МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение   
высшего образования   
«Санкт-Петербургский государственный технологический институт  
(технический университет)»  
СПбГТИ(ТУ)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| УГНС | | 09.00.00 | | Информатика и вычислительная техника | | |
| Направление подготовки | | 09.03.01 | | Информатика и вычислительная техника | | |
| Направленность (профиль) | |  | | Системы автоматизированного проектирования | | |
| Форма обучения | |  | | очная | | |
|  | |  | |  | | |
| Факультет | |  | | Информационных технологий и управления | | |
| Кафедра | |  | | Систем автоматизированного  проектирования и управления | | |
| Учебная дисциплина | |  | **Вычислительные системы, сети и**  **телекоммуникации** | | | |
| Курс | III | | | | Группа | 494 |

**КУРСОВОЙ ПРОЕКТ**

|  |  |
| --- | --- |
| **Тема:** | **Компьютерные сети** |
| **Задача:** | **Реализация DHCP-сервера** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент |  |  |  | Казанцев А.М. |
| Руководитель,  доцент, к.т.н. |  |  |  | Макарук Р.В. |
| Оценка за курсовой проект |  |  |  |  |

Санкт-Петербург

2021-2022 уч. год

Содержание

[ВВЕДЕНИЕ 3](#_Toc94244559)

[1 Аналитический обзор 4](#_Toc94244560)

[1.1 Теоретический обзор компьютерных сетей 4](#_Toc94244561)

[1.2 Анализ технологии DHCP 5](#_Toc94244562)

[1.3 Существующие примеры реализации DHCP-сервера 9](#_Toc94244563)

[1.4 Обзор и обоснование выбора инструментальных средств разработки программного комплекса. 11](#_Toc94244564)

[2 Основная часть 12](#_Toc94244565)

[2.1 Определение структурной (иерархической) схемы решения задачи 12](#_Toc94244566)

[2.2 Определение основных этапов проектирования 12](#_Toc94244567)

[2.3 Анализ ограничений и исключительных ситуаций для алгоритмов 12](#_Toc94244568)

[2.4 Разработка основных алгоритмов задачи 13](#_Toc94244569)

[2.5 Разработка архитектуры программы 13](#_Toc94244570)

[2.6 Разработка дисплейных фрагментов 14](#_Toc94244571)

[2.7 Отладка программного комплекса 16](#_Toc94244572)

[2.8 Создание справочной системы приложения 16](#_Toc94244573)

[2.9 Создание инсталляционного пакета 17](#_Toc94244574)

[2.10 Тестирование разработанного программного продукта 20](#_Toc94244575)

[ВЫВОДЫ ПО РАБОТЕ 24](#_Toc94244576)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 25](#_Toc94244577)

[ПРИЛОЖЕНИЕ А 26](#_Toc94244578)

ВВЕДЕНИЕ

После появления интернет он довольно быстро стал очень популярен. Обмен информацией на больших расстояниях очень увлекателен. Именно межсетевой протокол IP был тем, кто объединил отдельные компьютерные сети во всемирную паутину. Неотъемлемой частью протокола является адресация сети или же IP-адрес [1].

Подробнее про IP можно сказать, что он объединяет сегменты сети в единую сеть, обеспечивая доставку пакетов данных между любыми узлами сети через произвольное число промежуточных узлов. Он классифицируется как протокол сетевого уровня по сетевой модели OSI. Важно, что он не гарантирует надёжной доставки пакета до адресата — в частности, пакеты могут прийти не в том порядке, в котором были отправлены, продублироваться оказаться повреждёнными или не прийти вовсе [2].

В современной сети Интернет используется IP четвёртой версии, также известный как IPv4. В протоколе IP этой версии каждому узлу сети ставится в соответствие IP-адрес длиной 4 байта. При этом компьютеры в подсетях объединяются общими начальными битами адреса. Количество этих бит, общее для данной подсети, называется маской подсети. Ещё существует IPv6, которое появилось ради увеличения количества узлов, а также возможности иерархической адресации [1, 2].

Таким образом получается, что очень важно уметь динамически определить свой IP-адрес, чтобы обмениваться информацией с другими. Одним из часто используемых способов являются DHCP-сервера, которые проверяют занятость адресов и арендуют для тебя один.

