита **pathping**

Утилита **pathping** фактически совмещает функциональность утилит **ping** и **tracert** и выполняется в две фазы: сначала, подобно **tracert**, собирается и выводится маршрут до заданного узла (только IP-адреса и имена) и затем, подобно **ping**, в течение некоторого времени (чем дольше выполнялась трассировка, тем больше будет это время) собирается статистика времен передачи пакетов, количеств и относительных долей потерянных пакетов для каждого из промежуточных маршрутизаторов (а не только для заданного узла, как **ping**).

Формальный синтаксис:

pathping [-g Список] [-h Число_прыжков] [-i Адрес] [-n] [-p Пауза] [-q Число_запросов] [-w Таймаут] [-P] [-R] [-T] [-4] [-6] узел

Наиболее полезен результат работы второй фазы утилиты **pathping** - он наглядно показывает, на каком из маршрутизаторов имеются проблемы с передачей пакетов. Для Windows существует мощный визуальный (показывает движение пакетов на карте Планеты) трассировщик VisualRoute фирмы VisualWare (<http://visualware.com>).

Утилита **arp**

Утилита **arp** дает возможность просматривать и изменять ARP-таблицу, в которой хранятся пары “**MAC-адрес - IP-адрес**” для тех узлов, с которыми в недавнем происходил обмен данными. Эта таблица формируется автоматически при работе сетевого узла, но администратор сети может вносить в нее записи вручную.

Формальный синтаксис:

arp -s inet_addr eth_addr [if_addr] или ARP -d inet_addr [if_addr] или ARP -a [inet_addr] [-N if_addr].

Здесь **if_addr** суть задает номер интерфейса.

Параметр “-a” позволяет вывести всю ARP-таблицу на экран.

Параметр “-a <IP-адрес>” запрашивает вывод записи об узле с заданным адресом на экран.

Параметр “-S <IP-адрес> <MAC-адрес>” позволяет добавить запись об узле с заданными адресами в ARP-таблицу.

Параметр “-d <IP-адрес>” служит для удаления записи об узле с заданным адресом из ARP-таблицы.

Параметр “-d *” очищает ARP-таблицу.

Утилита **hostname**

Утилита **hostname** всего-навсего выводит имя узла. Может быть использована в файлах сценариев для пакетной обработки.

Утилита **ipconfig**

Утилита **ipconfig** отображает и настраивает настройки протоколов TCP/IP. Без дополнительных параметров выводится IP-адрес, маска подсети и шлюз по умолчанию для всех сетевых интерфейсов. С параметром “/all” кроме сказанного, выводятся MAC-адреса сетевых интерфейсов, имя узла, адреса серверов DNS и WINS и некоторая другая информация.

Формальный синтаксис:

```
ipconfig [/? | /all | /release [адаптер] | /renew [адаптер] | /flushdns |  
displaydns /registerdns | /showclassid адаптер | /setclassid адаптер  
[устанавливаемый_код_класса_dhcp]].
```

Параметр “/flushdns” очищает кэш разрешенных имен DNS.

Параметр “/displaydns” выводит кэш разрешенных имен DNS на экран.

Параметр “/release [адаптер]” освобождает арендованный по DHCP (Dynamic Host Configuration Protocol) IP-адрес (если указан адаптер, то только для этого адаптера, иначе для всех адаптеров).

Параметр “/renew [адаптер]” запрашивает обновление аренды по DHCP IP-адреса (если указан адаптер, то только для этого адаптера, иначе для всех адаптеров).

Параметр “/registerdns” запрашивает обновление аренды по DHCP всех адресов и повторную их регистрацию в DNS.

Утилита **route**

Утилита **route** отображает таблицу маршрутов и позволяет ее изменять. Формальный синтаксис:

**route [-f] [-p] [команда [узел]] [MASK маска] [шлюз]
[METRIC метрика] [IF-интерфейс].**

При использовании **route** параметр “**метрика**” определяет качество данного маршрута (в **хопax** – количестве промежуточных маршрутизаторов, времени прохождения пакета по линиям связи, характеристикой надежности линии связи на данном маршруте и т.п.) в соответствии с заданным в сетевом пакете критерием (т.н. классом сервиса).

Команда “**PRINT**” выводит таблицу маршрутов: сетевой адрес; маска сети; адрес шлюза; интерфейс; метрика

Команда “**ADD**” позволяет добавить новый маршрут

Команда “**DELETE**” – удалить маршрут

Команда “**CHANGE**” – изменить (существующий) маршрут).

Утилита **netstat**

Утилита **netstat** отображает текущие соединения, порты, ожидающие соединения и статистические данные по протоколам TCP/IP. Без дополнительных параметров выводится список текущих соединений (протокол: TCP или UDP; локальный адрес и порт; внешний адрес и порт; состояние соединения).

Формальный синтаксис:

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

Параметр “**-a**” дополнительно отображает порты, ожидающие соединения; ожидающие TCP-порты обозначены состоянием “**LISTENING**”, а UDP-порты - внешним адресом “*:*”.

Параметр “**-n**” требует выводить все адреса и номера портов в числовом формате, поскольку по умолчанию **netstat** пытается разрешить IP-адреса и имена и заменить номер порта на его имя.

Параметр “**-r**” выводит таблицу маршрутов (сетевой адрес; маска сети; адрес шлюза; интерфейс; метрика). Подобную информацию

можно получить с помощью утилиты **route**.

Параметр “**-e**” позволяет получить статистику Ethernet.

Параметр “**-s**” выводит статистику по протоколам TCP, UDP и IP.

Параметр “**-e <протокол>**” применяется совместно с параметром “**-s**” для ограничения выдаваемой статистики заданным протоколом (TCP, UDP или IP).

Утилита **nbtstat**

Утилита **nbtstat** отображает статистику протокола и текущих подключений TCP/IP при использовании технологии NetBIOS через TCP/IP. Формальный синтаксис:

**nbtstat [-a Узел] [-A IP-адрес] [-c] [-n] [-r] [-R] [-RR] [-s]
[-S] [интервал]**

Параметр “**-c**” дает возможность просматривать содержимое кэша разрешенных имен.

Параметр “**-n**” позволяет просматривать список зарегистрированных данным компьютером локальных NetBIOS-имен.

Параметр “**-a <имя удаленного компьютера>**” выводит содержимое таблицы имен для удаленного компьютера, указанного по имени.

Параметр “**-A <IP-адрес удаленного компьютера>**” выводит содержимое таблицы имен для удаленного компьютера, указанного по IP-адресу.

Параметр “**-r**” дает возможность просматривать статистические данные о разрешении имен.

Параметр “**-интервал**” задает циклический вывод статистики через указанный интервал в секундах (прекращение вывода – **Ctrl+C**).

2.2 Необходимое оборудование.

IBM PC-совместимая ЭВМ, подключенная к компьютерной сети с помощью Ethernet-сетевой карты.

2.3 Порядок проведения работы.

Первой частью работы является назначение IP-имен входящим в

локальную сеть компьютерам и настройка стека протоколов TCP/IP (для локальной сети IP-адрес указывается статическим, адреса DNS-серверов задаются преподавателем).

Второй этап работы – проверка работоспособности сетевого соединения с использованием стандартных утилит Windows. При этом следует проверить корректность задания хотя бы одного адреса DNS-сервера (рекомендуется, например, **192.190.241.65**) при настройке сетевого ПО, иначе сопоставление IP-адреса заданному доменному имени окажется невозможным.

2.4 Типовые задания.

- Просмотр и анализ настроек стека протоколов TCP/IP с использованием утилиты **ipconfig**.
- Тестирование сетевого соединения утилитой **ping** (конечный адрес указывается преподавателем – им может быть, например, IP-адрес местного Proxy-сервера, адрес одной из машин в сети, доменное имя некоторого узла в Сети и др.).
- Прослеживание маршрута пакетов до заданного узла утилитой **tracert**.
- Отображение и изменение таблицы хранения соответствия MAC- и IP-адресов с помощью утилиты **arp**.
- Просмотр текущих сетевых соединений, портов, TCP/IP-статистики утилитой **netstat**.
- Отображение статистики протокола и текущих подключений TCP/IP при использовании NetBIOS через TCP/IP утилитой **nbtstat**.
- Отображение и изменение таблицы маршрутов утилитой **route** (преподавателем может быть задано изменение или дополнение маршрутов).

2.5 Оформление отчета по работе.

В отчете указываются параметры используемой сети (тип ПЭВМ, MAC- и IP-адреса сетевой карты, тип и число портов коммутатора, топология локальной сети).

При использовании утилит **ping**, **tracert** или **pathping** необходимо отметить время передачи пакетов (обычно среднее), число и долю по-

терянных пакетов (по ней сделать вывод о корректности работы каждого промежуточного маршрутизатора).

При использовании утилиты **netstat** необходимо зафиксировать назначение TCP- и UDP-портов имени ПЭВМ, локальному адресу и их состояние (параметр “-a”). Общую статистику обменов можно получить при использовании параметра “-e”, с использованием параметра “-s” следует зафиксировать и проанализировать статистику по всем протоколам стека TCP/IP (включая ICMP, UDP).

При использовании утилиты **nbtstat** первым шагом рекомендуется просмотреть таблицы NetBIOS-имен на локальном компьютере (параметр “-n”) и далее на других компьютерах в сети (параметры “-a” или “-A”).

При использовании утилиты **route** (с параметром **PRINT** для вывода на экран) просматривается как список интерфейсов, так и список активных маршрутов. Удаление маршрута достигается вводом **route DELETE** узел (где узел -IP-адрес удаляемого из маршрута узла), добавление - **route ADD** узел **MASK** маска шлюз **METRIC** метрика IP интерфейс (где узел – IP-адрес добавляемого узла, маска – значение маски, шлюз – IP-адрес шлюза, метрика – значение метрики добавляемого маршрута, интерфейс – номер сетевого интерфейса; может быть опущен, тогда выбирается наиболее подходящий для указанного шлюза).

2.6 Вопросы к защите работы.

