

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**

**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**(ДГТУ)**

Факультет Информатика и вычислительная техника

Кафедра Кибербезопасность информационных систем

**ОТЧЕТ ПО ЛАБОРАТОРНЫМ РАБОТАМ**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Автор |  | | |  | Ковалев Д. П. | | | |
|  | (подпись, дата) | | |  |  | | | |
| Обозначение | ИиВТ.10.05.01.13 | | Группа | | | | ВКБ32 |  |
| Направление подготовки | | 10.05.01 Компьютерная безопасность | | | | | |  |
| Профиль | Математические методы защиты информации | | | | | | |  |
| Проверил |  | | |  | |  | |  |
|  | (подпись, дата) | | |  | |  | |  |

Ростов-на-Дону

2024

Содержание

[Лабораторная работа №1 3](#_Toc159059439)

[Лабораторная работа №2 4](#_Toc159059440)

[Лабораторная работа №3 8](#_Toc159059441)

[Лабораторная работа №4 10](#_Toc159059442)

[Лабораторная работа №5 12](#_Toc159059443)

[Лабораторная работа №6 14](#_Toc159059444)

[Лабораторная работа №7 16](#_Toc159059445)

# 17

# **Лабораторная работа №1**

**Тема:** «Изучения создания сети в Cisco Packet Tracer»

**Цель:** научиться создавать простые сетевые соединения.

**Задание 1**. Создание сети из двух ПК в программе Cisco Parket Tracer.

Нажмем на кнопку “End devices”, где находятся персональные компьютеры. На рисунке 1 также видно знак молнии, чтобы подключить наши 2 компьютера. Во вкладке “Connections” есть медные провода с обжимом.

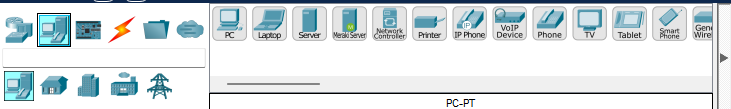


Рисунок 1 – интерфейс программы с Drag & Drop

В результате проделанных операций получилось подключение, которое представлено на рисунке 2.

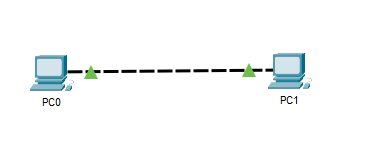


Рисунок 2 – физическое подключение 2 ПК

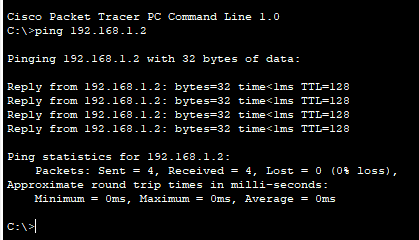
 Далее нам нужно настроить IP адрес и маску для каждого ПК, как было показано в методическом материале. Теперь можно приступить к проверке видимости другого ПК в локальной сети с помощью команды “ping 192.168.1.2”. На рисунке 3 представлен результат.

Рисунок 3 – результат команды ping

**Задание 2.** Организация Режим симуляции работы сети.

Нужно задать в рабочем пространстве программы сеть из 4х ПК и 2х хабов. Результат подключения представлен на рисунке 4.

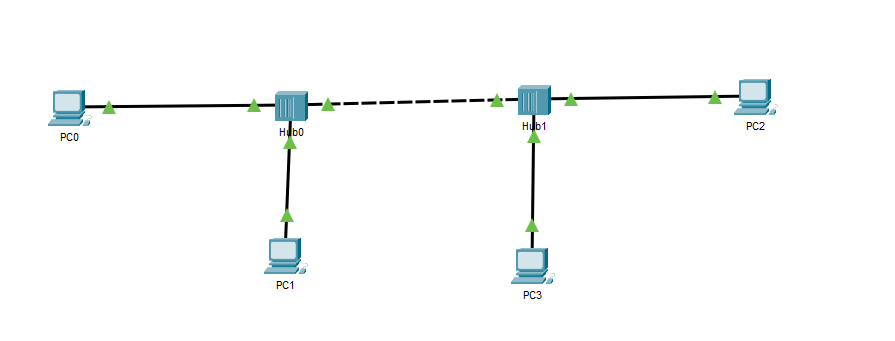


Рисунок 4 – результат подключения устройств

Далее я отключил все протоколы, оставив только ICMP (Internet Control Message Protocol) — сетевой протокол, входящий в стек протоколов TCP/IP. В основном ICMP используется для передачи сообщений об ошибках и других исключительных ситуациях, возникших при передаче данных. Фильтры были отключены, как представлено на рисунке 5.

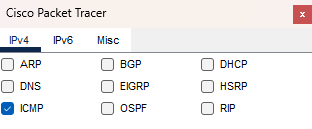


Рисунок 5 – фильтры, которые я использовал

Теперь создадим пакет ICMP, который будет передаваться от 1 ПК к другому, здесь вот создал я пакет, который будет в дальнейшем отправлен, данный конверт представлен на рисунке 6.