Целью данного курсового проекта является разработка программного комплекса реализующего основные функции DHCP-сервера, а именно: приём запроса на получение IP-адреса, ответ на него. Также очень важным будет проблема хранения занятых адресов, чтобы сервер мог знать, когда завершиться время аренды занятых IP-адресов.

Необходимо ознакомится с имеющимися на рынке аналогами небольших DHCP-серверов, проанализировать имеющиеся возможности для реализации поставленной цели в Microsoft Windows 10, а также обеспечить возможности для переноса реализуемого приложения на другой компьютер.

1 Аналитический обзор

1.1 Теоретический обзор компьютерных сетей

Компьютерная сеть (Computer Network) – это система компьютеров, связанных каналами передачи информации; программно-аппаратный комплекс, обеспечивающий автоматизированный обмен данными между компьютерами по каналам связи. Компьютерную сеть называют телекоммуникационной сетью (рисунок 1), а процесс обмена информации по такой сети называют телекоммуникацией [3].

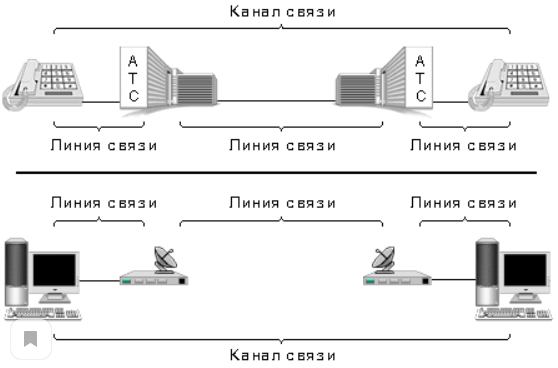


Рисунок 1 — Наглядный пример компьютерной сети

Подлинией связи обычно понимают совокупность технических устройств, и физической среды, обеспечивающих передачу сигналов от передатчика к приемнику. В реальной жизни примерами линий связи могут служить участки кабеля и усилители, обеспечивающие передачу сигналов между коммутаторами телефонной сети. На основе линий связи строятся каналы связи [3].

Каналом связи обычно называют систему технических устройств и линий связи, обеспечивающую передачу информации между абонентами. Соотношение между понятиями "канал" и "линия" описывается следующим образом: канал связи может включать в себя несколько разнородных линий связи, а одна линия связи может использоваться несколькими каналами.

По типу сетевой топологии:

* шина;
* кольцо;
* звезда.

К локальным сетям (Local Area Network, LAN) обычно относят сети, компьютеры которых сосредоточены на относительно небольших территориях (как правило, в радиусе до 1–2 км). Классическим примером локальных сетей является сеть одного предприятия, расположенного в одном или нескольких стоящих рядом зданиях. Небольшой размер локальных сетей позволяет использовать для их построения достаточно дорогие и высококачественные технологии, что обеспечивает высокую скорость обмена информацией между компьютерами [4].

Локальная сеть делится на два вида:

* одноранговая сеть;
* сеть с выделенным сервером.

Важнейшей характеристикой локальных сетей является скорость передачи данных, поэтому компьютеры соединяются с помощью высокоскоростных адаптеров со скоростью передачи данных не менее 10 Мбит/с. В локальных сетях применяются высокоскоростные цифровые линии связи. Кроме того, локальные сети должны легко адаптироваться, обладать гибкостью: пользователи должны иметь возможность располагать компьютеры, подключенные к сети там, где понадобится, добавлять или перемещать компьютеры или другие устройства, а также по необходимости отключать их без прерываний в работе сети.

Глобальные сети (Wide Area Network, WAN) – это сети, предназначенные для объединения отдельных компьютеров и локальных сетей, расположенных на значительном удалении (сотни и тысячи километров) друг от друга. Поскольку организация специализированных высококачественных каналов связи большой протяженности является достаточно дорогой, то в глобальных сетях нередко используются уже существующие и изначально не предназначенные для построения компьютерных сетей линии (например, телефонные или телеграфные). В связи с этим скорость передачи данных в таких сетях существенно ниже, чем в локальных.

В глобальных сетях для передачи информации применяются следующие виды коммутации:

коммутация каналов (используется при передаче аудиоинформации по обычным телефонным линиям связи);

коммутация сообщений применяется в основном для передачи электронной почты, в телеконференциях, электронных новостях);

коммутация пакетов (для передачи данных, в последнее время используется также для передачи аудио - и видеоинформации).

Большой интерес представляет глобальная информационная сеть Интернет.