- Из каких соображений назначаются IP-адреса сетевым платам, если для входящих в локальную сеть компьютеров не предполагается доступ к InterNet? Если работа в InterNet предполагается?
- Какие параметры сетевой платы настраиваются при ее конфигурировании?
- Какие этапы предполагает настройка стека протоколов TCP/IP для Windows'NT? С какой целью применяется маска подсети? Что такое основной шлюз?
- Какой формат имени сетевого ресурса используется при обращении к нему?
- Каким путем утилиты **ping**, **tracert** и **pathping** осуществляют прослеживание маршрутов пакетов к заданному узлу?
- В каких случаях и с какой целью используется утилита **route**?

Что такое **метрика** в списке параметров этой утилиты?
ЛАБОРАТОРНАЯ РАБОТА № 3.

ПРОЕКТИРОВАНИЕ ЛОКАЛЬНОЙ КОРПОРАТИВНОЙ КОМПЬЮТЕРНОЙ СЕТИ

Цель работы – приобретение практических знаний и навыков в проектировании локальных компьютерных сетей с помощью системы автоматизированного проектирования NetWizard.

3.1 Теоретические основы.

InterNet-pecypc <http://NetWizard.ru> (*‘мастер построения сетей’*) позволяет на основе простого интерактивного взаимодействия пользователя и программы расчета на сервере (пользователь просто отвечает на вопросы, определяя тем самым исходные для проектирования параметры) спроектировать локальную компьютерную сеть вплоть до масштаба здания (*кампусная ЛВС*).

NetWizard помогает оценить стоимость сети с необходимыми параметрами. При этом удобно оптимизировать стоимость сети путем изменения требований к ней. Такие ситуации возникают в следующих случаях:

- когда организация переезжает в новое здание,
- если открывается новый филиал организации,
- планируется модернизация сети,
- когда создается новая фирма и так далее.

В течение диалога можно неоднократно уточнять задачу. Предусмотрено несколько квалификационных уровней:

- для пользователей с опытом работы простого оператора ПЭВМ (уровень *“особенности сети”*),
- для пользователей, обладающих представлениями о требованиях к ЛВС (*“требования к сети”*),
- для пользователей, знакомых с техническими параметрами оборудования (*“технические параметры”*),
- для пользователей, знакомых с ассортиментом продукции (*“варианты оборудования”*).

В системе NetWizard имеется несколько типовых заданий (проектов), пользователь может дополнить этот список. Типовыми заданиями являются:

- Спроектировать активное оборудование локальной вычислительной сети (версия 3.2).
- Спроектировать пассивное сетевое оборудование (версия 2.03).
- Проект ЛВС в одном здании (версия 2.03)

По умолчанию предлагается оборудование фирмы 3Com (<http://www.3com.com>), однако возможен выбор оборудования и иных фирм. Параметры системы устанавливаются на основе экспертных оценок с учетом имеющихся ограничений.

Результаты расчетов выдаются в нескольких формах (отчетах):

- Схема размещения оборудования ЛВС (выдается как схема размещения оборудования в типовых приборных шкафах, так и основные параметры размещенного оборудования).
- Спецификация спроектированной ЛВС. Спецификация выдается в виде таблицы с указанием артикула и наименования товара, количества и цены отдельно для активного (сетевые концентраторы - *switchs*) и пассивного оборудования (сетевые кабели, розетки и т.п.), иного (серверы и рабочие станции, программное обеспечение, корпуса, источники бесперебойного питания и т.п.) оборудования стоимость работ по монтажу сети; подсчитываются итоговые расходы на создание ЛВС.
- Структурная схема ЛВС выдается в виде графической схемы с конкретизацией основных технических параметров каждого узла (серверов коммуникационных центров, пользовательских ПЭВМ).
- Техническое задание на создание ЛВС включает общие требования к создаваемой ЛВС, описание поставленной задачи, параметры технического решения.

После получения результатов расчетов имеется возможность вернуться ко вводу исходных данных, пересмотреть их и произвести новый расчет; результаты всех расчетов сохраняются в проекте с заданным наименованием.

При работе не следует использовать кнопку броузера **“Назад”**. Чтобы вернуться назад, имеются специальные экраны диалога, которые предложат вернуться к началу текущего раздела.

Например, для проектирования ЛВС здания (общая схема для случая двух этажей приведена на рис.3.1) порядок диалога включает следующие пункты:

- Указать количество этажей здания, в котором будет расположена проектируемая сеть (диапазон **1÷20**).
- Задать общее количество ПЭВМ в проектируемой сети

(1÷1000).

- Определить список задач, решаемых проектируемой сетью (сетевое хранение файлов и сетевая печать, поддержка электронной почты, коллективная работа с базами данных, публикация документов в InterNet или IntraNet, резервное копирование файлов сервера, резервное копирование файлов сетевых приложений и др.).

- Количество коммуникационных центров (КЦ, обычно не менее чем по одному на каждом этаже, но не более 30).

Коммуникационный центр включает оборудование, к которому подключаются группы компьютеров (в простейшем случае КЦ суть концентратор, *switch*). Корректный выбор числа КЦ является залогом правильности всего процесса проектирования ЛВС. На этом этапе обязательно следует ознакомиться с содержимым экрана контекстной помощи, где даются рекомендации по выбору числа КЦ.

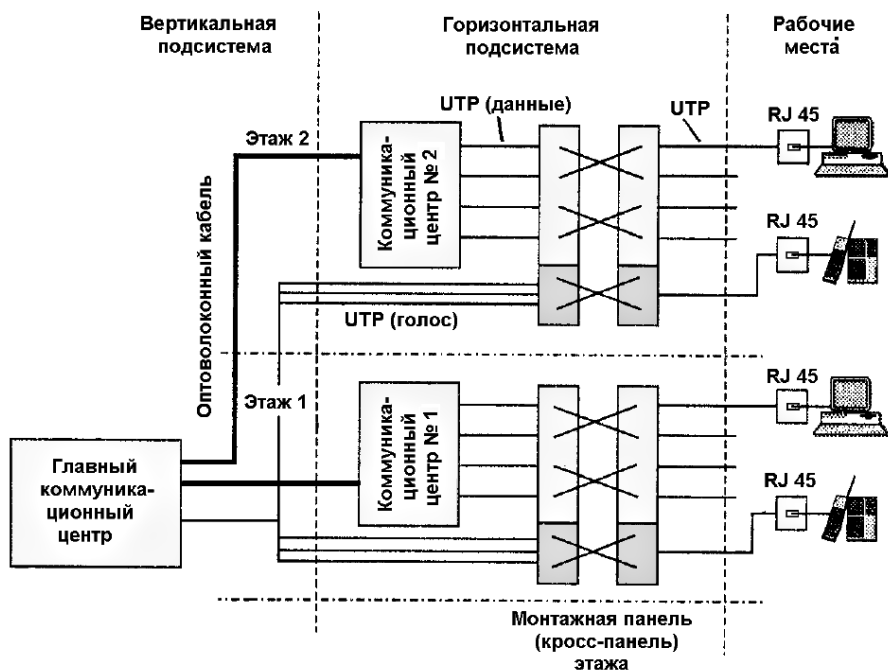


Рис.3.1 Общая схема расположения сетевого оборудования в 2-х этажном здании

- По желанию можно принудительно задать скорость передачи данных, возможность управления устройствами и трафиком, отказоустойчивость, параметры каналов связи.
- Указать места использования оптоволокну (между КЦ, в связях с серверами, рабочими станциями и/или для связи с ЛВС других зданий).
- Указать на необходимость использования экранирования медных кабельных систем (с целью снижения уровня излучения кабелями и уменьшения влияния на сеть помех от внешних источников электромагнитного поля).
- Количество портов кабельной системы для каждого рабочего места (в случае использования на рабочем месте одновременно ПЭВМ и телефона следует указать 2).
- Значения основных параметров главного КЦ (средние расстояния между главным и этажными КЦ, между КЦ и рабочими местами, способ монтажа кабельной системы и др.).
- Выбрать операционную систему для серверов и язык операционной системы, офисные пакеты и иное необходимое программное обеспечение для рабочих станций.
- Указать способ организации системы бесперебойного питания и ее характеристики.

3.2 Необходимое оборудование

ЭВМ любого типа, подключенная к сети InterNet и поддерживающая WEB-браузер.

3.3 Порядок проведения работы.

Студент получает задание на проектирование сети (например, при проектировании ЛВС уровня здания преподаватель задает число этажей и число ПЭВМ в здании, список выполняемых сетью задач; при этом число КЦ и иные исходные параметры определяются студентом самостоятельно и согласуются с преподавателем.

3.4 Типовые задания.

- Произвести расчет сети с целью минимизации стоимости сети
- После решения произвести корректировку исходных данных и

повторить расчет.

3.5 Оформление отчета по работе.

В отчете указываются заданные преподавателем и определенные студентом исходные данные для расчета ЛВС, результаты расчетов (удобно использовать распечатки экранных форм соответствующих отчетов). В случае оптимизации по заданному параметру необходимо привести данные всех итераций по выбранному параметру.

3.6 Вопросы к защите работы.

- Что такое коммуникационный центр и каковы его функции? Какие сетевые устройства могут являться коммутационными центрами (привести примеры)?
- Каково максимальное расстояние между узлами сети при использовании медного кабеля? То же при применении оптоволокна?
- Что такое вертикальная и горизонтальная подсистемы ЛВС? Трафик какой из них выше?
- В каких случаях необходимо применять экранированные сетевые кабели?
- В каких случаях необходимо использовать несколько коммутационных центров на каждом этаже?

ЛАБОРАТОРНАЯ РАБОТА № 4.

ПОСТРОЕНИЕ И ИССЛЕДОВАНИЕ КОМПЬЮТЕРНЫХ СЕТЕЙ С ПОМОЩЬЮ СИСТЕМЫ NETCRACKER

Цель работы – приобретение практических знаний и навыков в проектировании компьютерных сетей с помощью системы NetCracker.

4.1 Теоретические основы.


Компьютерная система NetCracker Technology (фирма *Division of Advanced Visual Data, Inc.*, <http://www.netcracker.com>) является мультимедиа-инструментом (использует анимацию при визуализации движения пакетов и нагрузки сети) для моделирования и анализа эффективности сети, позволяет проверять связность сети, возможность и эффективность использования в каждом конкретном случае сетевых устройств (как стандартных, так и заказных), включает обширную библиотеку готовых сетевых устройств и возможность определять новые типы устройств.

