МЕТОДИЧЕСКОЕ ПОСОБИЕ

к выполнению лабораторных работ

по курсу

«Вычислительные машины, системы и сети»

Ростов н/Д

2024

**Аннотация**

Методическое пособие к выполнению лабораторных работ по дисциплине «Вычислительные машины, системы и сети». Выполнение работ ориентировано на использование персональных компьютеров и программ ВАРИАНТ и Computer Network Simulator. Методическое пособие предназначено для студентов заочной формы обучения по направлению15.04.03 «Автоматизация технологических процессов и производств».

Содержание

Лабораторная работа №1 4

Лабораторная работа №2 8

Лабораторная работа №3 15

Лабораторная работа №4 22

Список литературы 28

Лабораторная работа №1

Моделирование полупроводниковых постоянных запоминающих устройств на компьютере

Цель работы:

1. Ознакомление со структурой и методами увеличения объема памяти постоянных запоминающих устройств (ПЗУ).

2. Приобретение навыков моделирования ПЗУ на компьютере средствами программы ВАРИАНТ.

Порядок выполнения работы:

1. Ознакомиться с содержанием лабораторной работы и ее оборудованием.

2. Изучить методы наращивания объема и разрядности модулей ПЗУ.

3. Изучить систему логического моделирования цифровых микросхем и схем на их основе ВАРИАНТ.

4. Синтезировать на ПЭВМ схему ПЗУ с организацией 64 x 8 бит.

5. Синтезировать на ПЭВМ схему ПЗУ с организацией 32 x 16 бит.

6. Сформулировать выводы по результатам работы.

Краткие теоретические сведения

Запоминающие устройства, в т.ч. и ПЗУ предназначены для хранения информации. Основными характеристиками ПЗУ являются их разрядность и объем, быстродействие, потребляемая мощность, устойчивость к возникновению ошибок.

Отличительной особенностью всех ПЗУ является их энергонезависимость. ПЗУ реализуются на БИС ПЗУ четырех типов: с масочной записью данных (МПЗУ или ROM), однократно программируемые (ППЗУ или PROM), многократно программируемые со стиранием ультрафиолетовым освещением (СППЗУ или EPROM) и многократно программируемые с электрическим стиранием (ЭСППЗУ или EEPROM) [1].

Обобщенная структурная схема БИС ПЗУ, представленная на рисунке 1, включает следующие функциональные узлы: накопитель Нк, дешифратор кода адреса ДШ, устройство вывода УВ, устройство управления УУ.

… …

. . .

. Нк . .

… …

. . .

. . .

… …

1

ДШ

Аm-1 ЯП

. .

Адресная . j .

шина Aj

. m .

. 2 . ЭП

Ао

\_\_\_

Разрешение CEO

УУ

выходов

\_\_ 1 . . . . . . n

Вход CS

выборки

DOo

УВ

. Вых.

Вход PR . шина

программирования . данных

DOn-1

Рисунок 1 - Обобщенная структурная схема БИС ПЗУ и назначение

входов/выходов

Накопитель представляет собой совокупность элементов памяти (ЭП), объединенных в матрицу. В матрице ЭП размещены на пересечениях горизонтальных и вертикальных проводников, называемых соответственно строками и столбцами. Каждый ЭП может хранить один бит (0 или 1) информации. Для хранения n-разрядного слова требуется n элементов памяти. Совокупность ЭП, предназначенная для хранения одного слова, называется ячейкой памяти (ЯП).

Дешифратор ДШ преобразует m- разрядный код адреса в активный сигнал выборки ЯП. Число ЯП в накопителе равно Ni=2 и называется объемом БИС ПЗУ. Произведение Ni x ni отражает информационную емкость или организацию i- ой микросхемы ПЗУ. Устройство вывода УВ предназначено для усиления и нормализации сигналов данных, подаваемых при программировании на выходы DOo. . .DOn-1 и снимаемых при считывании с тех же выходов. Формирователи выходов данных выполняются по схеме с открытым коллектором или по схеме с тремя состояниями. БИС ПЗУ, выполненные по схеме с тремя состояниями, имеют вход разрешения выходов CEO. Программируемые БИС ПЗУ имеют также вход программирования PR, а для записи данных при программировании используются выводы данных БИС.

Устройство управления формирует внутренние сигналы для воздействия на функциональные узлы ПЗУ, соответствующие внешним сигналам управления: "Считывание", "Выбор кристалла (микросхемы)", "Программирование", "Разрешение выходов".

ПЗУ больших объемов обычно разбиваются на несколько модулей, каждый из которых имеет объем, выбираемый исходя из возможностей реализации на БИС ПЗУ, и выполняется в виде автономного конструктивного модуля с заданными объемом N и разрядностью n, т.е. с организацией N x n.

Для получения требуемой разрядности n ПЗУ k=n/ni БИС ПЗУ объединяются в блоки ПЗУ как показано на рисунке 2. Все БИС блока имеют общие входы адресов, выборки БИС и разрешения выходов.