Интернет объединяет множество различных компьютерных сетей (локальных, корпоративных, глобальных) и отдельных компьютеров, которые обмениваются между собой информацией по каналам общественных телекоммуникаций [3, 4].

1.2 Анализ технологии DHCP

*Получение адреса при одном DHCP-сервере*



Рисунок 2 — Связь DHCP-сервер - клиент

Протокол DHCP работает поверх UDP на портах 67 и 68. Сервер работает всегда только на 67, а клиент только на 68. Так как клиент не имеет адреса (имеет адрес 0.0.0.0), рассылка DHCP-пакетов производится широковещательным образом. Т.е. клиент всегда отправляет пакеты на адрес 255.255.255.255:67 с адреса 0.0.0.0:68, а сервер отправляет со своего адреса :67 на адрес 255.255.255.255:68 [5].

Получение адреса клиентом происходит в четыре пакета (**DORA**):

* Клиент выясняет, где здесь DHCP-сервер (**D**iscover)
* Сервер отзывается и предлагает свой адрес (**O**ffer)
* Клиент запрашивает предложенный адрес у конкретного сервера (**R**equest)
* Сервер соглашается и выдаёт адрес (**A**ck)

Визуально схема изображена на рисунке 3.

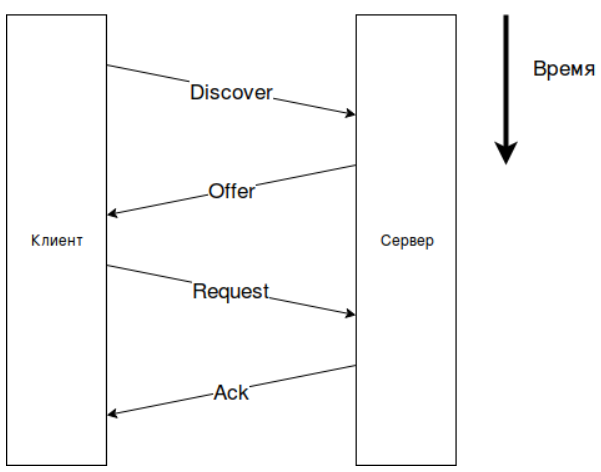


Рисунок 3 — Обмен сообщениями клиента с DHCP-сервером

*Получение адреса при нескольких DHCP-серверах*

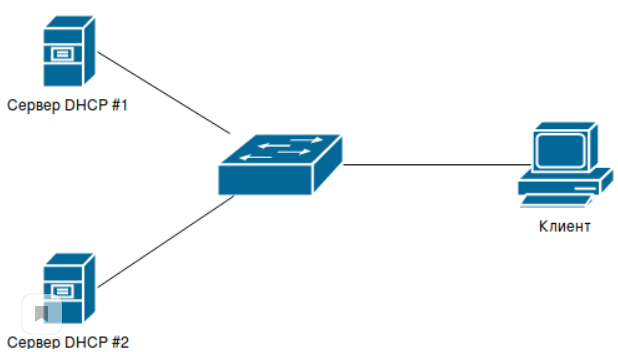


Рисунок 4 — Связь двух DHCP-серверов с клиентом

Когда клиент отправляет Discover, все сервера, которые могут слышать, присылают свой Offer клиенту. Но клиент должен выбрать кого-то одного. Выбор клиента оглашается в сообщении Request опцией 54 (DHCP-сервер), которая содержит IP-адрес предпочтённого DHCP-сервера. Хотя Request отправляется так же всем в сети, реагирует только тот DHCP-сервер, чей IP указан в опции 54 [5].