Главное окно NetCracker (помимо заголовка, главного меню и панелей инструментов) состоит из трех фреймов: браузера (*Browser*) слева, рабочей зоны (*Workspace*) справа и панели изображений (*Image*) снизу. В главном окне NetCracker располагается шесть блоков (*Toolbars*) меню, оформленных в виде кнопок, дублирующих выпадающие варианты главного меню. Панель рисования активизируется кнопкой:



или выбором **View → Bars → Drawing** в главном меню (рис.4.1).

Для ознакомления с работой программы NetCracker удобно пользоваться имеющимся проектом **TUTOR.NET** (расположен в каталоге `./NetCracker /Samples`). Проект открывается (рис.4.2) последовательным выбором вариантов главного меню **File → Open** или щелчком по соответствующей кнопке в блоке расположенных в верхней части главного окна кнопок. В окне рабочей зоны (изменение масштаба

осуществляется **View → Zoom**) появляется схема сети, причем в окне Top располагается укрупненная схема проекта (включающая два здания Building и Building(2) и схематизированное в виде облака изображение канала связи технологии ATM - *Asynchronous Transfer Mode*), в окне Building - конкретизация схемы в здании Building (указанием иерархической схемы служит обведение красной полоской объекта Building в окне Top и символ  в окне Building). Подобный проект называется *многоуровневым*.

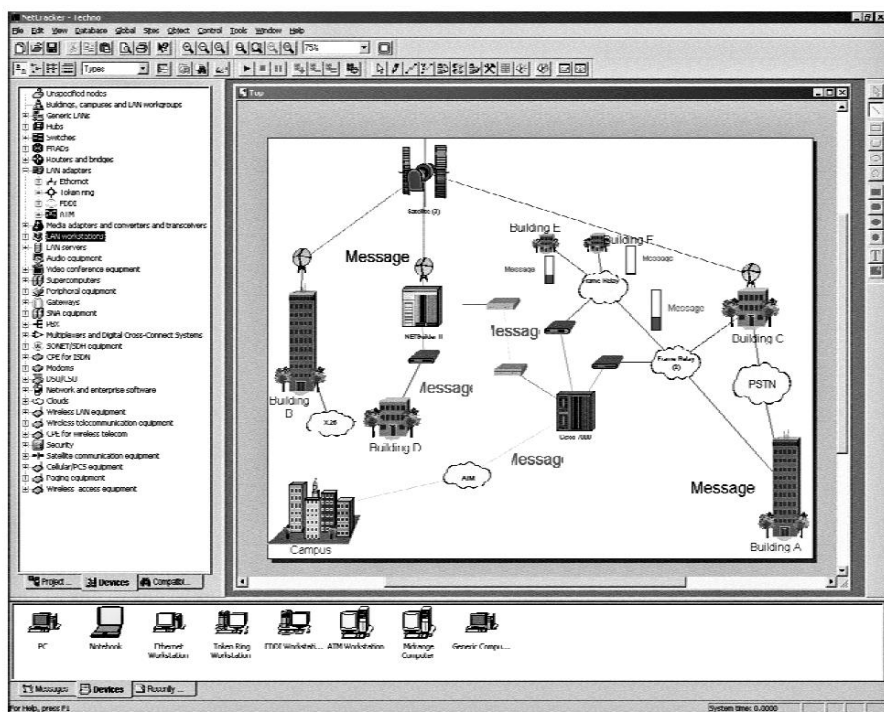







Рис.4.1 Главное окно программы NetCracker
(загружен проект Techno)

Переименование окна сайта достигается выбором **Sites → Sites Setup**, переименование объекта в рабочей зоне (например, здания Building) реализуется выбором варианта Properties из меню, выпадающего при щелчке правой кнопки мыши на выбранном объекте

(устройстве). Рисовать, вводить текст или изображения в рабочую зону можно, используя меню **Draw** (активируется нажатием кнопки  или выбором **View → Bars → Drawing** в главном меню), возврат из режима **Draw** осуществляется щелчком по кнопке  (переход в стандартный режим).

Запуск анимации достигается выбором **Controls Start** или **F5**, остановка - **Control → Stop** или **Alt+F5**, пауза - **Controls Pause** или **Space**; кнопки  соответственно. Кнопки  позволяют увеличить, уменьшить или установить по умолчанию скорость анимации, кнопка  - вручную настроить скорость анимации, интенсивность движения пакетов **Packet speed** и размер пакетов **Packet size** (действия также доступны посредством варианта **Control** главного меню).

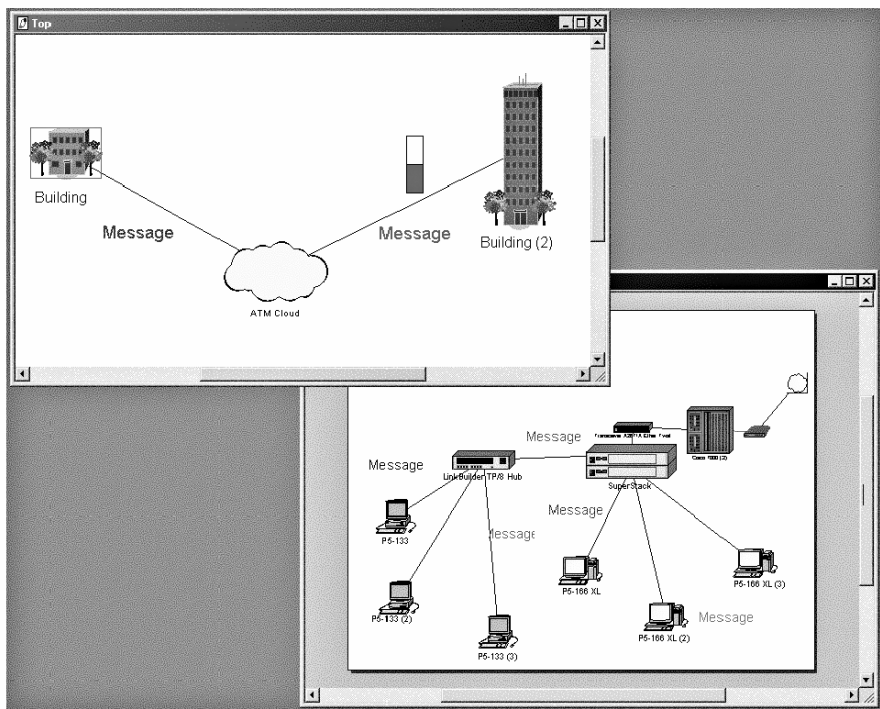


Рис.4.2 Рабочая зона NetCracker
с загруженным проектом TUTOR.NET

Устройства для добавления в рабочую зону выбираются из размещенной в окне браузера иерархической базы данных, причем возможна сортировка по типам, производителям или определенным пользователям устройствам (варианты **Types**, **Vendors** и **User** в выпадающем списке блока меню:



Первые 4 кнопки служат для выбора режима показа изображений устройств в панели изображений – **Крупные/Мелкие значки/Список/Таблица**).



При выборе закладки **Recently User** (*Используемые*) в левой нижней части панели изображений на последней показываются все устройства, используемые в данном проекте.

Например, добавления в рабочую зону конкретного маршрутизатора фирмы Cisco Systems следует (сортировка по типам устройств, нажата кнопка Крупные значки) в браузере выбрать группу **“Routers and bridges”** (*Маршрутизаторы и мосты*), щелчком мыши по знаку “плюс” раскрыть ее и в подгруппе **“Backbone”** (*Магистральные*) выбрать **“Cisco Systems”** и далее подгруппу **“Cisco 7010”**; в панели появится сам маршрутизатор **Cisco 7010** и модули расширения для него.

Для выбора сетевых адаптеров фирмы 3Com для сети Ethernet следует открыть группу **“LAN adapters”**, подгруппу **“Ethernet”** и далее подгруппу **“3COM Corp.”**; в панели изображений устройств появятся изображения сетевых адаптеров (см. рис.4.1).

Для добавления выбранного устройства в рабочую зону достаточно отбуксировать (с помощью мыши) его изображение из панели изображений в рабочую зону.

Для получения информации об имеющемся в рабочей зоне конкретном устройстве следует выбрать вариант **Configuration** из меню, выпадающего при щелчке правой клавиши мыши при нахождении курсора мыши в области изображения устройства. Для маршрутизатора Cisco 7000, например, откроется диалог конфигурации Cisco 7000 (рис.4.3). Диалоги конфигурации содержат изображение устройства, панель выбора конфигурации (справа), кнопки Device Setup (*Настройка устройства*), Plug-in Setup (*Настройка дополнительного*

модуля), Close (Заккрыть) и Help (Помощь). Диалоговое окно рис.4.2 позволять в графическом режиме добавлять (путем ‘буксировки’ мышью) модули в ‘корзину’ устройства. Например, на рис.4.3 в ‘корзину’ устройства Cisco 7000 добавлены 4 модуля HSSI Interface Processor (процессор высокоскоростного последовательного интерфейса). В процессе буксировки курсор имеет форму  в тех местах, куда нельзя “положить” модуль, и форму  в тех местах, куда модуль поместить можно. При щелчке мышью по любому находящемуся в списке модулю становится доступной кнопка **Plug-in Setup**, что говорит о возможности настройки этого модуля.

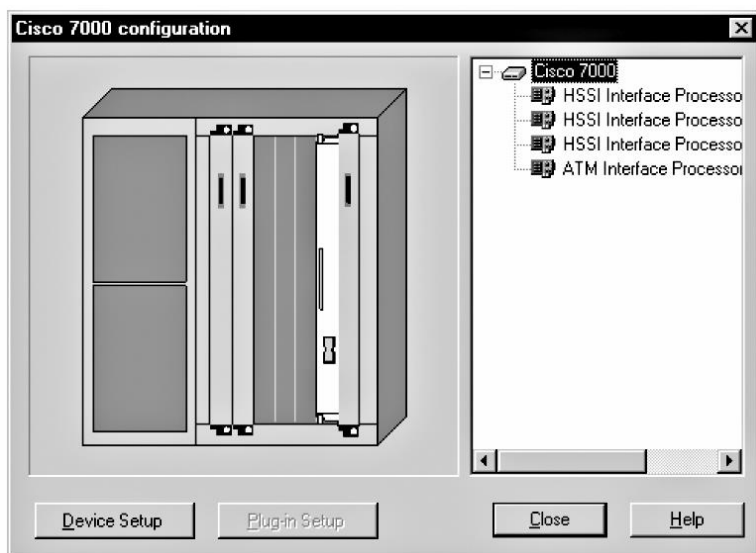


Рис.4.3 Диалоговое окно конфигурации устройства Cisco 7000

Диалоговое окно настройки модуля приведено на рис.4.4 (вызывается нажатием кнопки **Plug-in Setup** в окне конфигурирования, рис.4.3) или посредством пункта **Properties** локального меню, вызываемого правой кнопкой мыши для модуля из списка в правой части рис.4.3.