\_\_ \_\_\_

CS A(m) CEO

m

m

CS A(m) CEO

БИС ПЗУ (к-1) (Ni x ni)

DO (ni)

# CS A(m) CEO

БИС ПЗУ0 (Ni x ni)

DO (ni)

EO ¦

¦БИС ПЗУ(k-1)(Ni x ni) БИС ПЗУ0 (Ni x ni) ¦

¦ DO(ni) ¦ ¦ DO (ni) ¦

L----------T--------- -- L----------T-----------

­

DO(n) , n=k ni

Рисунок 2 - Схема наращивания разрядности ПЗУ

Выходы данных БИС образуют n- разрядные выходы данных блока. При этом БИС ПЗУ0 хранит младшие ni разрядов данных, а БИС ПЗУ(k-1) - cтаршие ni разрядов. Для получения требуемого объема N запоминающего устройства L=N/Ni блоков ПЗУ соединяются, как показано на рисунке 3. Для их адресного разделения обычно используют адресный дешифратор АДШ, число выходов L которого равно числу блоков ПЗУ, а число входов l=log L. Все входы и выходы блоков ПЗУ, кроме входов выборки, объединяют между собой, а входы выборки соединяют с соответствующими выходами адресного дешифратора. Т.о., модульПЗУ с

\_\_

CS A(l) , l=logL

\_\_\_\_

l A(m) CEO

АДШ

m

m

# CS A(m) CEO

Блок ПЗУ0 (Ni x n)

БИС ПЗУ х k

DO(n)

CS A(m) CEO

Блок ПЗУ (L-1) (Ni x n)

БИС ПЗУ х k

DO(n)

n n

n

DO(n)

Рисунок 3 - Схема наращивания объема ПЗУ.

организацией N x n имеет входы адреса разрядностью (l+m)=log N, выходы данных разрядностью n и общий вход разрешения СЕО выходов. Вход выборки модуля CS используется для размещения модуля в требуемом месте адресного пространства запоминающего устройства, а вход разрешения выходов - для подачи сигнала чтения RD.

В ПЗУ, которые не имеют входа разрешения выходов, сигнал чтения подается на один из входов выборки CS.

Применение ПЗУ не ограничивается построением модулей памяти. На основе ПЗУ создаются устройства хранения и индикации кодовых последовательностей, формирователи цифровых и аналоговых сигналов произвольной формы, разнообразные знакогенераторы [1].

Особенности работы моделей ПЗУ

Моделирование ПЗУ и схем на их основе осуществляется на компьютере с помощью программы ВАРИАНТ версии 2.1. При работе с программой необходимо учитывать особенности работы моделей схемотехнических элементов в этой среде.

Модель ПЗУ по логике работы соответствует однократно программируемой микросхеме К155РЕ3, имеющей 5 адресных входов, вход разрешения CS и 8 выходов. Организация 32 x 8 бит. Записанная информация появляется на выходах при подаче на вход CS логического 0. При подаче на вход CS логической 1 на всех выходах устанавливаются логические 1. Незадействованные (висящие в воздухе) входы ПЗУ воспринимаются как входы с поданным на них уровнем логической 1.

Для занесения информации в реальную микросхему используется программатор. Для занесения информации в модели ПЗУ1 или ПЗУ2 необходимо нажать клавиши F8 или F9 соответственно. После этого на экране появится окно, отображающее информацию, соответствующую прошивке моделируемой микросхемы. С помощью клавишей управления курсором достаточно поместить перекрестье логического промптера на изменяемый бит и нажать клавишу Enter. Т.о., информация в ПЗУ1 или ПЗУ2 может неоднократно изменяться, что принципиально невозможно в микросхеме-прототипе.

Другие особенности работы моделей, а также библиотека элементов и режимы работы программы ВАРИАНТ приведены в файле variant.doc и в руководстве пользователя [2].

Лабораторная работа №2

Программируемая оболочка построения виртуальных вычислительных сетей

Цель работы:

1. Изучение основ сетевого взаимодействия.

2. Изучение программируемой оболочки построения виртуальных вычислительных сетей.

3. Приобретение навыков работы с программой Computer Network Simulator (CNS).

Порядок выполнения работы

1. Изучить программу моделирования виртуальных вычислительных сетей Computer Network Simulator (CNS) [3, 7].

2. Произвести кроссирующее подключение двух удаленных рабочих станций (PC1 и PC2) средствами отрезка кабеля. В качестве допустимых IP-адресов, необходимо использовать адреса из диапазона 1 (см. таблицу заданий, вариант соответствует двум последним цифрам студенческого билета).

3. Запустив эмулятор терминала на удаленных рабочих станциях настроить соответствующие IP-адреса (команда ifconfig). В оболочке командного интерпретатора, выполнить команду ping для проверки доступности рабочих станций.

4. С помощью виртуального устройства «Концентратор», эмулирующего работу реального Ethernet концентратора сети и нескольких отрезков кабеля произвести объединение удаленных узлов (PC3 и PC4). В качестве допустимых IP-адресов, необходимо использовать адреса из диапазона 2 (см. таблицу вариантов заданий).

5. Запустив эмулятор терминала, настроить IP-адреса рабочих станций. С помощью утилиты командной строки ping, реализованной в оболочке виртуального терминала проверить доступность удаленных узлов.

6. Добавить к проекту сетевое устройство «Коммутатор», эмулирующее работу реального 8 портового Ethernet коммутатора сети и с помощью нескольких отрезков кабеля объединить удаленные узлы (PC5 и PC6). В качестве IP-адресов удаленных узлов, использовать адреса из диапазона 192.168.99.0/30.

7. С помощью утилиты командной строки ping, проверить доступность удаленных узлов.

8. С помощью пункта «Экспортировать в html…» меню «Проект» сформировать электронный отчет проекта в формате html, предварительно сохранив xml-проект CNS. Файл html-проекта следует распечатать и прикрепить к основному отчету по лабораторной работе.

9. Сформулировать выводы по результатам работы.

Краткие теоретические сведения

С развитием сетевых технологий возникла необходимость в создании стандартов взаимосвязи в сетях открытых систем (вычислительных сред, состоящих из аппаратных и программных продуктов и использующих технологии, разработанные в соответствии с общедоступными и общепринятыми стандартами). Модель OSI (Open System Interconnection), часто называемая эталонной моделью, разбивает все процессы взаимодействия и передачи данных по сети на семь уровней. Этой же моделью для каждого уровня определяются выполняемые ими функции, протоколы (набор правил и процедур), даются термины и определения основных понятий. При передаче данных по сети связь уровней между собой осуществляется посредством интерфейсов, а пересылка с одного узла на другой – с помощью протоколов.