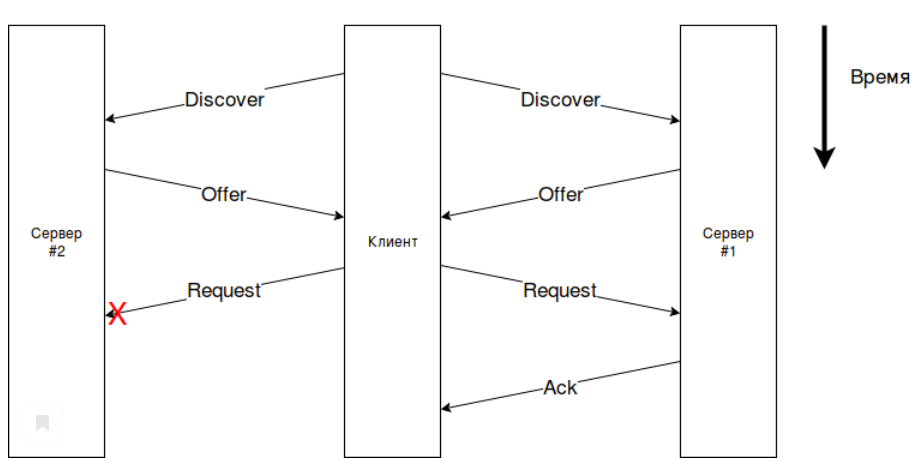


Рисунок 5 — Обмен сообщениями клиента с двумя DHCP-серверами

*Содержимое DHCP-пакета*

DHCP-пакет состоит из двух частей: постоянной, размером в 236 байт и переменной, которая несёт в себе опции (DHCP Option).

Таблица 1 — Состав постоянной части DHCP-пакета

|  |  |  |
| --- | --- | --- |
| **Поле** | **Описание** | **Длина (в байтах)** |
| **op** | Тип сообщения. Например, может принимать значения: BOOTREQUEST (0x01, запрос от клиента к серверу) и BOOTREPLY (0x02, ответ от сервера к клиенту). | 1 |
| **htype** | Тип аппаратного адреса. Допустимые значения этого поля определены в RFC 1700 «Assigned Numbers». Например, для MAC-адреса Ethernet это поле принимает значение 0x01. | 1 |
| **hlen** | Длина аппаратного адреса в байтах. Для MAC-адреса Ethernet —0x06. | 1 |
| **hops** | Количество промежуточных маршрутизаторов (так называемых *агентов ретрансляции DHCP*), через которые прошло сообщение. Клиент устанавливает это поле в 0x00. | 1 |
| **xid** | Уникальный идентификатор транзакции в 4 байта, генерируемый клиентом в начале процесса получения адреса. | 4 |
| **secs** | Время в секундах с момента начала процесса получения адреса. Может не использоваться (в этом случае оно устанавливается в 0x0000). | 2 |
| **flags** | Поле для флагов — специальных параметров протокола DHCP. | 2 |
| **ciaddr** | IP-адрес клиента. Заполняется только в том случае, если клиент уже имеет собственный IP-адрес и способен отвечать на запросы ARP (это возможно, если клиент выполняет процедуру обновления адреса по истечении срока аренды). | 4 |
| **yiaddr** | Новый IP-адрес клиента, предложенный сервером. | 4 |
| **siaddr** | IP-адрес сервера. Возвращается в предложении DHCP (см. ниже). | 4 |
| **giaddr** | IP-адрес агента ретрансляции, если таковой участвовал в процессе доставки сообщения DHCP до сервера. | 4 |
| **chaddr** | Аппаратный адрес (обычно MAC-адрес) клиента. | 16 |
| **sname** | Необязательное имя сервера в виде нуль-терминированной строки. | 64 |
| **file** | Необязательное имя файла на сервере, используемое бездисковыми рабочими станциями при удалённой загрузке. Как и **sname**, представлено в виде нуль-терминированной строки. | 128 |
|  |  |  |

Продолжение таблицы 1

|  |  |  |
| --- | --- | --- |
| **Поле** | **Описание** | **Длина (в байтах)** |
| **options** | Поле *опций DHCP*. Здесь указываются различные дополнительные параметры конфигурации. В начале этого поля указываются четыре особых байта со значениями 99, 130, 83, 99 («волшебные числа»), позволяющие серверу определить наличие этого поля. Поле имеет переменную длину, однако DHCP-клиент должен быть готов принять DHCP-сообщение длиной в 576 байт (в этом сообщении поле **options** имеет длину 340 байт). | переменная |

В опциях после особых байтов идут параметрах, по принципу:

* 1 байт номер опции;
* 1 байт показывает Х количество байт в опции;
* Х байт сами данные.

Признаком конца поля опций является параметр с номером 255 (0xFF) и длиной 0.  
  
Чаще всего клиент вкладывает параметр 55 (список параметров, которые он хочет получить в ответ) в DHCP Discover, однако, мы имеем право выдать ему не всё, что он запросил [5].