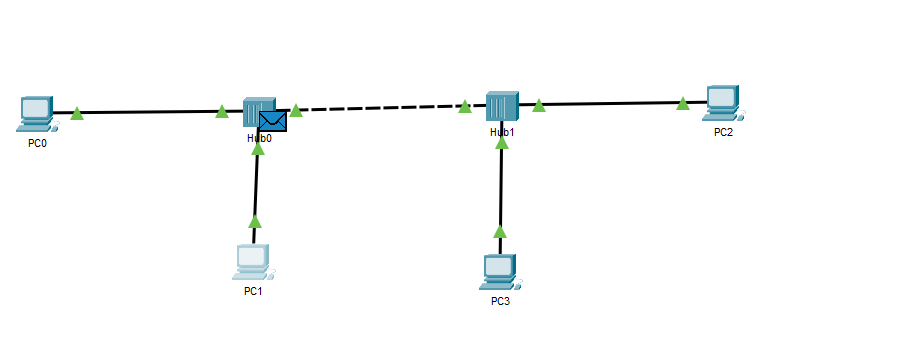


Рисунок 6 – процесс отправки пакетов с 1 ПК на другой

Чтобы отображалось в списке событий, нужно нажать на английскую букву “P” или нажать на иконку “письмо”. Дальше остается только связать два элемента. В результате должно появиться, как на рисунке 7.



Рисунок 7 – список событий после прогона пакета

**Задание 3.** Настройка сетевых параметров ПК в его графическом интерфейсе.

Настроить IP адрес и маску подсети можно, используя несколько вариантов. Первый вариант – использование GUI интерфейса, другой вариант – это использование командной строки. В нашем случае подойдет команда `ipconfig 192.168.0.1 255.255.255.0`, где первый аргумент – это IP, а второй 0 маска. Также с помощью данной команды можно проверять настройки. Ipconfig для 4 ПК представлен на рисунке 8.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 8 – ipconfig для 4 ПК

# **Лабораторная работа №2**

**Тема:** «Изучение построения сетей с различными топологиями»

**Цель:** научиться строить сеть с топологией звезда.

**Задание 1**. Моделирование сети с топологией звезда на базе концентратора.

Для начала нам нужно понять, что такое топология “Звезда”? На самом деле – это когда много объектов подключаются к центральному звену, в качестве центра, грубо говоря, может быть хаб, коммутатор или ПК. Сама топология представлена на рисунке 1.

Изображение выглядит как снимок экрана, текст, линия, диаграмма

Автоматически созданное описание

Рисунок 1 – топология звезда

Отправка пакетов в GUI делается с помощью “конвертика” в верхней панели, фото данного участка представлено на рисунке 2.



Рисунок 2 – фото раздела

После тот, как мы нажали на конверт, то мы теперь можем выбрать от какого ПК отправить к какому. В результате проделанной операции мы получим статус продвижения пакета. Все показано на рисунке 3.



Рисунок 3 – отчет о прохождении пакета

Далее требуется построить ещё одну сеть с целью изучения полезных приемов в CPT. В Cisco Packet Tracer можно копировать объекты, зажимая Ctrl и выбирая определенные объекты через ЛКМ. Новая сеть представлена на рисунке 4.

Изображение выглядит как диаграмма, снимок экрана, линия

Автоматически созданное описание

Рисунок 4 – сеть с использованием хаба и свитча

**Задание 2**. Моделирование сети с топологией звезда на базе коммутатора.

Нам нужно построить сеть на основе свитча. Здесь можно заметить, что свитч отправляет пакеты только определенному потребителю в прямом и обратном направлении. Сама сеть представлена на рисунке 5.

Изображение выглядит как снимок экрана, диаграмма

Автоматически созданное описание

Рисунок 5 – топология звезда с использованием свитча

Теперь по заданию нужно построить с использованием хаба и свитча. Сеть представлена на рисунке 6.

Изображение выглядит как диаграмма, снимок экрана

Автоматически созданное описание

Рисунок 6 – сеть с использованием свитчи и хаба

Тестировать подключение будет с использованием команды ping, делая запрос от к PC1 к PC3. Результат запроса представлен на рисунке 7. Там видно, что происходит 4 запроса, в среднем задержка 2 ms, что неплохо я считаю.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 7 – результат запроса ping от PC0 к PC3

**Задание 3.** Исследование качества передачи трафика по сети

Теперь мы здесь будем тестировать различные сети, построенные на хабе и коммутаторе по задержкам. В начале я построю сеть, где в центральной части будет хаб, все представлено на рисунке 8.

Изображение выглядит как диаграмма, линия, снимок экрана

Автоматически созданное описание

Рисунок 8 – строение сети, где центральный элемент – хаб

Я отправил в traffic generator PC1, а начал отправлять пакеты с 0 ПК на PC3. Потеряно где-то 6 пакетов. Я сделал поменьше запросов с той целью, чтобы долго не отчитывать. Количество потерянных пакетов представлено на рисунке 9.

Изображение выглядит как текст, снимок экрана, меню

Автоматически созданное описание

Рисунок 9 – потеря пакетов с хабом.

Теперь попробуем поменять так, чтобы у нас был свитч, а не хаб. В результате потеря пакетов составляет в количестве 4 шт., что получше намного, чем с хабом. Результат представлен на рисунке 10.

Изображение выглядит как снимок экрана, текст, Шрифт, линия

Автоматически созданное описание

Рисунок 10 – потеря пакетов со свитчем

# **Лабораторная работа №3**

**Тема:** «Знакомство с командами Cisco IOS»

**Цель:** научиться использовать командную строку Cisco Packet Tracer

**Задание 1.** Изучить работу CLI.

Столкнулся с проблемой, что нельзя выйти из настройки интерфейсов, используя команду Interface. Помогает только закрытие файла или простое ожидание. Все представлено на рисунке 1.