На нижних, физическом и канальном уровнях модели OSI используются в т.ч. и хорошо стандартизованные протоколы Ethernet («эфирная сеть»), позволяющие применять во всех сетях одинаковую аппаратуру: сетевые адаптеры, концентраторы, коммутаторы и т.д. Типы физических интерфейсов сетей Ethernet различаются на основе используемой среды передачи: коаксиальный кабель, витая пара, оптоволокно и радиоканал.

В большинстве сетевых технологий для однозначной адресации физических интерфейсов используются аппаратные МАС-адреса (Media Access Control), представляющие собой двоичное число длиной 48 бит. При этом первые три байта (224 или 16 млн.) определяют производителя сетевого оборудования, а остальные идентифицируют само устройство, выпущенное этим производителем. По прогнозам IEEE (Institute of Electrical and Electronic Engineers) этого запаса адресов хватит до 2100г.

МАС-адреса лежат в основе работы протоколов физического уровня. На третьем, сетевом уровне модели OSI используется иерархическая адресация, наиболее известным представителем которой являются сетевые IP-адреса протокола IP (Internet Protocol) стека TCP/IP (Transmission

Control Protocol). IP-адрес представляет собой двоичное число, имеющее длину 32 бита, которые разделяются точками на 4 раздела – октета. Каждый октет в десятичном формате записи может иметь значение от 0 до 255, например 130.57.30.56.

IP-адрес состоит из двух логических частей: номера сети и номера узла в сети. Границу между этими частями определяет маска – двоичное число (также 4 октета), содержащее непрерывную последовательность 1 в тех разрядах, которые в IP-адресе соответствуют номеру сети. Последовательность 0 в маске соответствует разрядам номера узла, например 255.255.255.0. Часто для записи масок используется формат «IP-адрес/префикс». Префикс определяет число 1 в маске. Например, 185.23.44.206/16 означает, что в указанном IP-адресе под номер сети отведено 16 двоичных разрядов. Для стандартных классов сетей маски имеют следующие значения: класс А – 255.0.0.0; класс В – 255.255.0.0; класс С – 255.255.255.0.

Некоторые IP-адреса интерпретируются особым образом. Так, *неопределенный* адрес состоит только из двоичных 0 и обозначает тот же самый узел сети, который сгенерировал пакет данных (под пакетом или кадром понимается единица информации, пересылаемая из одного узла сети на другой). *Ограниченный широковещательный* адрес состоит только из логических 1 и означает, что пакет с таким адресом должен рассылаться всем узлам только той сети, что и источник пакета. IP-адрес, первый октет которого равен 127, является *внутренним* и не передается в сеть. Как правило, он используется для тестирования программ. Для *широковещательного* адреса характерно наличие логических 1 в разрядах, соответствующих номеру узла. Пакет, имеющий такой адрес, рассылается всем узлам сети, номер которой указан в адресе назначения.

Для изучения программируемой оболочки построения виртуальных вычислительных сетей предлагается создать проект виртуальной сети, основываясь на приведенной ниже схеме, и, расположив элементы проектируемой сети, произвести конфигурирование узлов средствами CNS [6, 7]. Первый фрагмент схемы не использует сетевого оборудования, кроме специального кросс-кабеля, соединяющего между собой сетевые адаптеры рабочих станций РС1 и РС2. При реальном кроссирующем соединении передающие контакты одной стороны подключаются кабелем к принимающим контактам другой стороны и наоборот, т.е. имеет место перекрестное соединение пар проводников. Соединить три и более станций таким образом уже невозможно, поэтому для объединения рабочих станций в сети используются обычный «прямой» кабель и сетевое оборудование.



Варианты адресации сетей

|  |  |  |
| --- | --- | --- |
| Вариант задания | Диапазон адресов 1 | Диапазон адресов 2 |
| 01  02  03  04  05  06  07  08  09  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35 | 10.1.0.0/16  172.20.2.0/24  10.3.0.0/24  192.168.4.0/24  172.30.5.0/24  10.6.0.0/16  10.7.0.0/24  172.18.8.0/24  192.168.9.0/24  192.168.10.0/24  172.21.11.0/24  10.12.0.0/16  192.168.13.0/24  172.24.0.0/12  10.15.0.0/24  192.168.16.0/24  172.27.17.0/24  10.18.0.0/16  192.168.19.0/24  192.168.20.0/24  172.21.1.0/24  172.23.22.0/24  192.168.0.0/16  10.24.0.0/16  172.27.0.0/16  192.168.26.0/24  10.27.0.0/24  172.28.0.0/24  10.29.0.0/16  172.29.30.0/24  192.168.31.0/24  172.16.32.0/24  10.0.73.0/24  192.168.2.0/24  10.35.0.0/16 | 192.168.1.0/24  192.168.0.0/16  172.16.0.0/12  10.4.0.0/16  10.0.0.0/8  192.168.0.0/16  172.17.7.0/24  192.168.0.0/16  10.0.0.0/8  10.10.0.0/16  192.168.0.0/16  192.168.0.0/16  10.13.0.0/16  10.14.0.0/16  192.168.0.0/16  10.16.0.0/16  10.0.0.0/24  192.168.0.0/16  10.19.0.0/24  172.20.1.0/24  10.0.0.0/16  10.10.0.0/16  172.23.23.0/24  192.168.24.0/24  10.25.1.0/24  10.26.1.0/24  172.17.110.0/24  192.168.0.0/16  192.168.0.0/16  10.30.0.0/16  10.31.0.0/16  192.168.32.0/24  172.31.33.0/24  172.18.34.0/24  192.168.20.0/24 |