1.3 Существующие примеры реализации DHCP-сервера

Первым хотелось бы отметить программу DHCP-сервера, написанную на языке bash, основная работа с которой происходит в командной строке (рисунок 6). Она была написана энтузиастом, который хотел улучшить свои навыки программирования.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 6 — Пример реализации программы на языке bash

Такая реализация выполняет основные функции DHCP-сервера, но с определенными упрощениями:

* гарантируется, что один клиент в сети;
* гарантируется, что больше нет dhcp-серверов в сети;
* запускающий сам решает, какой адрес выдать;
* DHCP Release и DHCP Decline игнорируются.

Вторым примером я взял уже давно гуляющею по сети программу маленького DHCP-сервера (рисунок 7). Она точно так же реализует основные функции сервера, проверяя свободные адреса и выделяя пустые нуждающемся пользователям.



Рисунок 7 — Окно программы Маленького DHCP-сервера

Также нельзя не отметить программное обеспечение роутера, в котором всегда присутствует что-то подобное DHCP-серверу (рисунок 8).

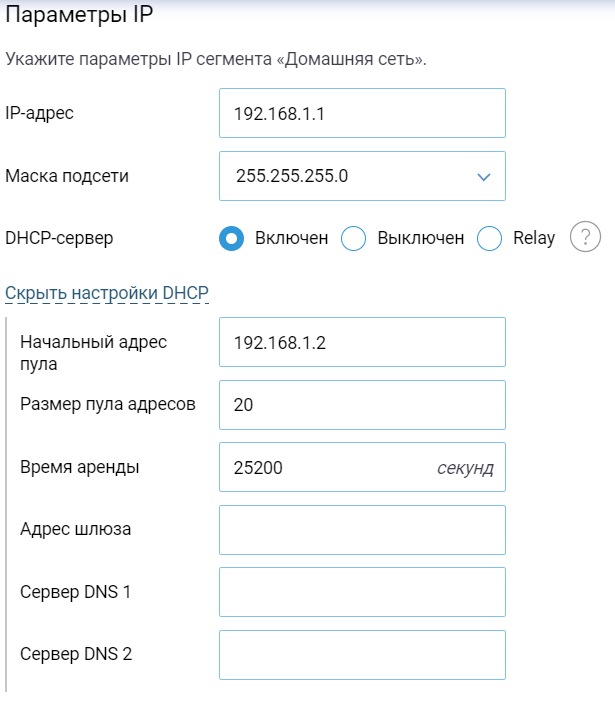


Рисунок 8 — Настройки DHCP-сервера роутера Keenetic

1.4 Обзор и обоснование выбора инструментальных средств разработки программного комплекса.

Реализация поставленной задачи требует разработки программного продукта для мониторинга контролируемых технологических параметров. Для этого было выбрано 3 языка программирования C#, C++ и Python и составлена их сравнительная характеристика, представленная в таблице ниже:

Таблица 2 — Сравнительная характеристика языков программирования

|  |  |  |  |
| --- | --- | --- | --- |
| **Критерий сравнения** | **C#** | **C++** | **Python** |
| Поддержка ООП | + | + | + |
| Сетевое программирование | + | + | + |
| Статическая типизация | + | + | - |
| Автоматический сбор мусора | + | - | + |
| Компилируемый язык | + | + | - |

По результатам сравнения (таблица 1) и учитывая, что для реализации программного комплекса не требуется возможности более низкоуровневого программирования, был выбран язык C#, располагающий автоматическим сборщиком мусора. Также данный язык программирования имеет сравнительно невысокий порог вхождения.

В качестве сред разработки были отобраны Visual Studio 2019 и Sublime Text. Sublime Text является простым редактором кода с возможностями автодополнения и выбора используемого языка, но без возможности отладки программ без установки дополнительных расширений.

Для данного курсового проекта в качестве среды разработки была выбрана Visual Studio 2019 от компании Microsoft, так как она имеет максимальную совместимость с выбранным языком программирования, позволяет отлаживать программы, располагает множеством инструментов для быстрой и качественной разработки, а также включает в себя все возможности Sublime Text.

Также для анализа сетей и изучения отправляемых и получаемых пакетов данных была выбрана программа Wireshark. Причины выбора именно этой программы состоят в том, что она проста в изучении и использовании, а также бесплатна.

2 Основная часть

2.1 Определение структурной (иерархической) схемы решения задачи

В рамках данного курсового проекта необходимо спроектировать программный комплекс, реализующий основные функции DHCP-сервера. Необходимо реализовать требуемые возможности, изображенные в виде иерархической схемы (рисунок 9).