Кнопка **Device Factory** (доступ из главного меню посредством **Databse → Device Factory** или **Ctrl+Shift+F**) позволяет самостоятельно создать новые (заказные) сетевые устройства с заданными параметра-

ми (при этом возможно их создание на основе существующих, но с измененными параметрами).

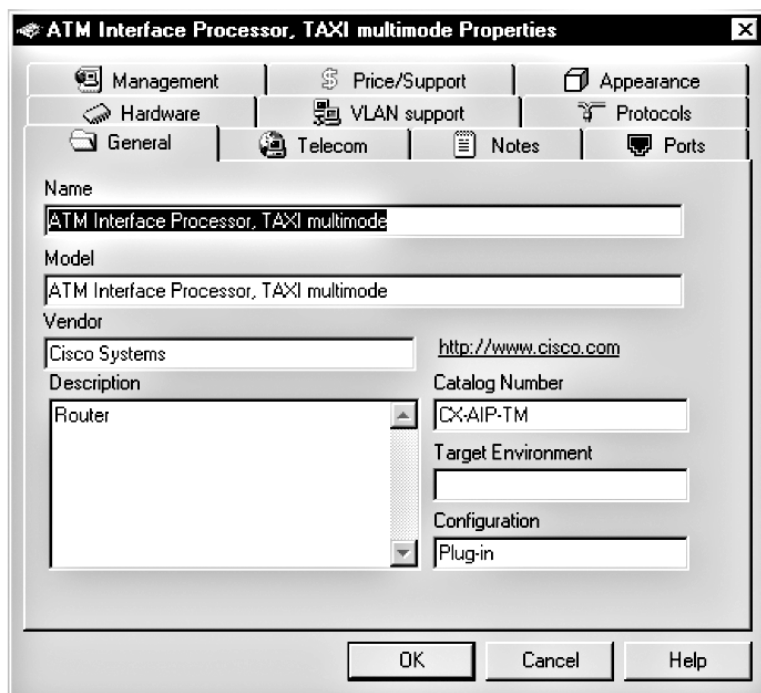



Рис.4.4 Диалоговое окно настройки модуля

В базе данных существует набор готовых (уже сконфигурированных) обобщенных устройств (*generic devices*). Например, в дереве устройств расположена группа **Switches**, в ней – подгруппа **Workgroup**, в ней – **Ethernet**, в ней – **Generic Devices** (выбранное та-

ким способом обобщенное устройство изображается как



Ethernet Switch

Установление связи между устройствами достигается щелчком по кнопке **Link Devices**  (вариант **Sites** → **Modest Link**) с последующим щелчком мышью последовательно по двум устройствам, подлежащими связыванию; в результате открывается диалоговое окно

Link Assistant (рис.4.5):

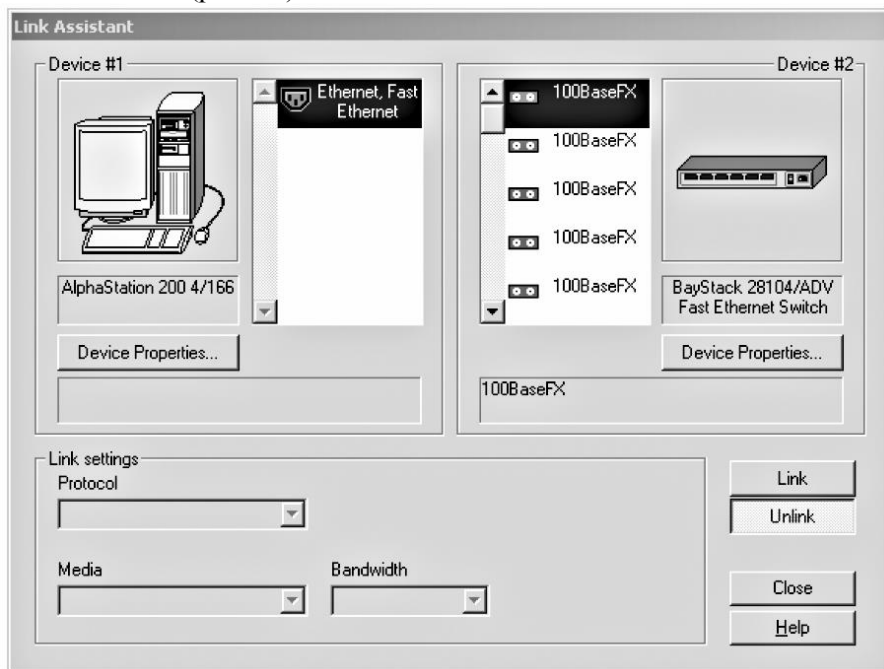



Рис.4.5 Диалоговое окно установления связи между устройствами

Справа и слева в окне **Link Assistant** показаны устройства, связь между которыми должна быть установлена (кнопка **Link**, кнопка **Unlink** разрывает связь); причем в случае наличия у устройства нескольких портов (как показано на рисунке справа для коммутатора Fast Ethernet) имеется возможность выбрать свободный. Кнопка **Device Properties** позволяет уточнить параметры узлов, группа **Link Setting** - параметры линии связи (протокол, тип линии связи и ее длину, скорость передачи данных). Цвет линий связи указывает на тип связи (коаксиал, витая пара, оптика и т.д.) и может быть изменен посредством диалога **View** → **Media Colors**. Возможен вариант быстрой установки связи - при нажатой клавише **Shift** на клавиатуре щелкнуть мышью последовательно по двум устройствам и отпустить **Shift**. Вызов окна **Link Assistant** возможен и после установления связи (например, для изменения параметров связи) через вариант **Link Assistant** выпадаю-

щего при нажатии правой кнопки мыши меню (курсор мыши при этом должен указывать на интересующую связь). Разорвать/восстановить связь (например, с целью анализа влияния данной коммуникационной ветви на общую картину функционирования сети) можно кнопкой

Break/Restore  (**Sites → Modes → Break/Restore** или **Ctrl+8**),

при этом разрыв связи показывается символом 

Для назначения потоков данных между сетевыми устройствами

активируется кнопка **Set Traffic**  (то же самое через главное меню **Sites → Modes → Set Traffic** или **Ctrl+5**) с последовательными щелчками по связанным устройствам, назначение потоков происходит в окне **Profiles** (рис.4.6).

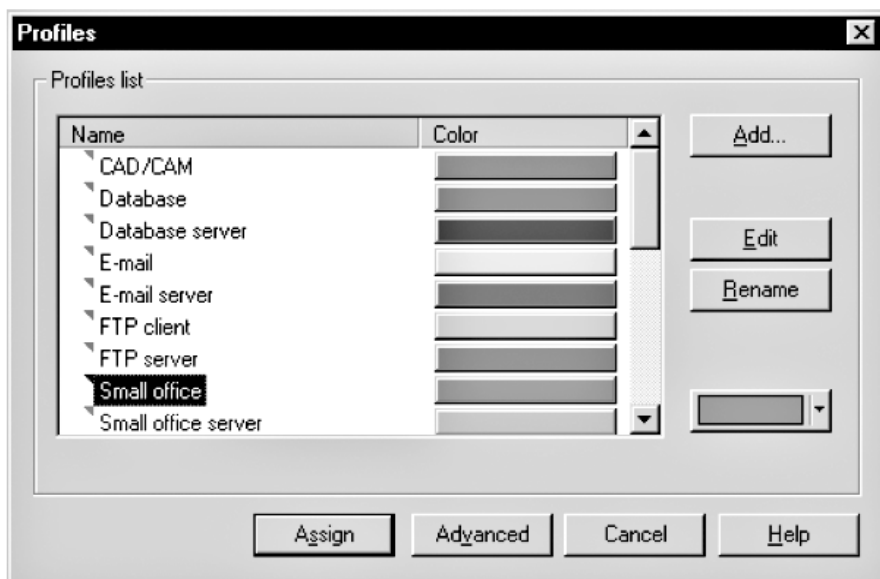


Рис.4.6 Окно Profiles назначения потоков данных.

Для назначения потока данных в обратном направлении следует щелкнуть по устройствам в обратном порядке, имеется возможность сменить цвет изображаемых при анимации пакетов. Работа с окном

Profiles завершается нажатием кнопки **Assign**.

Для указания вида статистической информации о потоках данных следует курсором мыши выбрать связь и из выпадающего по нажатию правой кнопки мыши меню выбрать вариант **Statistics**. Выбор вида указателя статистики осуществляется постановкой “галочек” в нужных столбцах открывающегося диалогового окна **Statistical Items** (вид указателя статистики указан в верхней части окна – вертикальный указатель, числовое значение в процентах и график), в левой части окна находятся указатели типа статистики (средняя и текущая нагрузки сети, текущее и усредненное использование, текущее число пакетов).

Изменить параметры выводимого в рабочей зоне текста можно путем выбора варианта **Properties** из меню, выпадающего по щелчку правой кнопки мыши при выборе (что подтверждается показом границ текста черными квадратами) курсором мыши самого текста.

Для проверки и установки протоколов маршрутизации служит закладка **Protocols** диалогового окна **Model Setting** (вызывается щелчком правой кнопки мыши в любом пустом месте в пределах рабочей зоны и выбором варианта **Model** → **Setting** в выпадающем меню или при использовании главного меню **Global Model Setting** или **Ctrl+G**), см. рис.4.7.