|  |  |  |
| --- | --- | --- |
| Вариант задания | Диапазон адресов 1 | Диапазон адресов 2 |
| 36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85 | 172.19.34.0/24  10.0.37.0/24  172.21.34.0/24  192.168.39.0/24  172.23.34.0/24  172.17.41.0/24  10.42.42.0/24  172.26.34.0/24  172.22.44.0/24  172.28.34.0/24  10.46.0.0/16  10.10.47.0/24  172.31.34.0/24  10.0.0.0/16  10.50.0.0/24  172.27.34.0/24  10.52.0.0/16  192.168.53.0/24  172.25.0.0/16  192.168.5.0/24  10.56.0.0/16  172.27.57.0/24  172.28.58.0/24  172.30.0.0/24  10.0.1.0/24  10.61.0.0/16  172.30.34.0/24  10.63.0.0/16  192.168.64.0/24  172.29.34.0/24  10.66.66.0/24  172.27.67.0/24  192.168.68.0/24  10.69.0.0/16  172.20.70.0/24  172.22.34.0/24  10.72.0.0/16  10.73.0.0/16  192.168.74.0/24  172.25.34.0/24  10.76.0.0/16  10.77.0.0/16  192.168.78.0/24  10.79.1.0/24  10.10.80.0/24  172.24.34.0/24  10.82.0.0/16  192.168.8.0/24  172.24.48.0/24  10.10.85.0/24 | 10.36.36.0/24  192.168.3.0/24  192.168.38.0/24  10.39.0.0/16  10.40.0.0/16  192.168.41.0/24  172.18.0.0/16  10.9.10.0/24  192.168.44.0/24  10.45.1.0/24  192.168.46.0/24  172.24.7.0/24  10.48.0.0/16  172.16.1.0/24  192.168.50.0/24  10.1.51.0/24  192.168.52.0/24  172.25.53.0/24  192.168.0.0/16  172.26.0.0/16  172.26.56.0/24  10.57.0.0/16  10.58.1.0/24  10.59.0.0/16  172.20.60.0/24  172.21.61.0/24  10.62.0.0/16  172.23.63.0/24  172.24.64.0/24  192.168.65.0/24  192.168.66.0/24  192.168.67.0/24  172.30.68.0/24  172.29.96.0/24  10.70.0.0/16  192.168.71.0/24  172.22.72.0/24  172.23.73.0/24  172.18.74.0/24  10.75.0.0/24  172.16.76.0/24  192.168.0.0/16  172.18.78.0/24  172.17.19.0/24  192.168.80.0/24  10.81.0.0/16  172.24.82.0/24  172.28.83.0/24  10.10.84.0/24  192.168.85.0/24 |
| Вариант задания | Диапазон адресов 1 | Диапазон адресов 2 |
| 86  87  88  89  90  91  92  93  94  95  96  97  98  99  00 | 192.168.6.0/24  10.78.0.0/16  10.88.1.0/24  192.168.89.0/24  10.90.0.0/16  10.91.1.0/24  192.168.29.0/24  10.93.0.0/16  10.94.0.0/16  192.168.95.0/24  10.96.1.0/24  10.97.0.0/16  192.168.1.0/24  10.99.0.0/16  172.30.0.0/24 | 172.26.86.0/24  192.168.87.0/24  192.168.9.0/24  172.29.89.0/24  192.168.90.0/24  192.168.0.0/16  172.28.92.0/24  172.23.93.0/24  172.30.94.0/24  10.95.0.0/16  172.19.96.0/24  192.168.11.0/24  172.18.98.0/24  172.17.99.0/24  10.0.10.0/24 |

Лабораторная работа №3

Структуризация локальных вычислительных сетей с помощью коммутаторов

Цель работы:

1. Изучение коммутаторов вычислительных сетей.

2. Изучение принципов функционирования коммутаторов в сетях с

помощью программы Computer Network Simulator (CNS).

Порядок выполнения работы

1. Произвести размещение шести узлов удаленных рабочих станций (PC1 - PC6), двух устройств коммутаторов и одного устройства концентратора средствами CNS в соответствии со схемой, приведенной выше. Объединить удаленные узлы и соответствующие порты коммутаторов и концентратора отрезками кабеля.

2. Запустив эмулятор терминала на удаленных рабочих станциях назначить каждой станции соответствующий IP-адрес (команда ifconfig). В качестве допустимых IP-адресов использовать адреса из диапазона 2 (см. таблицу заданий лабораторной работы №2, вариант соответствует двум последним цифрам студенческого билета).

3. В оболочке командного интерпретатора, выполнить команду ping для проверки доступности всех удаленных узлов с рабочих станций РС1, РС4 и РС5.

4. Проследить направление рассылки кадров в сети. Отметить узел отправителя и узел получателя в каждом случае, а также все узлы, участвующие в широковещательной рассылке кадра. Отметить отличия при обработке кадров концентратором и коммутаторами.

5. Запустить эмулятор терминала на каждом из устройств коммутаторов и с помощью команды mactable просмотреть содержимое таблицы МАС-адресов коммутатора.

6. Убедиться в достоверности соответствующего МАС-адреса номеру физического порта коммутатора, проверив физический адрес удаленной рабочей станции командой ifconfig.

7. С помощью пункта «Экспортировать в html…» меню «Проект» сформировать электронный отчет проекта в формате html, предварительно сохранив xml-проект CNS. Файл html-проекта следует распечатать и прикрепить к основному отчету по лабораторной работе.

8. На отчете проекта выделить границы широковещательного домена. Отметить узлы, участвующие в коммутировании кадров и в передаче широковещательных рассылок. Перечислить используемые адреса канального уровня.

9. Сформулировать выводы по результатам работы.