Рисунок 1 – попытка настроить интерфейсы

Установку пароля делал, как представлено на рисунке 2. Здесь главное в начале войти в config через команду `conf t`, а потом уже вводить настройки для установки.

Изображение выглядит как текст, снимок экрана, линия, Шрифт

Автоматически созданное описание

Рисунок 2 – установка пароля на роутер

Теперь я проверил, что для входа в привилегированный режим нужно использовать пароль, все работает. Для выхода из данного режима нужно ввести `logout`. Результат проверки находится на рисунке 3.

Изображение выглядит как текст, Шрифт, снимок экрана, дизайн

Автоматически созданное описание

Рисунок 3 – проверка входа в привилегированный режим

C помощью secret поменяли пароль для входа пользователя в привилегированный режим. Для сохранения всех наших операций можем воспользоваться сохранением пароля в память, используя `write memory`.

# **Лабораторная работа №4**

**Тема:** «Работа с сервером в Cisco Packet Tracer»

**Цель:** научиться строить Web серверы

**Задание 1**. Настраиваем WEB сервер.

Для начала проведем настройку сети, строение сети показано на рисунке 1. Здесь все было сделано по методичке. У сервера обязательно нужно включить HTTP во вкладке Services для работы.

Изображение выглядит как снимок экрана, диаграмма, линия

Автоматически созданное описание

Рисунок 1 – строение сети

Теперь напишем нашу HTML страницу для отображения, назвав обязательно “index.html”, теперь с ноутбука запустим Web Browser, чтобы проверить работоспособность. На рисунке 2 представлен результат отображения index.html.

Изображение выглядит как текст, программное обеспечение, Шрифт, веб-страница

Автоматически созданное описание

Рисунок 2 – результат отображения html cтраницы

**Задание 2.** Настройка сетевых сервисов DNS, DHCP и Web

Для начала построить схему, которая представлена на рисунке 3. Здесь мы изначально подключили ПК, выставив параметр DHCP у каждого ПК.

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Рисунок 3 – сеть для 2 задания

Зададим статический адрес для серверов, приложу фото как я это сделал. Для всех PC мы зададим DHCP, чтобы был автоматический IP адрес. Теперь начнем работать с DNS в Server1. Результат представлен на рисунке 4.

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Рисунок 4 – связь доменного имени с IP адресом

В начале мы задаем имя домена - `www.yandex.ru`, а также задаем имя хоста, на который мы будем ссылаться `server1.yandex.ru`. А на хост мы должны дать ссылку на адрес `10.0.0.1` (физический адрес cервера).

На Server0 открываем консоль для проверки запроса к нашему домену. Результат представлен на рисунке 5. Команда nslookup служит для определения IP-адреса по доменному имени (и наоборот).

Изображение выглядит как текст, электроника, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 5 – результат обращения к домену

На Server0 мы выставили DHCP, как показано на рисунке 6. Напоминаю, что DHCP нужен для внутреннего распределения IP адресов, чтобы было понятно как с домена идти на хост.

Изображение выглядит как текст, Шрифт, число, линия

Автоматически созданное описание

Рисунок 6 – включение протокола DHCP для Server0

Теперь для PC0 и PC1 сбросим параметры сети, и попросим выдать от сервера данные, чтобы могли взаимодействовать по локальной сети. Теперь через веб браузер мы можем попробовать обратиться к домену, чтобы получить HTML страницу.

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рисунок 7 – результат запроса к серверу

**Задание 3.** Конфигурирование DHCP сервера на маршрутизаторе.

Построим сеть, как сказано в условии. Результат построения представлен на рисунке 8.

Изображение выглядит как снимок экрана, диаграмма

Автоматически созданное описание

Рисунок 8 – сеть для 2 задания

Зайдем в командную консоль роутера, где мы будет работать по заданию. Для начала создадим пул IP адресов для DHCP cервера с именем TST. Команда, которую я использовал, выглядит так: `ip dhcp pool TST`, далее я оставляю ссылку – IP адрес роутера. Теперь укажем адрес основного шлюза, который будет рассылать в сообщениях DHCP. Теперь мне нужно указать адрес DNS сервера, который находится `dns-server 5.5.5.5`. В конце нам нужно исключить из пула.

Изображение выглядит как текст, снимок экрана, Шрифт, документ

Автоматически созданное описание

Рисунок 9 – команды для настройки роутера

Проверим DHCP с помощью команды `ipconfig /all`.

**Задание 4**. Пример настройки интерфейса маршрутизатора в качестве DHCP клиента.

Для начала построим сеть, как представлено на рисунке 10. После этого мы должны перейти в режим продвинутого пользователя через `enable`, теперь перейдем в настройку в `int fa0/0`, `ip address dhcp`. После этого на PC1 не будет работать DHCP, как представлено на рисунке 11.

После настройки интерфейса роутера на получение настроек по DHCP, DHCP клиент на PC1 перестал получать IP-адрес – IP из диапазона 169.254.x.x/16 назначается автоматически самим ПК при проблемах с

получением адреса по DHCP. Интерфейс роутера IP-адрес так же не получит т.к. в данной подсети нет DHCP серверов.

Изображение выглядит как снимок экрана, дизайн

Автоматически созданное описание

Рисунок 10 – сеть для задания

Видим, что тут на рисунке 11 не пустые поля, что свидетельствует о запасном IP в случае того, если DHCP не заработает.