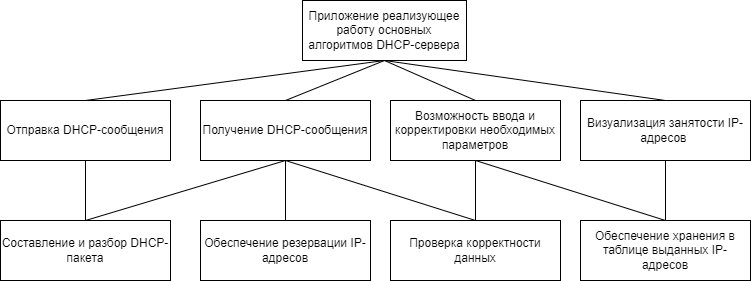


Рисунок 9 — Иерархическая схема решения задачи

2.2 Определение основных этапов проектирования

Среди этапов проектирования приложения выделены следующие:

* анализ требований и составление иерархической схемы решения задачи;
* анализ технологии DHCP;
* разработка программного обеспечения;
  + разработка специальной структуры для получения DHCP-пакета;
  + разработка интерфейсной составляющей проекта;
  + создание таблицы, сохраняющей текущие состояния выданных IP-адресов;
  + обеспечение проверки корректности входных данных:
  + обработка исключительных ситуаций;
* тестирование программного продукта;
* составление технической документации по проекту.

2.3 Анализ ограничений и исключительных ситуаций для алгоритмов

В процессе разработки программного продукта возникал ряд исключительных ситуаций, требующих обработки. Основные ограничения введены для обеспечения корректности ввода данных.

Приходящее DHCP-сообщение может просить непредставленный DHCP-серверу IP-адрес. Для определения этого происходит проверка и сервер игнорирует подобный запрос. Также может произойти ситуация, когда у сервера не будет доступа к сети, в таком случае для пользователя предусмотрена специальная надпись.

2.4 Разработка основных алгоритмов задачи

Основным алгоритмом программы являются получение, обработка, составление и отправка DHCP-пакетов.

В программе реализованы основные виды DHCP-сообщений как: **DHCPDISCOVER, DHCPOFFER, DHCPREQUEST и DHCPACK.**

**DHCPDISCOVER – это сообщение от клиента серверу, отправленное с целью предварительного получения IP-адреса, содержащие уникальный идентификатор сообщения и MAC-адрес клиента.**

**DHCPOFFER – это сообщения от сервера клиенту, содержащее свободный IP-адрес, адрес сервера, возможное время аренды IP-адреса и др.**

**DHCPREQUEST – это сообщение от клиента серверу, подтверждающее готовность клиента принять, отправленный ранее IP-адрес. Само сообщение отправляется с адреса 0.0.0.0 порта 68 на IP-адрес сервера, выданный в сообщении до этого, или на всё тот же адрес 255.255.255.255, порт всегда 67. Внутри в опции переносится принимаемый клиентом IP-адрес (50 номер опции) и может быть занесён IP-адрес сервера, с которым он работает.**

**DHCPACK – это сообщение от сервера клиенту, подтверждающее бронирование. После этого сообщения DHCP-сервер сохраняет в таблицу MAC-адрес клиента, время резервирования, а статус IP-адреса меняется со “Свободен” на “Занят”.**

2.5 Разработка архитектуры программы

Программа разбита на 2 формы:

* MainForm – отвечает за взаимодействие с пользователем, работы алгоритмов сортировки страниц, а также сохранение данных в таблицу для построения графика;
* TableRezervingIPForm – отвечает за хранение данных о выделенных программе IP-адресах.

Для более удобного использования DHCP-проекта была разработана структура данных, реализующая DHCP-пакет, структура, которого можно увидеть в таблице 1.

Основные функции алгоритмов работы с адресами и данными класса MainForm и показаны в таблице 3.