Краткие теоретические сведения

Коммутирующие концентраторы (Switched Hubs) или просто коммутаторы (Switches) позволяют разделить единую сеть на несколько сегментов для увеличения допустимого размера сети или с целью снижения нагрузки (*трафика*) в отдельных частях сети. Коммутаторы появились в ответ на растущие потребности в повышении пропускной способности связей высокопроизводительных серверов с сегментами рабочих станций. В отличие от концентраторов, которые распространяют трафик от одного подключенного устройства ко всем остальным, коммутаторы передают данные только непосредственно получателю, исключение составляет широковещательный трафик всем узлам сети. Это повышает производительность и безопасность сети, избавляя остальные сегменты сети от необходимости (и возможности) обрабатывать данные, которые им не предназначались.

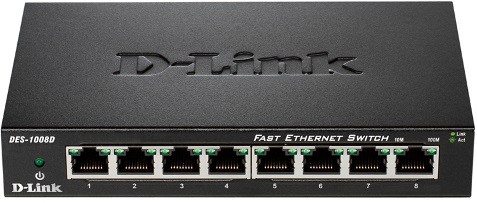
Коммутаторы работают на канальном уровне модели OSI, и потому в общем случае могут только объединять узлы одной сети по их MAC-адресам. Для этого коммутатор хранит в своей ассоциативной памяти таблицу коммутации, в которой указывается соответствие MAC-адреса

|  |  |
| --- | --- |
| *Пример коммутационной таблицы* | |
| Физический адрес | Порт |
| 00-53-45-00-00-00 | Port 05 |
| 35-F3-54-03-49-A0 | Port 12 |
| 2B-51-28-E1-00-5B | Port 01 |

узла порту коммутатора. При включении коммутатора эта таблица пуста, и он работает в режиме обучения. В этом режиме поступающие на какой-либо порт данные передаются на все остальные порты коммутатора. При этом коммутатор анализирует кадры (фреймы) и, определив MAC-адрес отправителя, заносит его в таблицу. Впоследствии, если на один из портов коммутатора поступит кадр, предназначенный для узла, MAC-адрес которого уже есть в таблице, этот

кадр будет передан только через порт, указанный в таблице. Если MAC-адрес получателя не ассоциирован с каким-либо портом коммутатора, то кадр будет отправлен на все порты. Со временем коммутатор строит полную таблицу для всех своих портов, и в результате трафик локализуется.

В коммутаторах реализуется три режима коммутации. Каждый из них характеризуется определенными временем задержки и степенью надежности передачи.

1. *С промежуточным хранением* (Store-and-Forward). Коммутатор читает всю информацию в кадре, проверяет его на отсутствие ошибок, выбирает порт коммутации и после этого посылает в него кадр. Главное отличие устройств, реализующих этот способ коммутации, заключается в наличии внутренней буферной памяти FIFO всех ретранслируемых пакетов. Коммутаторы, работающие по этой технологии, представляют собой наиболее дорогие, сложные и совершенные устройства.
2. *Сквозной* (Cut-Through). Коммутаторы этой технологии самые простые и быстрые, т.к. не производят буферирования пакетов в целом, а только считывают в кадре 6-байтовый адрес назначения и после выполняют коммутацию. Для этого способа коммутации характерны наименьшие задержки при передаче, однако при этом ретранслируются любые пакеты с нормальной головной частью, в том числе и заведомо ошибочные пакеты. Связано это с отсутствием в данной технологии средств обнаружения ошибок. Таким образом, ошибки одного сегмента ретранслируются в другой сегмент, что приводит к снижению пропускной способности сети в целом.
3. *Бесфрагментный* (Fragment-Free) или *гибридный*. Этот режим является модификацией сквозного режима. Передача осуществляется после фильтрации фрагментов коллизий (пакеты размером менее 512 бит автоматически отбрасываются из буфера, чего не делается в сквозном режиме). При малой нагрузке и низком уровне ошибок реализуется более быстрая технология Cut-Through, а при большой нагрузке и значительном количестве ошибок – более медленный, но более качественный режим Store-and-Forward.

Различают *симметричную* и *ассиметричную* коммутацию. Свойство симметрии при коммутации позволяет дать характеристику коммутатора с точки зрения ширины полосы пропускания для каждого его порта. Симметричный коммутатор обеспечивает коммутируемые соединения между портами с одинаковой шириной полосы пропускания, например, когда все порты имеют ширину пропускания 10 Мб/с или 100 Мб/с.

Асимметричный коммутатор (технологии Store-and-Forward) обеспечивает коммутируемые соединения между портами с различной шириной полосы пропускания, что позволяет комбинировать соединения сегментов с шириной полосы пропускания 10 Мб/с и 100 Мб/с или 100 Мб/с и 1000 Мб/с. В результате часть портов коммутатора может работать, например, с сетью Fast Ethernet, а другая часть – с Gigabit Ethernet, причем в ряде случаев коммутаторы автоматически настраивают свои порты на скорость передачи подключенного к порту сегмента.

Асимметричная коммутация используется в случае наличия больших сетевых потоков типа клиент-сервер, когда многочисленные пользователи обмениваются информацией с сервером одновременно, что требует большей ширины пропускания для того порта коммутатора, к которому подсоединен сервер, с целью предотвращения переполнения в этом порте. Для того, чтобы направить поток данных с порта 100 Мб/с на порт 10 Мб/с без опасности переполнения на последнем, асимметричный коммутатор должен иметь буфер памяти.

Асимметричный коммутатор также необходим для обеспечения большей ширины полосы пропускания каналов между коммутаторами, осуществляемых через вертикальные кросс-соединения, или каналов между сегментами магистрали.

Для временного хранения пакетов и последующей их отправки по нужному адресу коммутатор может использовать буферизацию (коммутаторы типа Store-and-Forward). Буферизация может быть также использована в том случае, когда порт пункта назначения занят. Буфером FIFO является область памяти, в которой коммутатор хранит передаваемые данные.