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Рисунок 12 – настройка PC1

**Задание 5**. DHCP сервис на маршрутизаторе 2811.

Для начала создадим нашу сеть, используя маршрутизатор 2811, сама сеть представлена на рисунке 13. Наша задача настроить так, чтобы ПК подключались с помощью DHCP, для этого в Config мы должны задать настройки для подключения.

Изображение выглядит как линия

Автоматически созданное описание

Рисунок 13 – сеть для задания 5

Далее мы зайдем в роутер, чтобы выставить все возможные IP адреса для настроек. Для начала зайдем в консоль, перейдем в режим продвинутого, вбив команду `enable`. После этого мы можем перейти в настройки, вбив команду `conf t`. Теперь у нас есть возможность зарезервировать адреса, используя команду `ip dhcp excluded-address 192.168.1.1 192.168.1.10`. Таким образом, данные адреса будут только для хостов, DHCP их выдавать устройствам не будет.

Теперь наша задача – создать пул адресов, которые будут выдаваться из нашей сети для подключение внешних устройств по типу ПК.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 14 – настройки для сети

Теперь настроим интерфейс для маршрутизатора, как сказано в задании. Все представлено на рисунке 15.

Изображение выглядит как текст, снимок экрана, Шрифт, алгебра

Автоматически созданное описание

Рисунок 15 – настройка роутера

Теперь может получить список выданных роутером адресов. Надо посмотреть на список выданных роутером адресов. Данный список представлен на рисунке 16.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 16 – IP адресы, выданные роутером

Итак, мы видим, что протокол DHCP позволяет производить автоматическую настройку сети на всех компьютерах.

# **Лабораторная работа №5**

**Тема:** «Статическая маршрутизация»

**Цель:** научиться работать со статической и динамической маршрутизацией внутри сети

**Задание 1**. Настраиваем связь двух сетей через маршрутизатор

Наша цель – настроить связь двух сетей через маршрутизатор (роутер). Тандем сетей представлен на рисунке 1. Для начала мы должны настроить ПК, связанные свитчем, чтобы они находились в одной подсети.

Изображение выглядит как диаграмма, линия, снимок экрана, График

Автоматически созданное описание

Рисунок 1 – сеть для 1 задания

Наша задача также настроить роутер так, чтобы он выступал в роли шлюза для первой сети на интерфейсе Fa0/0. Настройка роутера представлена на рисунке 2.

Изображение выглядит как текст, чек, Шрифт, алгебра

Автоматически созданное описание

Рисунок 2 – настройка роутера

Теперь настраиваем роутер так, чтобы он со второго порта принимал запросы от второй подсети, настройка в принципе ничем не отличается за исключением того, что нужно прописать адрес другой для шлюза.

Остается проверить только результат настройки, использовав команду `show ip route`. Результат команды представлен на рисунке 3.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 3 – результат запроса по IP

Проверим теперь, что проходит сигнал на разные подсети, использовав команду ping, результат представлен на рисунке 4.

Изображение выглядит как текст, снимок экрана, документ, Шрифт

Автоматически созданное описание

Рисунок 4 – пинг для двух сетей

Теперь проверим работоспособность сети, сделав запрос к каждому отдельному ПК.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 5 – проверка работы первой подсети

Аналогичные действия сделаем и со второй подсетью.

**Задание 2.** Настройка трех сетей с WEB сервером. Понятие маршрута по умолчанию.

Схема у нас будет следующая: два коммутатора 2950–25, два ПК в сети 192.168.10.0 с маской 255.255.255.0. Сервер и компьютер в сети 192.168.20.0 с маской 255.255.255.0. Сеть между маршрутизаторами (марки 1851) 192.168.1.0 с маской 255.255.255.252. Компьютеры из сети 192.168.10.0 должны достучаться к DNS серверу в сети 192.168.20.0. Сеть представлена на рисунке 6.

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Рисунок 6 – сеть для 2 задания

Мы будем делать статическую маршрутизацию, поэтому в начале мы настроим роутеры для корректной работы. Зайдем в терминал у Router0, теперь пишем команды, которые представлены на рисунке 7.

Изображение выглядит как текст, Шрифт, снимок экрана, белый

Автоматически созданное описание

Рисунок 7 – настройка первого роутера

То же самое проделаем и для Router1, правда мы воспользуемся в этот раз уже GUI для разнообразия. Результат представлен на рисунке 8.

Изображение выглядит как текст, снимок экрана, дисплей, число

Автоматически созданное описание

Рисунок 8 – настройка Router 1

Теперь настроим fa0/0 у Router 0, чтобы наш роутер стал шлюзом. Настройка представлена на рисунке 9.

Изображение выглядит как текст, снимок экрана, дисплей, число

Автоматически созданное описание

Рисунок 9 – настройка порта Router0 в качестве шлюза

Аналогично настроим порт fa0/1 у Router 1, чтобы он стал шлюзом. Результат представлен на рисунке 10.

Изображение выглядит как текст, снимок экрана, число, дисплей

Автоматически созданное описание

Рисунок 10 – настройка порта Router1 в качестве шлюза

Теперь перейдем к настройке всех ПК в сети, нужно задать им IP адреса, маску подсети и основной шлюз. Настройка шлюза представлена на рисунке 11, все параметры можно выставить в Config с разделом Settings.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 11 – настройка шлюза для PC1

Представлю ниже на рисунке 12 параметры – IP Adress, Subnet Mask.