Таблица 3 — Основные функции, реализующие работу алгоритмов замещения страниц

|  |  |
| --- | --- |
| **Название функции** | **Назначение** |
| Helpreadmessage | Получает и отфильтровывает данные DHCP-пакета |
| AddInDHCPOpption | Добавляет массив байтов в опции DHCP-пакета перед конечным значением 255 |
| OptionValue | Получает по номеру нужное опцию, если её не было, то отправляет null |
| TypeDHCPMessage | Возвращает тип полученного сообщения |
| MassivByteInMessage | Преобразует массив байт IP-адреса в тип string и возвращает его |
| AlreadyUsingIPAdress | Проверяет свободен ли IP-адрес |
| ReadDHCPStruct | Преобразует полученный массив байтов в DHCP структуру |
| MessagecreateDHCPStruct | Преобразует DHCP структуру в массив байтов для отправки сообщения |
| TimerRezervation\_Tick | Функция, проверяющая выбранные IP-адреса, а также уменьшающая время уже зарезервированных IP-адресов |

Также помимо создания перечисленных функций использованы методы для обработки визуального взаимодействия пользователя и программы.

2.6 Разработка дисплейных фрагментов

При первом запуске программного продукта появляется рабочая область, ожидающая ввода исходных данных и запуска сервера (рисунок 10).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 10 — Главное окно приложения при первом запуске

Размер пула адресов и максимальное время имеют ограничены, не дающее перейти пределы, также ip-адреса не буду регистрироваться, если они не подходят под общие правила.

После запуска сервера происходит блокирование настроек, а надпись на кнопке меняется с “Запустить сервер” на “Выключить сервер” (рисунок 11), также управляющему сервером доступно окно “Таблица IP-адресов”, в котором записываются забронированные адреса и те, кто осуществляет эту бронь, с временем бронирования (рисунок 12).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 11 — Окно запущенного сервера

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 12 — Таблица занятости IP-адресов

Одной из самых вероятных проблем, с которыми может столкнуться DHCP-сервер, когда у него ну будет подключение к сети. При таких обстоятельствах пользователю выводиться специальное сообщение (рисунок 13).

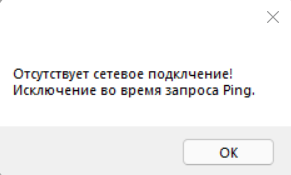


Рисунок 13 — Сообщение об отсутствии сети

В меню в верхней части окна приложения при наведении на «Общее» появиться две кнопки: «Справка» и «О программе», открывающие информационные окна с краткой справочной информацией и сведениях о программном продукте и его авторе (рисунок 14, 15).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 14 - Справочная система приложения

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 15 — Информационное окно «О программе»

2.7 Отладка программного комплекса

Отладка программного комплекса производилась путем проверки поведения различных частей программного кода при изменяющихся настройках программы: увеличение количества страниц, неправильный ввод IP-адреса, отрицательное время резервирования. В процессе отладки были выявлены, локализованы и устранены следующие ошибки:

* выдача IP-адресов, которых не было в пуле;
* ошибки, связанные с невозможностью выйти в сеть;
* ошибки, неправильного приёма DHCP-пакетов.

2.8 Создание справочной системы приложения

Для создания справочной системы использовался пакет программ CHM Editor (рисунок 16), позволяющий редактировать файлы с расширением .chm, добавлять текстовую и графическую информацию на страницы справки, а также структурировать ее содержимое.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 16 — Основное окно программы для создания справочной системы

В процессе создания справки были описаны основные возможности программного продукта и способы взаимодействия с приложением [6]. Шаги и способы работы с программным комплексом для полноты восприятия сопровождаются рисунками (рисунок 17).

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 17 — Фрагмент справочной системы

Также в справочной системе присутствует поиск по ключевым словам для более быстрого взаимодействия и поиска необходимой информации.

2.9 Создание инсталляционного пакета

Для создания инсталляционного пакета использовался программный комплекс Microsoft Visual Studio Installer Projects.

Первым шагом создания установочного пакета является создание проекта установочного файла внутри решения нашей программы (рисунок 18).