Буфер памяти может использовать два метода хранения и отправки пакетов — *буферизация по портам* и *буферизация с общей памятью*. При буферизации по портам, пакеты хранятся в очередях, которые связаны с отдельными входными портами. Пакет передается в выходной порт только тогда, когда все пакеты, находившиеся впереди него в очереди, были успешно переданы. При этом возможна ситуация, когда один пакет задерживает всю очередь из-за занятости порта его пункта назначения. Эта задержка может происходить даже в том случае, когда остальные пакеты могут быть переданы на открытые порты их пунктов назначения.

При буферизации в общей памяти, все пакеты хранятся в общем буфере памяти, который используется всеми портами коммутатора. Количество памяти, отводимой порту, определяется требуемым ему количеством. Такой метод называется *динамическим распределением буферной памяти*. После этого пакеты, находившиеся в буфере, динамически распределяются по выходным портам. Это позволяет получить пакет на одном порте и отправить его с другого порта, не устанавливая его в очередь.

Коммутатор поддерживает карту портов, в которые требуется отправить пакеты. Очистка этой карты происходит только после того, как пакет успешно отправлен.

Поскольку память буфера является общей, размер пакета ограничивается полным размером буфера, а не долей, предназначенной для конкретного порта. Это означает, что крупные пакеты, могут быть переданы с меньшими потерями, что особенно важно при асимметричной коммутации, т. е. когда порт с шириной полосы пропускания 100 Мб/с должен отправлять пакеты на порт 10 Мб/с. Чем больше объем памяти, тем лучше коммутатор справляется с перегрузкой. Но с ростом объема памяти повышается и стоимость оборудования. Растет и требование к быстродействию коммутатора. Иногда в состав коммутатора включается и универсальный процессор, но чаще коммутаторы выполняются на специализированных быстродействующих микросхемах, жестко ориентированных именно на задачи коммутации пакетов данных.

Коммутаторы подразделяются на *управляемые* и *неуправляемые* (наиболее простые). Соответствующие им фото 2 и фото 1 приведены на предыдущей странице. Более сложные коммутаторы позволяют управлять коммутацией на канальном (втором) и сетевом (третьем) уровне модели OSI. Обычно их именуют соответственно Layer 2 Switch или сокращенно L2 и Layer 3 Switch или сокращенно L3. Управление коммутатором может осуществляться посредством протокола Web-интерфейса или ряда других, специально разработанных производителями сетевого оборудования.

Коммутаторы выпускаются на различное число портов. Чаще всего встречаются устройства с 6, 8, 12, 16 и 24 портами. Различаются коммутаторы и допустимым количеством адресов на один порт. Этот показатель определяет предельную сложность подключаемых к порту сегментов (количество компьютеров в каждом сегменте). Некоторые коммутаторы позволяют разбивать порты на группы, работающие независимо друг от друга, т.е. один коммутатор может работать как два или три.

Так же, как и концентраторы, коммутаторы выпускаются трех видов в зависимости от сложности, возможности наращивания количества портов и стоимости:

* коммутаторы с фиксированным числом портов (обычно до 30);
* *модульные* коммутаторы с числом портов до 100 (фото слева);
* *стековые* коммутаторы (последнее фото).

Коммутаторы характеризуются двумя показателями производительности. Первый оценивает максимальную скорость ретрансляции пакетов при передаче пакетов из одного порта в другой, когда все остальные порты отключены. Второй оценивает совокупную скорость ретрансляции пакетов при активной работе всех имеющихся портов. Совокупная скорость больше максимальной, но максимальная скорость, как правило, не может быть обеспечена на всех портах одновременно, хотя коммутаторы и способны одновременно обрабатывать несколько пакетов.

При разбиении сети на сегменты с помощью коммутатора следует придерживаться «правила 80/20». Только при его выполнении коммутатор работает эффективно. Согласно этому правилу, необходимо, чтобы не менее 80% всех передач происходило в пределах одного сегмента сети. И только 20% всех передач должно происходить между разными сегментами сети и проходить через коммутатор. На практике это обычно сводится к тому, чтобы сервер и активно работающие с ним рабочие станции (клиенты) располагались в одном сегменте.

Для изучения принципов функционирования коммутаторов локальных вычислительных сетей предлагается создать проект виртуальной сети, основываясь на приведенной ниже схеме, и, расположив элементы проектируемой сети, структурировать ее на основе двух коммутаторов и одного концентратора, а также произвести конфигурирование IP-адресов рабочих станций средствами CNS [6, 7].



Лабораторная работа №4

Статическая маршрутизация в локальных вычислительных сетях

Цель работы:

1. Изучение назначения и принципов функционирования

маршрутизаторов.

2. Изучение принципов статической маршрутизации в локальных

вычислительных сетях с помощью программы Computer Network

Simulator (CNS).

Порядок выполнения работы

1. Произвести размещение шести узлов удаленных рабочих станций (PC1 - PC6), двух устройств маршрутизаторов и двух устройств коммутаторов средствами CNS в соответствии со схемой, приведенной выше. Объединить удаленные узлы и соответствующие порты маршрутизаторов и коммутаторов отрезками кабеля.

2. Рабочей станции РС1 и соответствующему порту маршрутизатора 1 (eth0) назначить IP-адреса из диапазона 91.122.40.4/30 (команда ifconfig).

3. Для назначения IP-адресов удаленным рабочим станциям РС2 и РС3, а также соответствующему порту маршрутизатора 1 (eth1) использовать адреса из диапазона 1 (см. таблицу заданий лабораторной работы №2, вариант соответствует двум последним цифрам студенческого билета).

4. Рабочей станции РС4 и соответствующему порту маршрутизатора 2 (eth1) назначить IP-адреса из диапазона 91.122.40.8/30.