Изображение выглядит как текст, линия, Шрифт, число

Автоматически созданное описание

Рисунок 12 – настройка IP и маски подсети у PC1

Теперь перейдем к настройке сервера и PC3 для дальнейшей работы. Настройки сервера прилагаю ниже на рисунках 13, 14.

Изображение выглядит как текст, Шрифт, линия, снимок экрана

Автоматически созданное описание

Рисунок 13 – настройки fa0/0 для сервера

Изображение выглядит как текст, снимок экрана, дисплей, Шрифт

Автоматически созданное описание

Рисунок 14 – настройки DNS и Default Gateway

Теперь перейдем к настройке маршрутизации на маршрутизаторах. У нас подсеть не возвращает обратно пакеты, поэтому мы должны настроить роутеры. Для этого зайдем в CLI для настройки роутеров. По умолчанию маршрут для Router0 представлена на рисунке 15.

Изображение выглядит как текст, Шрифт, снимок экрана, алгебра

Автоматически созданное описание

Рисунок 15 – настройка Router0

Настройка для Router1 делается аналогичным образом за исключением того, что IP должен быть 192.168.1.1. Запись означает, что все запросы, для которых не прописаны маршруты, R2 отправляет на 192.168.1.1, то есть,

на R1. Теперь если мы пропингуем, то увидим результат, что не все пакеты возвращаются обратно. Результат команды ping представлен на рисунке 16.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 16 – запросы с помощью ping от PC1 к PC2

Чтобы убедиться наверняка, давайте посмотрим, как идут пакеты по узлам сети и для этого воспользуемся командой tracert 192.168.20.20.

**Задание 3**. Сеть на двух маршрутизаторах.

Теперь наша задача настроить сеть так, чтобы она поддерживала статическую маршрутизацию. Сеть для задания представлена на рисунке 17.

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Рисунок 17 – сеть для задания 3

Теперь для каждого роутера покажем, что можно пересылать данные в нужные подсети. Для Router 1 покажем, что он может попасть в подсеть “192.168.10.0” с маской “255.255.255.0” c помощью интерфейса “192.168.200.1”. Аналогичным образом сделаем и для Router0. Результат настройки для Router1 представлен на рисунке 18.



Рисунок 18 – настройка подключения для Router1

Теперь таблица для маршрутизации для Router1 выглядит следующим образом, как на рисунке 19.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 19 – таблица маршрутизации для Router1

Обязательно нужно добавить шлюз для сети – это IP адрес, на который отсылаются пакеты в случае того, если нужное устройство не найдено было в подсети.

**Задание 4.** Статическая маршрутизация для пяти сетей и роутеров с тремя портами.

Для начала создадим сеть для нашего задания. Она представлена на рисунке 20. На данной схеме имеется пять сетей: 192.168.1.0, 152.20.20.0, 192.168.100.0, 10.10.10.0 и 192.168.2.0. В качестве шлюза по умолчанию у каждого компьютера указан интерфейс маршрутизатора, к которому он подключен. Маска у всех ПК одна – 255.255.255.0. Маска маршрутизаторов для каждого порта своя: Fa0/0 – 255.255.255.0, Fa0/1 – 255.255.0.0, Fa1/0 – 255.255.255.252.

Изображение выглядит как текст, снимок экрана, диаграмма, линия

Автоматически созданное описание

Рисунок 20 – сеть для 4 задания

Теперь мы должны соединить маршрутизаторы между собой, нужно добавить интерфейсную плату NM-1FE-TX. Результат представлен на рисунке 21.

Изображение выглядит как машина, электроника, панель управления, Электронная техника

Автоматически созданное описание

Рисунок 21 – конфигурация Router0

Не все пакеты будут возвращаться обратно, поэтому мы должны настроить путь по умолчанию для возврата, для этого мы пропишем в конфигурации R0 следующие команды, которые представлены на рисунке 21.

Изображение выглядит как текст, Шрифт, снимок экрана, алгебра

Автоматически созданное описание

Рисунок 21 – путь по умолчанию для R0

# **Лабораторная работа №6**

**Тема:** «Динамическая маршрутизация на протоколах RIP и EIGRP»

**Цель:** научится строить сети с динамической маршрутизацией.

**Задание 1**. Настройка протокола RIP версии 2 для сети из шести устройств.

Построим сеть, которую попросят по заданию. Результат представлен на рисунке 1.

Изображение выглядит как текст, снимок экрана, диаграмма, линия

Автоматически созданное описание

Рисунок 1 – настройка сети для задания 1

Перейдем в настройки Router0, чтобы настроить протокол RIP. Последовательность шагов представлена на рисунке 2.

Изображение выглядит как текст, Шрифт, снимок экрана, алгебра

Автоматически созданное описание

Рисунок 2 – параметры для Router0

То же самое мы должны сделать и c Router1. Теперь нужно перейти в Config Router0 и Router1 для проверки наличия параметров. Результат для Router0 представлен на рисунке 3.

Изображение выглядит как текст, снимок экрана, дисплей, число

Автоматически созданное описание

Рисунок 3 – RIP для Router0

**Задание 2.** Конфигурирование протокола RIP версии 2 для сети из четырех устройств.

Для начала построим нашу сеть, которую просят сделать по заданию. Все представлено на рисунке 4.

Изображение выглядит как линия, снимок экрана, График

Автоматически созданное описание

Рисунок 4 – сеть для 2 задания

По условию задания даны IPv4 для 4 устройств и 2 шлюза. Сначала сконфигурируем R0, как требуется в задании. Настройки представлены на рисунке 5.