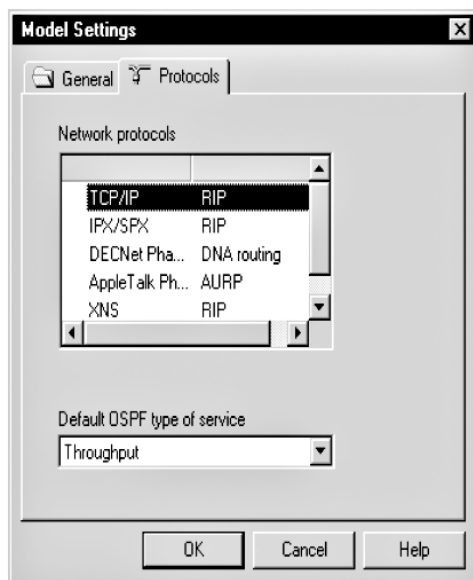


Рис.4.7 Закладка Protocols диалогового окна Model Setting

Как видно, в данном случае для маршрутизируемого протокола TCP/IP выбран метод маршрутизации RIP (***R**outing **I**nformation **P**roto-**col***), выбора другого следует щелкнуть по слову “RIP” и из выпадающего списка выбрать желаемый (для TCP/IP доступны два варианта RIP, OSPF и IGRP).

Для получения информации о любом пакете следует навести на него курсор мыши (в режиме **Pause** при анимации) и выбрать из выпадающего при щелчке правой кнопкой мыши меню вариант **Properties** (вид открывающегося окна Packet Properties приведен на рис.4.8).

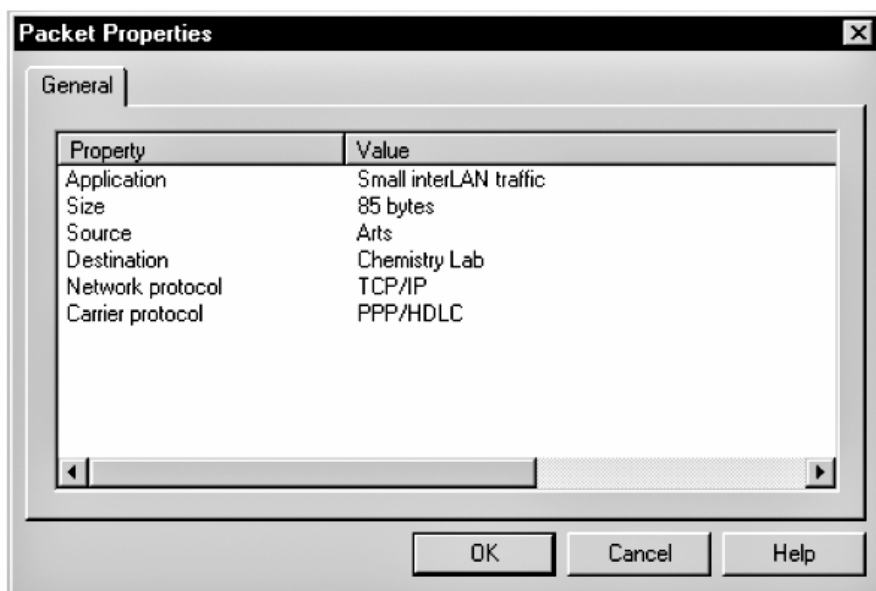


Рис.4.8 Диалоговое окно Packet Properties

В диалоговом окне приведена информация о сгенерировавшем пакет приложении, размере пакета, его источнике и месте назначения, сетевом и канальном протоколах.

Для получения твердой копии в NetCracker существуют средства генерации отчетов (варианты главного меню **Tools** → **Reports**). Например, вариант отчета в **Routers/Bridges** (*Маршрутизаторы и Мосты*) позволяет в табличной форме распечатать информацию об ис-

пользуемых в проекте маршрутизаторах и мостах, вариант **Bill Of Materials** (*Смета*) содержит полный список оборудования и цен на него (включая суммарные расходы на оборудование). Получить копию рабочей зоны (в любом графическом формате) можно используя главное меню как **Tools → Graphic Export**.

4.2 Необходимое оборудование

- IBM PC-совместимая ЭВМ
- предустановленная ОС Windows
- система NetCracker.

4.3 Порядок проведения работы.

Первым этапом работы является ознакомление с системой **Net-Cracker** и получение начальных навыков работы с ней (добавление устройств и установление связей между ними при создании проекта, настройка протоколов, определение интенсивности трафика, анализ загруженности сети и т.п.); при этом удобно использовать имеющийся проект TUTOR.NET, специально предназначенный для целей обучения.

На втором этапе студент получает индивидуальное задание для создания проекта и анализа работы сети.

4.4 Типовые задания.

- Составить модель компьютерной сети вычислительной лаборатории кафедры, проанализировать потоки данных, определить слабые места (работающие на пределе возможностей устройства и/или линии связи) сети.
- Составить модель компьютерной сети по заданию преподавателя (исходные данные – число подразделений эксплуатирующей сеть организации, число ПЭВМ в каждой комнате и усредненный трафик каждой, расстояния между комнатами, желательность дальнейшего расширения и т.д.).
- Проанализировать потоки данных, выявить слабые места, уровень живучести сети (например, наличие обходных путей при выходе из строя отдельных связей при использовании маршрутизации).

4.5 Оформление отчета по работе.

В отчете указываются исходные данные для проектирования сети, описывается выбранная стратегия создания сети, параметры сетевых устройств и линий связи, приводится схема (печатная копия рабочей зоны главного окна системы NetCracker) сети и один из стандартных видов выходной информации (обычно Bill Of Materials).

4.6 Вопросы к защите работы

- Какие задачи проектирования и исследования сетей могут быть решены с использованием пакета NetCracker?
- Для каких целей служит броузер устройств? Рабочая зона? Панель изображений?
- Что такое многоуровневый проект?
- Какие средства NetCracker позволяют количественно судить о степени загруженности конкретного канала связи?
- Каким образом в системе NetCracker можно решать задачи типа ‘а что, если...’?

ЛАБОРАТОРНАЯ РАБОТА 5

ИЗУЧЕНИЕ АЛГОРИТМА МАРШРУТИЗАЦИИ OSPF (*OPEN SHORTEST PATH FIRST*)

Цель работы – практическое изучение функционирования одного из наиболее мощных алгоритмов маршрутизации OSPF (*Open Shortest Path First*)

5.1 Теоретические основы.

Конечная цель маршрутизации состоит в доставке пакетов в пункт назначения при максимальной эффективности. Количественно эффективность может быть выражена, например, средневзвешенной суммой времен доставки сообщений (при нижнеограниченной вероятности доставки). Выбор одного из возможных в маршрутизаторе направлений зависит от топологии сети (а она обычно динамична вследствие, напр., неизбежного временного выхода из строя некоторых узлов), длин очередей в узлах коммутации, интенсивности потоков данных и т.п.

Алгоритм маршрутизации неизбежно должен включать процедуры:

- Измерение и (количественное) оценивание параметров сети (включая рассылку и прием служебной информации для определения текущих параметров сети – если, конечно, они не императивно заданы *администратором сети*).
- Расчет таблиц маршрутизации (ТМ, фактически путей передачи данных с порта на порт для конкретного маршрутизатора).
- Реализация принятых маршрутных решений (включая рассылку служебной информации для управления маршрутизаторами).

Если при расчете ТМ используется информация только о состоянии данного узла, то алгоритм маршрутизации является *изолированным*; при использовании данных о всей сети – *глобальным*. Если ТМ (динамически) отслеживает состояние сети – алгоритм является *адаптивным*, иначе – *фиксированным (статическим)*, а при редких изменениях в ТМ – *квазистатическим* (при статическом маршрутизировании изменения в ТМ вносит администратор сети).

Невозможно представить более простого алгоритма маршрутиза-

ции, чем обладающий свойством изолированности и статичности. При использовании анализа длины очереди сообщений в каждом маршрутизаторе алгоритм называется *алгоритмом кратчайшей очереди* (пакеты посылаются по направлению, в котором для данного узла длина очереди минимальна). *Лавинный алгоритм* является многопутевым и основан на рассылке копий пакета по всем возможным направлениям (если в данном направлении данный пакет уже проходил, вторичное прохождение не допускается – пакет уничтожается).

Широко используются протоколы маршрутизации RIP (*Routing Information Protocol*, RFC 1058, RFC 1388) и OSPF (*Open Shortest Path First*, RFC 1247).

RIP использует *метод рельефов*, основан на *алгоритме Беллмана-Форда* и применяется в основном на нижних уровнях иерархии сети.

OSPF применяет итерационный *алгоритм Дейкстры* для поиска кратчайшего пути в графе и более эффективен при использовании в больших сетях (однако в каждом маршрутизаторе должна иметься информация о состоянии всей сети).

Компьютерная сеть моделируется графом, причем его вершины соответствуют маршрутизаторам, а ребра – каналам связи; вес ребер это оценка *метрики* (например, расстояния; причем метрика неотрицательна) между инцидентными – принадлежащими заданному ребру – узлами (рис.5.1). Такой граф является *взвешенным* (т.к. каждому ребру поставлено в соответствие некоторое число), *неориентированным* (направленность ребер не существенна), не имеет *петель* (ребер, соединяющих вершину саму с собой) и *связным* (из любой вершины можно найти путь в любую другую).

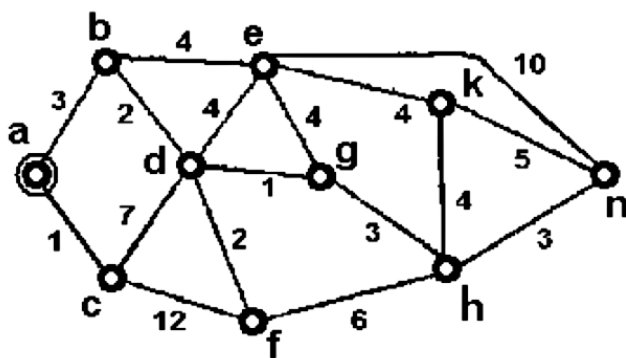


Рис.5.1 Пример графа сети для маршрутизации.

Алгоритм Дейкстры поиска кратчайшего пути между двумя заданными вершинами графа (*) является типичным примером *поиска в ширину* в графе (“жадный” алгоритм) и фактически основан на выделении и анализе подграфов в исходном графе. На рис.5.2 приведена схема алгоритма Дейкстры применительно к первому шагу поиска кратчайшего пути в графе рис.5.1.