5. Для назначения IP-адресов удаленным рабочим станциям РС5 и РС6, а также соответствующему порту маршрутизатора 2 (eth0) использовать адреса из диапазона 2 (см. таблицу заданий лабораторной работы №2, вариант соответствует двум последним цифрам студенческого билета).

6. Интерфейсу eth2 маршрутизатора 1 и интерфейсу eth2 маршрутизатора 2 назначить IP-адреса из диапазона 91.122.40.0/30.

7. Запустив эмулятор терминала, с помощью команды route установить правила статической маршрутизации для всех непосредственно подключенных и удаленных сетей на обоих маршрутизаторах.

8. Вновь, используя команду route эмулятора терминала, настроить правила маршрутизации на узлах рабочих станций РС1 – РС6. При этом узлам РС2 и РС3 должны быть доступны узлы РС5 и РС6, а узлу РС1 – узел РС4.

9. С помощью утилиты командной строки ping последовательно проверить доступность удаленных узлов с рабочих станций РС1, РС2, РС4 и РС5. Проследить при этом направление рассылки кадров в сети. Отметить узел отправителя и узел получателя в каждом случае, а также все узлы, участвующие в рассылке кадра.

10. Запустить эмулятор терминала на каждом из устройств маршрутизаторов и с помощью команды route просмотреть содержимое таблицы маршрутизации (IP routing table), которая должна иметь вид, подобный приведенному ниже:

**=>route**

**IP routing table**

**Destination Gateway Netmask Flags Metric Iface**

**10.0.0.0 \* 255.0.0.0 U 1 eth0**

**11.0.0.0 11.0.0.1 255.0.0.0 UG 1 eth1**

**192.168.120.1 10.0.0.1 255.255.255.255 UGH 1 eth2**

Здесь в столбце назначений (Destination) указывается адрес сети, а в последующих столбцах - ее маска (Netmask), номер интерфейса (Iface) маршрутизатора и адрес шлюза (Gateway), через который осуществляется доступ к этой сети. Если маршрут не использует шлюз, вместо адреса шлюза выводится (\*). Столбец флагов (Flags) может иметь значения: U — маршрут активен, G — маршрут использует шлюз, H — назначением является узел. Метрика (Metric) в условных единицах оценивает степень близости между двумя объектами сети.

11. С помощью пункта «Экспортировать в html…» меню «Проект» сформировать электронный отчет проекта в формате html, предварительно сохранив xml-проект CNS. Файл html-проекта следует распечатать и прикрепить к основному отчету по лабораторной работе.

12. На отчете проекта выделить границы сетей и широковещательных доменов. Перечислить используемые IP-адреса подсетей.

13. Сформулировать выводы по результатам работы.

Краткие теоретические сведения

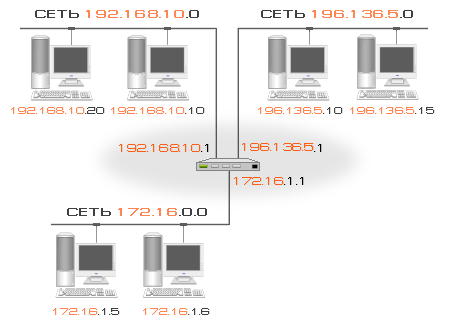
Маршрутизация — процесс определения маршрута следования информации в сетях связи. Маршруты могут задаваться непосредственно сетевым администратором (статическая маршрутизация), либо вычисляться с помощью специальных алгоритмов, базируясь на информации о топологии и состоянии сети, полученной с помощью протоколов маршрутизации (динамическая маршрутизация).

Статическими маршрутами могут быть:

* маршруты, не изменяющиеся во времени;
* маршруты, изменяющиеся по расписанию;
* маршруты, изменяющиеся по ситуации — административно в момент возникновения стандартной ситуации.

Маршрутизация пакетов данных включает в себя две основные задачи: определение оптимального маршрута между источником и приемником информации и передачу информации по сети. По-существу, процесс маршрутизации состоит в определении следующего узла в пути следования пакета и пересылки пакета этому узлу. Такой узел называют хопом (hop – прыжок, скачок).

Маршрутизация в компьютерных сетях выполняется специальными программно-аппаратными средствами - маршрутизаторами (router). В отличие от коммутаторов эти устройства работают не с физическими адресами пакетов данных, а с логическими сетевыми адресами (IP-адресами). При этом ретрансляции подлежат только те пакеты, которые адресованы в ту или иную подсеть, к которой данный маршрутизатор подключен. Кроме того, маршрутизаторы позволяют строить сети, имеющие петли, т.е. более одного пути возможного следования пакета от одного узла сети к другому. Несколько путей позволяют повышать пропускную способность сети, а также служат резервными каналами передачи данных на случай выхода из строя основных. При этом маршрутизатор отвечает за выбор маршрута.

Маршрутизаторы сложнее коммутаторов и, следовательно, дороже (например, стоимость коммутации в Ethernet примерно в 10 раз ниже стоимости маршрутизации). Маршрутизаторами сложнее управлять, они почти всегда значительно медленнее коммутаторов. Однако они обеспечивают самое глубокое разделение сети на части.

Если концентраторы всего лишь повторяют все поступившие на них пакеты (1-й, физический уровень модели OSI), а коммутаторы ретранслируют только межсегментные и широковещательные пакеты (2-й, канальный уровень модели OSI), то маршрутизаторы соединяют практически самостоятельные, не зависящие друг от друга сети, сохраняя при этом возможность передачи информации между ними (3-й, сетевой уровень модели OSI).

Маршрутизация подразумевает два параллельных процесса: подготовку маршрутной таблицы и переадресацию пакетов данных с помощью этой таблицы, которая содержит:

* номера всех сетей, подключенных к данному маршрутизатору;
* список всех соседних маршрутизаторов;
* список МАС-адресов и IP-адресов всех абонентов сетей, подключенных к маршрутизатору.