Изображение выглядит как текст, Шрифт, снимок экрана, алгебра

Автоматически созданное описание

Рисунок 5 – настройки R0

В Config, где находится RIP, добавились следующие поля, которые представлены на рисунке 6.

Изображение выглядит как текст, Шрифт, снимок экрана, белый

Автоматически созданное описание

Рисунок 6 – Config R0

Теперь перейдем к настройке R1, все команды представлены на рисунке 7.

Изображение выглядит как текст, Шрифт, снимок экрана, алгебра

Автоматически созданное описание

Рисунок 7 – настройки R1

Наблюдаем результат, представленный на рисунке 8.

Изображение выглядит как текст, Шрифт, снимок экрана, белый

Автоматически созданное описание

Рисунок 8 – Config R1

Теперь проверим доступность от одного ПК к другому, использовав команду ping. Результат отправки пакетов представлен на рисунке 9.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 9 – результат команды ping

**Задание 3**. Конфигурирование протокола EIGRP.

Протокол EIGRP более прост в реализации и менее требователен к вычислительным ресурсам маршрутизатора, чем протокол OSPF. Также EIGRP имеет более продвинутый алгоритм вычисления метрики. В формуле вычисления метрики есть возможность учитывать загруженность и надежность интерфейсов на пути пакета. Недостатком протокола EIGRP является его ограниченность в его использовании только на

оборудовании компании Cisco.

Построим теперь сеть для 3 задания. Она представлена на рисунке 10.

Изображение выглядит как текст, снимок экрана, диаграмма

Автоматически созданное описание

Рисунок 10 – сеть для 3 задания

Теперь нам нужно настроить параметры протокола для двух роутеров. На рисунке 11 представлена настройка для Router0.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 11 – настройки для Router0

Для проверки работоспособности сделаем ping c одного ПК на другой. Результат представлен на рисунке 12.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 12 – результат ping

**Задание 4**. Пример конфигурирования протокола OSPF для 4-х устройств. Дана прямая маска 266.266.266.246. Выполните расчет и докажите, что обратная равна 0.0.0.7.

Чтобы рассчитать обратную маску (wildcard mask) из данной прямой маски, нужно использовать следующее правило: каждый байт обратной маски равен 255 минус соответствующий байт прямой маски.

Однако, 266 — это недопустимое значение для байта (максимум — 255).

Получается у нас есть прямая маска 255.255.255.248 (или /29). Вычисляем обратную маску:

- 255 - 255 = 0

- 255 - 255 = 0

- 255 - 255 = 0

- 255 - 248 = 7

Таким образом, обратная маска будет 0.0.0.7. Теперь построим сеть, которую попросили в задании 4. Все представлено на рисунке 13.

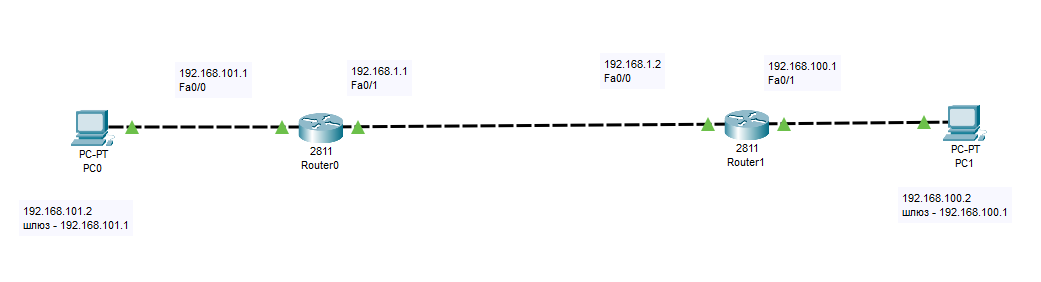


Рисунок 13 – сеть для задания 4

Теперь перейдем к настройке роутеров, чтобы все работало по протоколу OSPF. Для этого нужно выполнить операции, которые представлены ниже на рисунках 14 и 15. Рисунок 14 – это настройка Router0, а рисунок 15 – настройка Router1.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 14 – настройка Router0

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 15 – настройка для Router1

Теперь остается сделать проверку, что пакеты с одного ПК идут на другой ПК. Результат пинга представлен на рисунке 16.

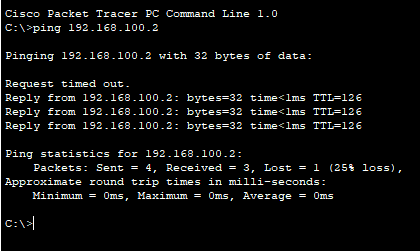


Рисунок 16 – отправка пакетов с PC0 на PC1

**Задание 5**. Настройка маршрутизации по протоколу OSPF для 6 устройств.

Построим для начала сеть для задания 5. Она представлена на рисунке 17.