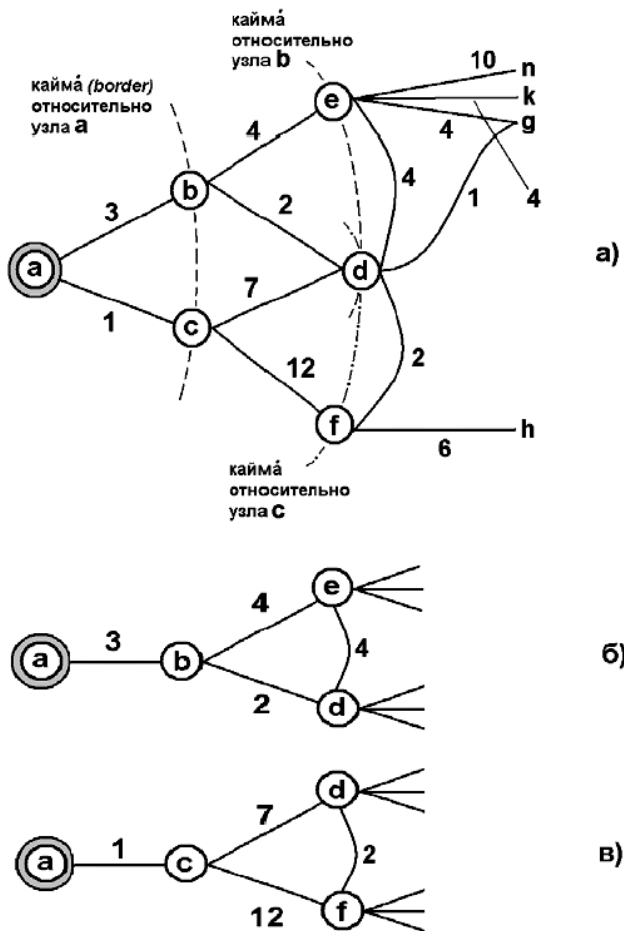


Рис.5.2 К описанию алгоритма Дейкстры нахождения минимального пути в графе.

Рассмотрим работу алгоритма для случая поиска кратчайшего пути между вершинами a и n . Через $R(i, j, k \dots)$ будем обозначать длину пути по вершинам $i, j, k \dots$. Рассматриваемый алгоритм описан подробно в применении к первому шагу поиска кратчайшего пути в графе рис.5.1 (в кайму относительно заданной вершины входят все вершины, отстоящие от нее на одно ребро):

1. Занести в список вершин кратчайшего пути начальную точку a .
2. Проанализировать число ребер, ведущих от начальной точки (в данном случае их два, см. рис.5.2а), в соответствии с этим разбить исходный граф на 2 подграфа (рис.5.2б и 5.2в).

3. Для каждого подграфа анализировать вершины, составляющие кайму (*border*) относительно a (это вершины $[b, c]$); при этом из рассмотрения исключаются вершины, уже входящие в список вершин кратчайшего пути (путь “назад” бесперспективен). Для первого подграфа (рис.5.2б) кайма относительно b есть $[e, d]$, для второго (рис.5.2в) - $[d, f]$ (вершина a входит в обе каймы, но по указанным причинам исключается из рассмотрения). Имеем:

$$R(a, b, e) = 7$$

$$R(a, b, d) = 5$$

$$R(a, c, d) = 8$$

$$R(a, c, f) = 13$$

Заметим, что к вершине d от a ведут два ребра (через b и c)!

4. Определить минимум из этих расстояний – наименьшее есть $R(a, b, d)=5$, что и позволяет принять в качестве частичного минимального пути $[a, b, d]$. В случае двух или более одинаковых значений можно взять любое (обычно берется первое).

5. К списку вершин кратчайшего пути добавить только что определенные вершины b, d .

6. Повторять пункты 2÷5 алгоритма, принимая в качестве исходной последнюю из списка вершин частичного кратчайшего пути (в нашем случае это вершина d) до тех пор, пока не будут проанализированы все входящие в вершину n ребра (для графа на рис.5.1 их 3).

Таким образом, при каждой итерации продвигаемся к конечной вершине на две вершины (на два ребра), каждый раз выбирая минимум пути на протяжении смежных ребер. После получения списка вершин кратчайшего пути его длина определяется примитивным суммирова-

нием метрик ребер.

Для графа рис.5.1 имеем:

$$R^{min}(a, b, d, g, h, n) = 12.$$

Пользовательский интерфейс программы OSPF приведен на рис.5.3. В окне ввода описания графа каждое ребро задается строкой, где первые два слова – имена определяющих ребро вершин (длина имени не более **10** символов), далее - метрика ребра (целое число); данные разделяются пробелами. Т.к. все ребра являются ненаправленными, порядок указания определяющих их -вершин несущественен. Главное меню является стандартным, вычисления начинаются выбором варианта “**Calculate**”, проект и результаты расчета могут быть сохранены в файл для последующего использования, для получения дополнительной информации рекомендуется использовать вариант “**Help**”. В применении к компьютерным сетям аналогом понятия “вершина” является “узел” (“сетевой узел”), “ребро” – “путь” (“меж-узловой путь”, “линия связи”).

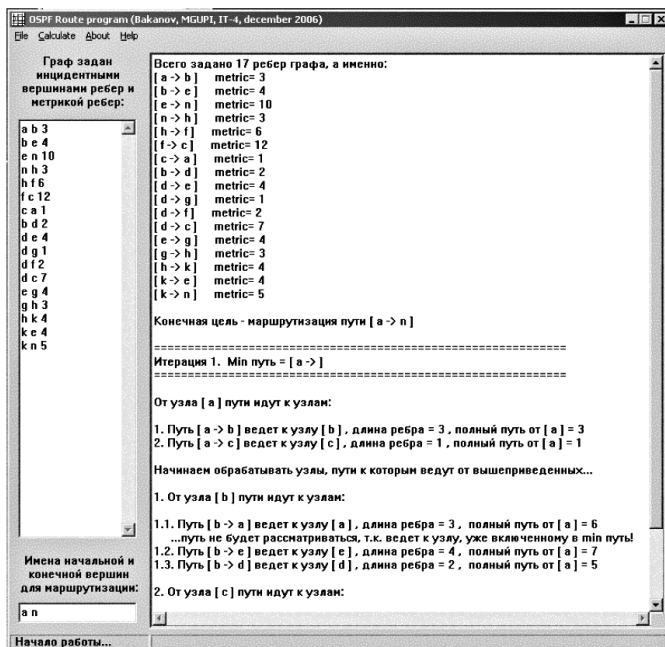


Рис.5.3 Пользовательский интерфейс программы OSPF

5.2 Необходимое оборудование

- IBM PC-совместимая ЭВМ
- предустановленная ОС Windows
- программа OSPF.

5.3 Порядок проведения работы.

Студенты получают индивидуальные варианты топологии сети (сеть представляется аналогично рис.5.1), выполняют описание графа в виде списка инцидентных вершин и метрики, вводят данные в программу OSPF и сохраняют проект задания, проводят расчет и анализируют полученные результаты. По указанию преподавателя моделируется изменение сети (путем уничтожения части путей или добавления новых) и операция поиска кратчайшего маршрута повторяется.

5.4 Типовые задания.

- Составить граф сети заданной топологии
- Промоделировать изменение сети (путем уничтожения части путей или добавления новых)
- Выполнить операцию поиска кратчайшего маршрута.

5.5 Оформление отчета по работе.

В отчет вносится визуальное представление сети, полученный список инцидентных вершин, результаты маршрутизации (список составляющих минимальный путь вершин и результирующая длина пути) для заданной серии расчетов.

5.6 Вопросы к защите работы

- В чем заключается суть процесса маршрутизации компьютерной сети? Какие обязательные процедуры должен включать процесс маршрутизации?
- Какие алгоритмы маршрутизации используются в современных сетях? Какие из них более применимы к относительно небольшим и крупным сетям?

- По какой причине для математического моделирования компьютерной сети удобно использовать представление ее в виде графа?
- Почему такой граф является взвешенным? Неориентированным? В каком случае подобный граф может являться ориентированным?
- Каким образом в данной модели можно учесть величину латентности по каждому выходному порту маршрутизатора?
- Почему веса ребер должны быть неотрицательными? Каков физический смысл нулевой метрики ребра?

ЛАБОРАТОРНАЯ РАБОТА 6.

ИЗУЧЕНИЕ И НАСТРОЙКА АППАРАТНОЙ И ПРОГРАММНОЙ ПОДДЕРЖКИ СЕТЕВОГО ОБЕСПЕЧЕНИЯ МНОГОПРОЦЕССОРНОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ НА ОСНОВЕ LINUX-МАШИН

Цель работы – приобретение практических знаний и навыков в настройке и рациональном использовании сетевого обеспечения многопроцессорной вычислительной системы (МВС) с распределенной памятью архитектуры MPP (*Massively Parallel Processing*, вычислительный кластер) на основе Linux-машин.

6.1 Теоретические основы.

Системное ПО многопроцессорных вычислительных систем (МВС) в подавляющем большинстве строится на основе сетевой ОС Linux как наиболее надежной в эксплуатации и гибкой в настройке (напр., допускающей эксплуатацию в режиме управления с командной строкой без использования лишнего в данном случае графического интерфейса). Значительная вариативность настройки Linux позволяет создавать на ее основе МВС, в которых сотни и тысячи вычислительных узлов функционируют совместно, обеспечивая огромную производительность (конечно, при использовании специальных приемов алгоритмизации и программирования).

Дальнейшее описание соответствует принципам одного из первых высокоуспешных кластерных разработок - проекту BEOWULF (<http://www.beowulf.org>).

Основные положения проекта:

- В качестве *вычислительных узлов* (ВУ) используются широко-распространенные (и готовые к употреблению) блоки - системные платы (*motherboard*) современных ПЭВМ с установленными процессорами (часто двух-или четырехпроцессорные или с двух- или четырехядерными процессорами) и массивами оперативной памяти (ОП), причем загрузка ОС осуществляется со стандартных жестких дисков (бездисковые вычислительные узлы с загрузкой по сети используются редко, т.к. для полноценной работы ОС все же необходим *своппинг* –

возможность временного сброса части ОП на жесткий диск). Практически всегда на каждом ВУ в данный момент времени выполняется *одна пользовательская задача* - не считая, конечно, сетевых и др. *демонов* (*daemon* - программа, выполняющаяся в фоновом режиме без выхода на консоль и обслуживающая запросы программ-клиентов), обеспечивающих функционирование данного ВУ как части МВС.