Следует отметить, что список всех доступных маршрутизаторов должен быть у каждого абонента сети. В случае статической маршрутизации таблица маршрутов заполняется вручную администратором сети и должна своевременно корректироваться в связи с любыми изменениями в топологии сети.

Основные функции, выполняемые маршрутизатором на сетевом уровне модели OSI: создание и ведение таблицы маршрутизации; определение маршрута по таблице маршрутизации; анализ информации из заголовка сетевого уровня пакета и изменение этого заголовка при необходимости (время жизни пакета и т.п.); фильтрация пакетов; проверка контрольной суммы пакетов и отбрасывание пакетов, содержащих ошибки; буферизация пакетов и управление очередями пакетов. На канальном уровне модели OSI: инкапсуляция пакетов сетевого уровня в кадры канального уровня при передаче пакетов, обратный процесс при их приеме и обработке; преобразование адреса следующего маршрутизатора или узла назначения из сетевого в физический. На физическом уровне модели OSI: обеспечение интерфейса со средой передачи данных; прием и передача кадров.

Маршрутизаторы применяются для объединения нескольких локальных сетей в единую составную сеть либо же, наоборот, для разграничения большой сети на несколько независимых малых подсетей.

Маршрутизаторы применяются для объединения разнородных сетей как локальных, так и глобальных. Например, для соединения локальной сети с сетью Интернет, или для объединения локальных сетей, работающих на различных скоростях передачи данных, например Ethernet и Fast Ethernet.

Маршрутизаторы позволяют сделать невидимыми сетевые адреса узлов локальной сети из внешней сети, подменяя их своим адресом. Такой прием используется как мера безопасности, усложняющая несанкционированное проникновение в «скрытую» с помощью маршрутизатора сеть, и как мера, позволяющая увеличить адресное пространство, поскольку адреса внутри локальных сетей, можно сказать, не существуют для внешней глобальной сети и могут совпадать. Такое сокрытие называется трансляцией сетевых адресов.

В случае вырожденной сети, когда из огромного числа узлов задействуется всего два, а также при организации подсетей с малым числом рабочих станций целесообразно разделение большой сети на части с помощью масок. Например, маска из 30 единиц (в десятичной нотации 255.255.255.252) сокращает доступное для нумерации в сети класса С число узлов на 6 бит, а число возможных подсетей, наоборот, увеличивает на эти же 6 бит. В примерах записи IP-адресов они выделены жирным шрифтом:

91.122.40.0/30 (01011011.01111010.00101000.**000000**00)

91.122.40.4/30 (01011011.01111010.00101000.**000001**00)

91.122.40.8/30 (01011011.01111010.00101000.**000010**00)

В зависимости от областей применения и функциональной сложности маршрутизаторы могут быть как программными, так и аппаратными. При этом они делятся на несколько классов.

Первый класс – магистральные маршрутизаторы, применяемые для сетей операторов связи и провайдеров сетевых услуг. Для таких маршрутизаторов характерны очень высокий уровень производительности, а также наличие мощных средств обеспечения отказоустойчивости, причем как отдельного узла, так и всей сети в целом, использование сверхскоростных интерфейсов (10 – 40 Гбит/с).

 Второй класс – маршрутизаторы корпоративных сетей, используемые как в этих сетях, так и для подключения к каналам операторов связи. Характеристики данного класса гораздо скромнее, однако функциональные возможности оказываются зачастую более широкими за счет, например, реализации поддержки разнообразных телефонных стандартов связи.

Третий класс – маршрутизаторы локальных сетей, ориентированные на использование в небольших компаниях или в небольших домашних сетях пользователей (см. фото 1). Такие маршрутизаторы достаточно просты в настройке с помощью Web-интерфейса и являются сравнительно недорогими. Кроме того, маршрутизаторы этого класса часто снабжаются дополнительными возможностями, такими как встроенные серверы печати или точки беспроводного доступа.

На своем, третьем уровне модели OSI маршрутизатор часто выполняет функции сетевого шлюза, т.е. обеспечивает «шлюзование» передаваемой информации между сетями, использующими различные протоколы. Если при этом маршрутизатор защищает сеть от любых нежелательных воздействий, способных нанести вред (ущерб) пользователям сети или хранящейся в ней информации, он приобретает также функции пакетного фильтра – брандмауэра (brandmauer) или файервола (firewall). Обычно брандмауэр устанавливается на узлы, в т.ч. маршрутизаторы, соединяющие локальную сеть с внешней сетью, например с Интернетом.

Для изучения принципов функционирования маршрутизаторов локальных вычислительных сетей предлагается создать проект виртуальных сетей, основываясь на приведенной ниже схеме, и, расположив элементы проектируемых сетей, структурировать их на основе двух маршрутизаторов и двух коммутаторов, а также произвести конфигурирование IP-адресов рабочих станций средствами CNS [6, 7].



Список литературы

1. Угрюмов Е.П. Цифровая схемотехника. - СПб.: БХВ – Санкт-Петербург, 2007.

2. ВАРИАНТ. Система логического моделирования цифровых микросхем и схем на их основе. Руководство пользователя.- Ростов н/Д: РВВКИУРВ, 1993.

3. Кельнер А., Терешин М. <http://www.net-simulator.org/ru/index.shtml> Ресурс WWW, 2011.

4. Олифер В. Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы: Учебник для Вузов. 4-е издание. СПб.: Питер, 2014.

5. Крейг Х. TCP/IP. Сетевое администрирование, 3-е издание. СПб.: Символ-Плюс, 2004.

6. Небаев И.А. Компьютерные сети передачи данных: Учебное пособие. СПб.: СПбГУТ, 2013.

7. Болдырев А.В. Среда эмуляции вычислительных сетей Computer Network Simulator: Метод. указания. – Ростов н/Д: Издательский центр ДГТУ, 2015.