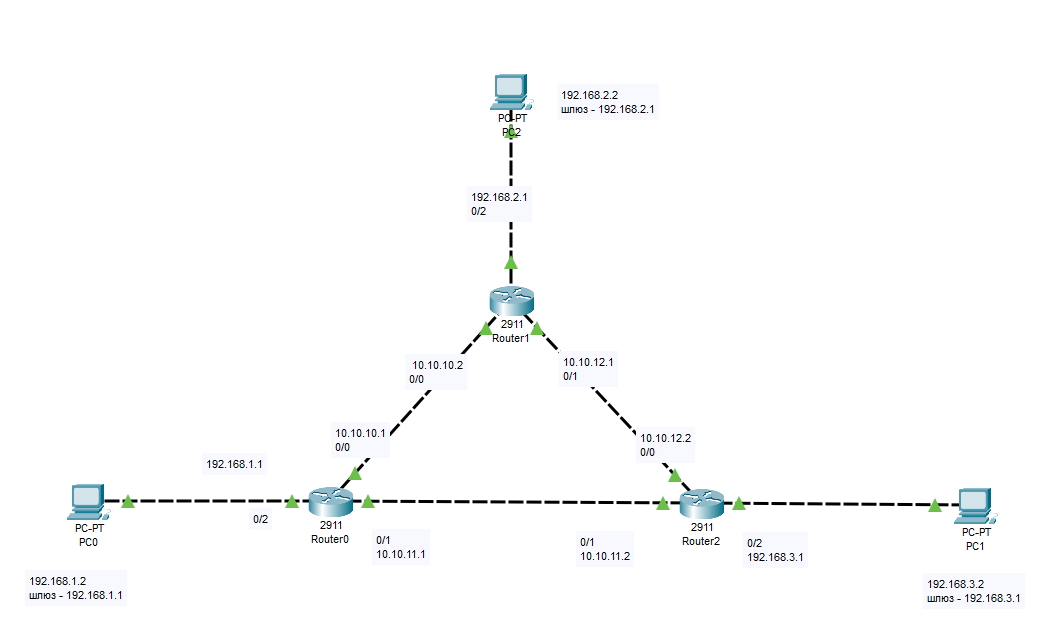


Рисунок 17 – сеть для задания 5

Теперь настроим на R0 настроим программный loopback интерфейс — алгоритм, который направляет полученный сигнал (или данные) обратно отправителю. Все команды представлены ниже на рисунке 18.

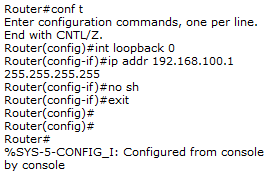


Рисунок 18 – настройка loopback на R0

Можем перейти к настройке OSPF на R0. Включаем OSPF на R0, все маршрутизаторы должны быть в одной зоне area 0. Настройка данного протокола представлена на рисунке 19.

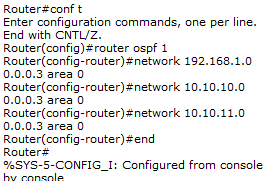


Рисунок 19 – настройка OSPF на Router0

Теперь можем перейти к настройке Router1. Все настройки представлены на рисунке 20.

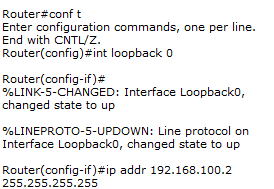


Рисунок 20 – настройка loopback на R1

Настраиваем OSPF на R1: включаем протокол OSPF на R1, все маршрутизаторы должны быть в одной зоне area 0.

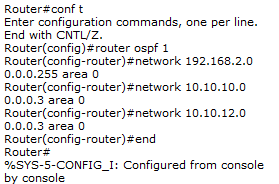


Рисунок 21 – настройка OSPF на R1

Настраиваем loopback интерфейс на R2. Аналогично все делаем, как у прошлых роутеров. Команды представлены на рисунке 22.

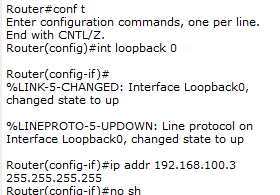


Рисунок 22 – настройка loopback на R2

Убедимся, что Router2 видит Router1 и Router0. Заходим в CLI у Router2. Переходим в режим администратора с помощью команды “en”, а дальше набираем “sh ip ospf neighbor”. Результат представлен на рисунке 23.



Рисунок 23 – результат команды

Теперь выведу таблицу маршрутизации, её можно получить, использовав команду “sh ip route”. Результат представлен на рисунке 24.

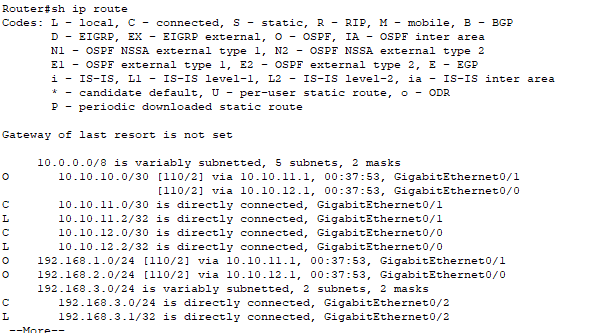


Рисунок 24 – таблица маршрутизации

В этой таблице запись с буквой "О" говорит о том, что данный маршрут прописан протоколом OSPF. Мы видим, что сеть 192.166.1.0 доступна для R2 через адрес 10.10.11.1 (это порт gig0/1 маршрутизатора R0). Аналогично, сеть 192.166.2.0 доступна для R2 через адрес 10.10.12.1 (это порт gig0/1 маршрутизатора R1).

Теперь проверим доступность разных сетей. Для этого зайдем в CLI у Router2 и выполним ping 192.168.1.2, а ещё ping 192.168.2.2. Результат представлен на рисунке 25.