- Для управления массивом вычислительных блоков служит *управляющая ЭВМ* (УМ); она же выполняет функцию “*front-end машин*” (т.е. обеспечивающей доступ на вычислительный кластер удаленных пользователей по сети InterNet или IntraNet). Именно на управляющей ЭВМ компилируется программа, она же осуществляет управление вычислительными узлами (с помощью команд семейства **rlogin, rsh, rshell** и т.п.).

Все ВУ имеют доступ к копиям выполняемой программы с использованием протокола NFS (*Network File System*) файловой системы (определенный каталог является общим для УМ и всех ВУ; таким же образом реализуется сбор данных выходных файлов от всех экземпляров выполняемой программы). УМ одновременно может являться файл-сервером для хранения пользовательских программ (для этой цели возможно использовать и стороннюю ЭВМ с дисковым массивом значительной емкости). • Для связи УМ и ВУ применяется *коммуникационная среда*, использующая стандартные (напр., Ethernet, Fast- или GigaBit Ethernet) или специализированные *высокоскоростные низколатентные* (Myrinet, SCI, InfiniBand) сетевые технологии.

Обычно применяются две сети – *сеть управления* (именно она служит для управления ВУ со стороны УМ) и *сеть обмена данными* (предназначена для обмена данными между ВУ); причем только от второй сети требуется *максимальное быстродействие* и *минимальная латентность*. В наиболее производительных вычислительных кластерах в отдельную сеть иногда выделяют сеть поддержки NFS (она интенсивно используется в моменты старта/окончания задания и при выводе данных).

Администрирование МВС осуществляется с УМ или посредством удаленного доступа (УД) суперпользователем **root**, рядовые пользователи имеют ограниченный доступ к ресурсам кластера исключительно через УМ в режиме УД. Возможность *захвата* (т.е. перехвата управления удаленной ЭВМ со своего терминала посредством УД по сети) на ВУ имеет исключительно администратор (суперпользователь **root**).

На рис.6.1 изображена *укрупненная* схема сетевых соединений для

вычислительного кластера.

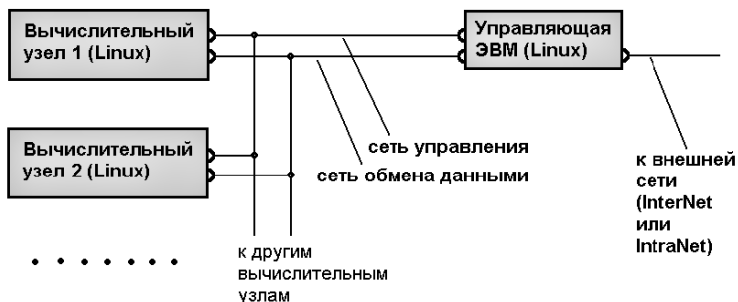


Рис.6.1 Укрупненная схема сетевых коммуникаций многопроцессорной вычислительной системы (вычислительного кластера)

Более подробная схема сетевых соединений (с уточнением IP-адресов и использованием Fast Ethernet для сети управления и Gigabit Ethernet для сети обмена данными) представлена на рис.6.2 (в данном случае используется топология “звезда” относительно коммутатора, применяемый сетевой протокол – TCP/IP).

Технология Gigabit Ethernet позволяет довести пропускную способность сети до 60÷80 Мбайт/сек, однако латентность зачастую выше, чем у Fast Ethernet. Если для повышения быстродействия Fast Ethernet - сети возможно применить технологию Channel Bonding (*связывание каналов* -фактически их дублирование путем совместного использования нескольких сетевых плат и кабелей, причем в Linux начиная с ядер версии 2.4.x эта возможность является стандартно включаемой; см. <http://linux-cluster.org.ru/bonding.php.htm>), то снизить латентность реально лишь с помощью применения (существенно) более дорогостоящего сетевого оборудования.

Технология SCI (*Scalable Coherent Interface*, фирма Dolphin Interconnect Solutions, Inc., <http://www.dolphinics.com>) является международным стандартом (IEEE 1596) высокоскоростных соединений. Фирма Dolphin Interconnect Sol. производит различное оборудование для SCI и в том числе PCI-SCI адаптеры, которые применяются в кластерных системах SCALI. Адаптеры обеспечивают высокую производительность - пропускная способность превышает 300 Мбайт/сек, латентность - до 3 мксек. Коммутаторы обычно не используются, топологии - кольцо (при рекомендуемом числе ВУ до 8), 2D-тор (количе-

ство ВУ до 24) и 3D-тор (при сотнях ВУ).

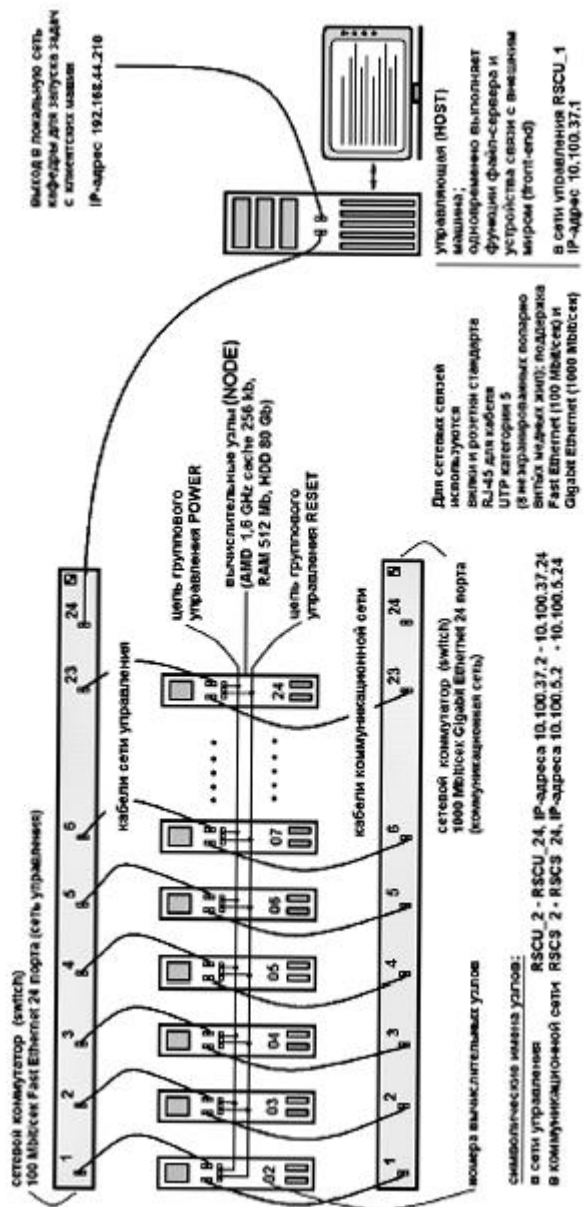


Рис.6.2 Схема сетевых соединений отдельных компонентов

многопроцессорной вычислительной системы

Технология Myrinet (фирма Myricom, Inc., <http://www.myri.com>) требует использования специального коммутатора, пропускная способность до 200 Мбайт/сек, латентность – до 10 мксек.

Технология InfiniBand (консорциум InfiniBand Trade Association, <http://www.infinibandta.org>) использует удаленный прямой доступ в память (RDMA, *Remote Direct Memory Access*), позволяющий доставлять данные непосредственно в память процесса без вовлечения системных вызовов. Для InfiniBand характерно использование коммутаторов, пропускная способность до 1000 Мбайт/сек, латентность – до $4 \div 5$ мксек.

В данном варианте MBC используется операционная система Linux SlackWare 11 (ядро версии 2.6.17); нижеприведенные настройки приведены для нее (но аналогичны SlackWare 9.1, ядро 2.4.22).

На всех ВУ используются два *сетевых интерфейса* (символьных имен, через которые можно обращаться к аппаратным средствам ЭВМ) – eth0 для сети управления (Fast Ethernet) и eth1 для сети обмена данными (Gigabit Ethernet); причем интерфейс eth0 связан со смонтированным на системной плате сетевым адаптером, а eth1 – с дополнительной PCI-сетевой картой Gigabit Ethernet. Для настройки сетевых интерфейсов используется утилита с командной строкой **ifconfig** из каталога **/etc**; для автоматизации настройки целесообразно нужные команды добавить в файл **/etc/rc.d/rc.inet1** (выполняющийся при загрузке ОС скрипт, служащий для настройки сетевых интерфейсов). Эти команды могут быть, например, такими:

```
/etc/ifconfig eth1 10.100.5.N broadcast 10.100.31.255 netmask 255.255.224.0  
/etc/ifconfig eth0 10.100.37.N broadcast 10.100.63.255 netmask 255.255.224.0
```

где

N – номер ВУ ($N=2 \div 24$).

Соответственно символьные имена ВУ есть **rscu_2÷rscu_24** для сети управления и **rscs_2÷rscs_24** для цепи обмена данными, для управляющей машины **rscu_1** и **rscs_1** (см. рис. 6.2).

В случае использования единой физической сети для сетей **rscu_n** и **rscs_n** соответствующие настройки могут быть таковы (нотация **eth0:0** как раз и дает возможность связать еще одну виртуальную сеть с тем же самым сетевым адаптером):

```
/etc/ifconfig eth0 10.100.5.N broadcast 10.100.31.255 netmask 255.255.224.0
/etc/ifconfig eth0:0 10.100.37.N broadcast 10.100.63.255 netmask 255.255.224.0
```

В файле **/etc/HOSTNAME** находится имя текущего ВУ в виде строки **rscu_N.rscu** (где N описано выше).

В файле **/etc/hosts** (доступный для локального компьютера файл соответствий IP-адресов и доменных имен) должен находиться список всех ВУ в виде:

10.100.5.1	rscs_1
.....
10.100.5.24	rscs_24
10.100.37.1	rscu_1
.....
10.100.37.24	rscu_24

Настройка NFS заключается в описании общих для всех ВУ и управляющей машине каталогов **/common** (общесистемный каталог) и **/home** (рабочие каталоги рядовых пользователей); само описание должно находиться в файле **/etc/fstab**. Таким образом, на всех ВУ и УМ должны иметься каталоги **/common** и **/home**, а в файл **/etc/fstab** следует добавить строки:

```
rscu_1:/common  /common  nfs    rsize=8192,wsizе=1024,noac      0  0
rscu_1:/home    /home      nfs    rsize=8192,wsizе=8192,actimeo=1 0  0
```

На всех ВУ и на УМ должно быть разрешено открытие не менее **16384** файлов; это число следует записать в файлы **/proc/sys/fs/file-max** и **/etc/sysctl.conf** на всех ВУ и на управляющей ЭВМ.

В конфигурационных файлах **/etc/hosts.allow** и **/etc/hosts.deny** находятся адреса удаленных ЭВМ, доступ которых по протоколу TCP/IP (с использованием служб **rlogin**, **rsh**, **telnet**, **finger**, **talk**) к данному компьютеру соответственно предоставляется или не предоставляется (если адреса нет ни в одном из указанных файлов, доступ разрешен). Список ЭВМ и их пользователей, которым разрешено выполнять команды **rlogin** (заход на удаленную ЭВМ) и **rsh** (выполнение заданной команды на удаленной ЭВМ), хранится в файле

/etc/hosts.equiv (если ЭВМ или пользователя в этом файле нет, то с этой машины этот пользователь никакими силами ни **rsh**, ни **rlogin** на данную машину выполнить не сможет; причем обратное неверно: если в **/etc/hosts.equiv** пользователь и ЭВМ описаны, то попытка пользователя зайти подлежит дополнительной проверке). В рассматриваемом случае на ВУ в этом файле находится строка **rscu_1 root**, что дает возможность суперпользователю получать полное управление над ВУ. При этом выполняется дополнительная проверка – в файле **/etc/inetd.conf** в строках, отвечающих за настройку TCP-демонов **login** и **shell** необходимо указать следующее (этим подтверждается разрешение суперпользователю **root** заходить с УМ на ВУ без ввода пароля):

```
shell stream tcp nowait.10000 root /usr/sbin/tcpd in.rshd -lh
login stream tcp nowait root /usr/sbin/tcpd in.rlogind -lh
```

Параметр настройки **nowait.10000** отменяет режим выключения InterNet-демона **inetd** на минуту в случае излишне частых обращений; этот режим по умолчанию включен для пресечения хакерских атак типа DoS (*Denial of Service*, отказ в обслуживании).

Типовым приемом изменения настроек каждого ВУ является заход суперпользователя **root** на этот ВУ с УМ с помощью команды **rlogin rscu_N** (возврат по **logout**); при вышеуказанных настройках пароль затребован не будет.

Для удаленной работы пользователей на МВС применяются клиентские программы **TelNet**, **SSH**, **PuTTY** и др. (для любителей FARa может быть интересен WinSCP-базирующийся Far SCP/SFTP client plugin; подробнее см. <http://winscp.net/eng/far.php>).

Администрирование МВС осуществляется суперпользователем **root** непосредственно с консоли УМ или в режиме УД доступа с использованием разработанного ИПМ им. В.М.Келдыша специализированного WEB-интерфейса, при этом необходимые действия выполняются Perl-скриптами под управлением функционирующего на МВС сервера Apache.

Дополнительными заданиями (*исследовательского уровня*) могут являться определение зависимости межузловой скорости обмена данными от размера передаваемого пакета и оценка времени выполнения

основных операций по обмену данными (величины *латентности* в первую очередь).

6.2 Необходимое оборудование

- многопроцессорная вычислительная система (МВС) на основе IBM PC-совместимых вычислительных узлов под управлением Linux SlackWare
- управляющая ЭВМ с системой удаленного доступа к ресурсам МВС.

6.3 Порядок проведения работы.

На первом этапе работы студенты знакомятся с аппаратной частью коммуникационной среды МВС (сетевыми коммутаторами, сетевыми картами, кабельной системой); по заданию преподавателя может быть проведен частичный демонтаж системы (с восстановлением исходной схемы).

На втором этапе изучается функционирование программной части сетевой компоненты МВС.

6.4 Типовые задания:

- Работа с консоли технического обслуживания (дисплей + клавиатура) по изменению сетевых настроек с целью изменения прав доступа по сети с УМ на конкретный ВУ.
- Тестирование сетевой защищенности ВУ (т.е. возможности сетевого доступа с УМ на ВУ и извне УМ (по IntraNet-сети кафедры) к ВУ.
- Проверка возможности захода с управляющей машины на ВУ для суперпользователя и произведения нужных изменений.
- Оптимизация скорости доступа к ресурсам жестких дисков на ВУ (применение команды **hdparm** в режиме УД с правами суперпользователя), производится под руководством преподавателя.
- Настройка параметров функционирования NFS для обеспечения наибольшей скорости доступа к общим ресурсам (производится по указанию преподавателя).
- Определение зависимости *реальной* (с учетом латентности) скорости передачи данных от размера передаваемого пакета. Целесо-

образно воспользоваться программой PROG_MPI

- Определение времени выполнения основных операций по обмену данными (включая величину латентности); рекомендуется тестовая программа MPI_TEST (<http://parallel.ru>.)
- Влияние алгоритма решения задачи и размера данных на величину сетевой деградации МВС (на примере рассмотрения алгоритмов умножения матриц с различной величиной гранулы зерна параллелизма).

6.5 Оформление отчета по работе.

В отчет должна входить схема сетевых соединений МВС (с указанием распределения IP-адресов, назначением и количественными характеристиками каждой сети и указанием фирм-изготовителей сетевого оборудования), твердые копии содержимого основных файлов сетевых настроек, результаты выполнения заданных преподавателем индивидуальных заданий.

6.6 Вопросы к защите работы

- Какова роль коммуникационной среды в конструкции МВС? Какие сети используются и каковы требования к каждой?
- Какие быстродействующие и низколатентные сетевые технологии используются в МВС?
- Каким образом удаленные пользователи осуществляют доступ к ресурсам МВС? Каким образом УМ управляет ВУ? Какими путями реализуется защита ВУ от несанкционированного доступа по сети?
- Каким программным путем достигается использование для различных сетей единственного сетевого адаптера? Что такое *сетевой интерфейс* в терминологии Linux?
- С какой целью в МВС используется технология NFS?
- Какими путями осуществляется техническое обслуживание и настройка сетевого программного обеспечения ВУ?
- В каком случае отрицательное влияние латентности на производительность МВС более значительное – при обмене относительно большими или малыми пакетами данных? Почему?

Какой компонент в коммуникационной сети данной МВС является слабым (т.е. существенно ограничивает пропускную способность сети)? Какая архитектура сети может быть предложена для устранения

этого явления?

СПИСОК ЛИТЕРАТУРЫ

1. Фред Халсалл. Передача данных, сети компьютеров и взаимосвязь открытых систем. — М.: Радио и связь, 1995.
2. Столлипс В. Современные компьютерные сети, 2-е изд. — СПб.: Питер, 2003.
3. Куроуз Дж., Росс К. Компьютерные сети, 4-е изд. — СПб.: Питер, 2004.
4. Таненбаум Э. Компьютерные сети, 4-е изд. — СПб.: Питер, 2002.
5. Марк Спортак, Френк Паппас и др. Компьютерные сети и сетевые технологии. - ТИД «ДС», 2002.
6. Куинн Л., Рассел Р. Fast Ethernet. - BHV-Киев, 1998.
7. Фейт Сидни. TCP/IP. Архитектура, протоколы, реализация. — М.: Лори, 2000.
8. Стивен Браун. Виртуальные частные сети. — М.: Лори, 2001.
9. Шринивас Вегешна. Качество обслуживания в сетях IP. — Вильяме, 2003.
10. Оливер Ибе. Сети и удаленный доступ. Протоколы, проблемы, решения. — ДМК Пресс, 2002.
11. Стерн, Монти. Сети предприятий на основе Windows NT для профессионалов/Перев. с англ. — СПб.: Питер, 1999.
12. Блэк Ю. Сети ЭВМ: протоколы стандарты, интерфейсы/Перев. с англ. — М.: Мир, 1990.
13. Ричард Стивене. Протоколы TCP/IP. Практическое руководство. — СПб.: БХВ, 2003.
14. Щербо В. К. Стандарты вычислительных сетей. Взаимосвязи сетей. Справочник. — М.: Кудиц-образ, 2000.
15. Олифер Н. Агрегирование каналов в локальных сетях. — LAN/Журнал сетевых решений, февраль 2002.
16. Олифер В., Олифер Н. Виртуальные частные сети на основе MPLS. — LAN/ Журнал сетевых решений, январь 2002.
17. Олифер Н. Маршрутизатор как он есть. — LAN/Журнал сетевых решений. июль-август 2001.
18. Олифер В. Г., Олифер Н. А. Новые технологии и оборудование IP-сетей. — СПб.: БХВ-Санкт-Петербург, 2000.
19. Олифер В. Г., Олифер Н. А. Сетевые операционные системы.

— СПб.: Птер. 2001.

Учебное издание

**ЛАБОРАТОРНЫЙ ПРАКТИКУМ ПО ПРЕДМЕТУ
“КОМПЬЮТЕРНЫЕ СЕТИ”**

Учебно-методическое пособие

Составители: КОТЕЛЬНИКОВ Анатолий Александрович
МАСЛОВА Наталья Васильевна
ВЕНТ Дмитрий Павлович
КОТЕЛЬНИКОВА Марина Георгиевна

Редактор Туманова Е.М.
Подписано в печать2009. Формат 60×84 1/16.
Бумага Svetocopy. Отпечатано на ризографе.
Усл. печ. л. Уч.-изд.л.
Тираж 50. Заказ №

ГОУ ВПО «Российский химико-технологический университет им.
Д.И.Менделеева»
Новомосковский институт (филиал). Издательский центр
Адрес университета: 125047, Москва, Миусская пл., 9
Адрес института: 301670, Новомосковск, Тульской обл., Дружбы, 8