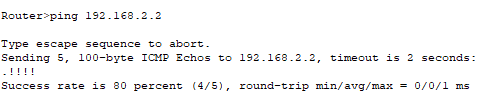


Рисунок 25 – результат запроса на IP 192.168.2.2

# **Лабораторная работа №7**

**Тема:** «Самостоятельное кодирование платы»

**Цель:** Запрограммировать плату так, чтобы начальная температура была 40 градусов (при этом не горит ни один светодиод), потом при поднятии температуры на каждые 10 градусов зажигался один светодиод.  
(50 град.- один, 60 град.- два, 70 град.- все три светодиода).

Используя термодатчик и три светодиода на рисунке 3 показан способ реализации задания.

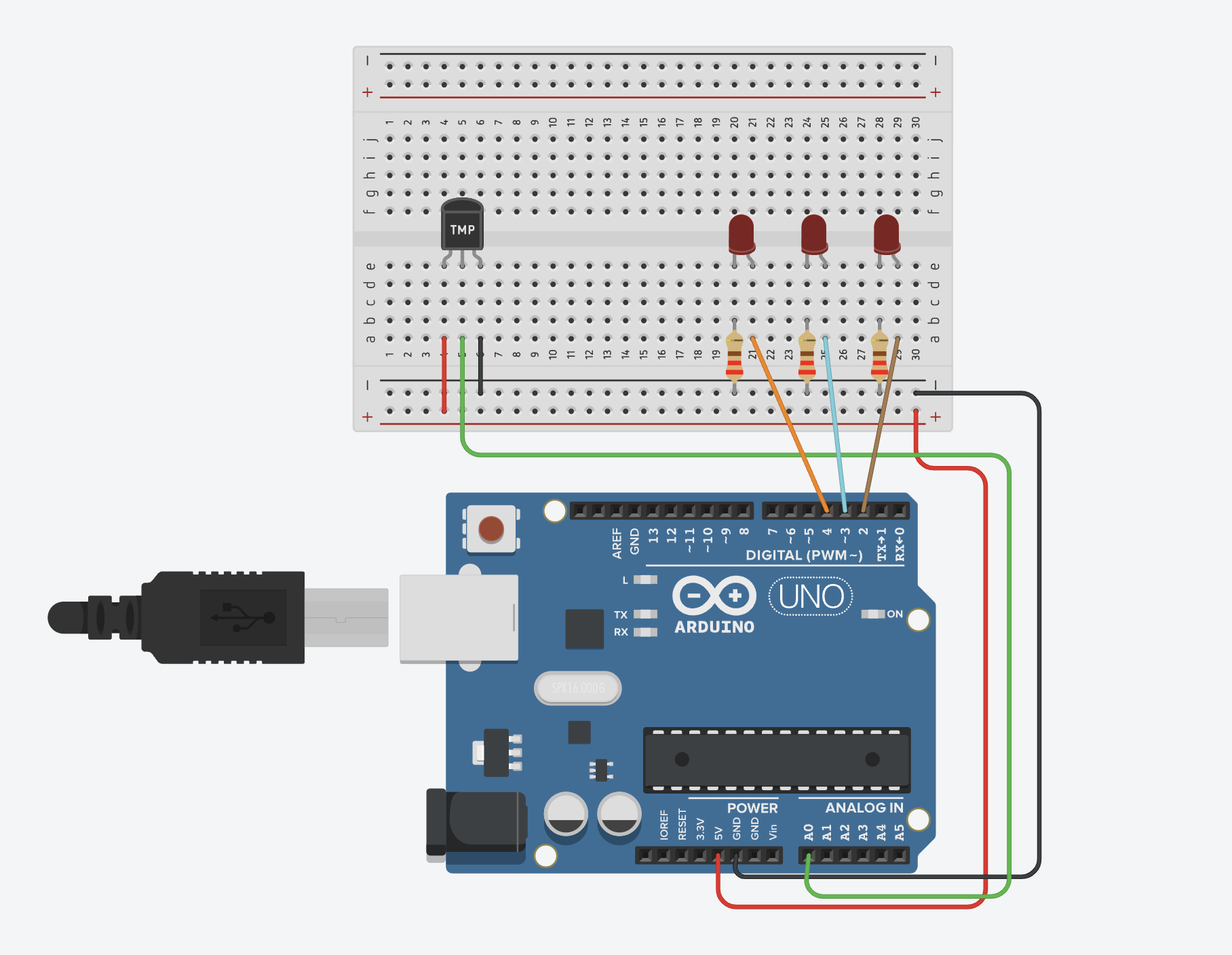


Рисунок 8 – Реализация с помощью термодатчика

Программный код:

// C++ code

//

float tempC;

void setup()

{

pinMode(A0, INPUT);

pinMode(2, OUTPUT);

pinMode(3, OUTPUT);

pinMode(4, OUTPUT);

}

void loop()

{

tempC = map(((analogRead(A0) - 20) \* 3.04), 0, 1023, -40, 125);

if (tempC > 40) {

digitalWrite(2, LOW);

digitalWrite(3, LOW);

digitalWrite(4, LOW);

}

if (tempC > 50) {

digitalWrite(2, HIGH);

digitalWrite(3, LOW);

digitalWrite(4, LOW);

}

if (tempC > 60) {

digitalWrite(2, HIGH);

digitalWrite(3, HIGH);

digitalWrite(4, LOW);

}

if (tempC > 70) {

digitalWrite(2, HIGH);

digitalWrite(3, HIGH);

digitalWrite(4, HIGH);

}

if (tempC >= 70) {

digitalWrite(2, HIGH);

digitalWrite(3, HIGH);

digitalWrite(4, HIGH);

}

delay(1000);

}