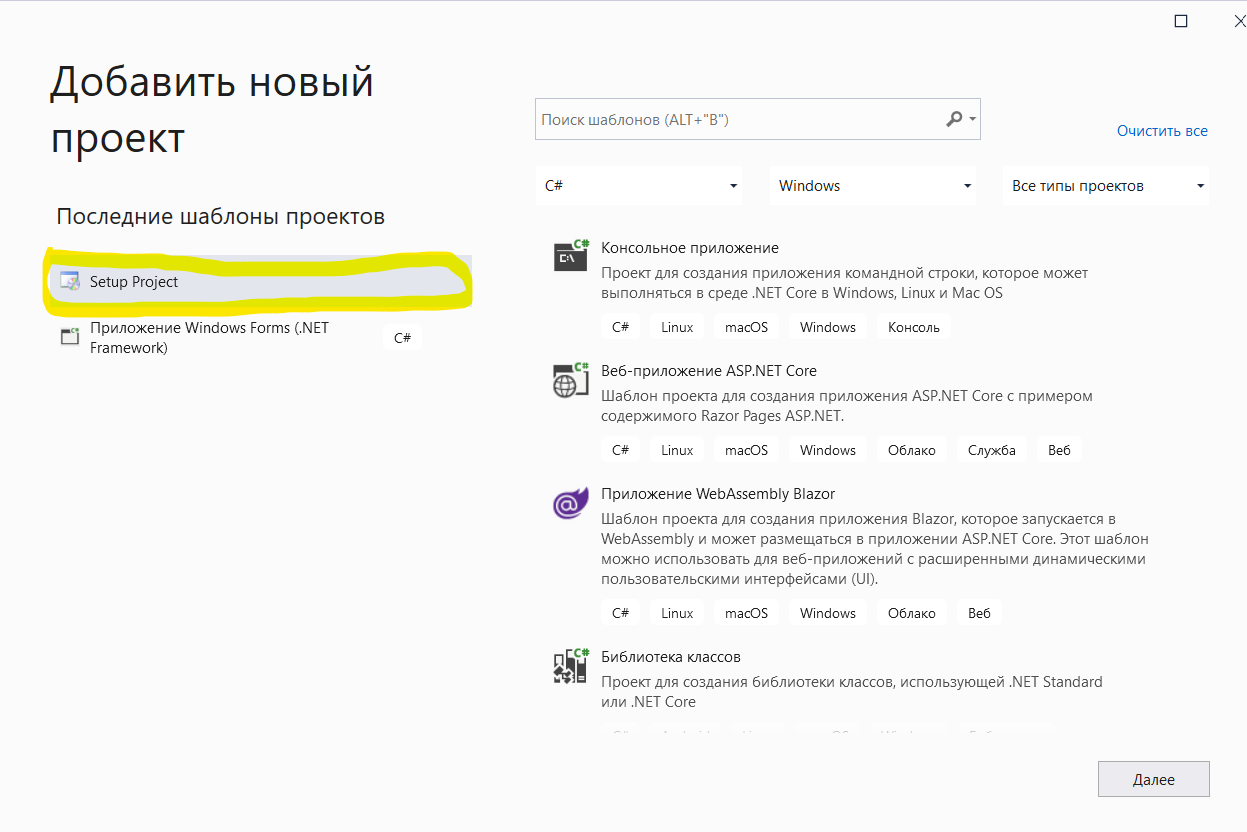


Рисунок 18 — Начало создания инсталляционного пакета

Далее необходимо указать название проекта (рисунок 19, 20).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 19 — Создание установочного проекта

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 20 — Созданный проект

На следующем шаге необходимо создать Shortcut проекта и добавить его в папки User’s Desktop и App, созданный внутри User’s Programs Menu (рисунки 21, 22, 23).

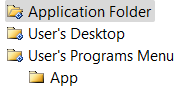
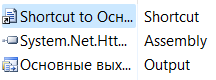
 

Рисунок 21 — Создание Shortcut

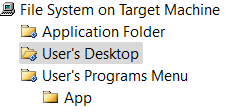
 

Рисунок 22 — Перемещение в User’s Desktop и переименование Shortcut

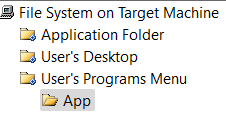
 

Рисунок 23 — Перемещение в App и переименование Shortcut

Далее в свойствах установочного проекта (рисунок 24) можно изменить автора приложения, компанию и другие конфигурации.

Изображение выглядит как текст, стол

Автоматически созданное описание

Рисунок 24 — Изменение некоторых конфигураций установочного проекта

Последним шагом является сборка установочного проекта, далее, если посмотреть в файлы проекта, то можно найти его установщик (рисунок 25).

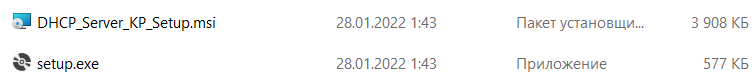


Рисунок 25 — Файл установки программы и его дополнительные файлы

2.10 Тестирование разработанного программного продукта

Для установки программного продукта необходимо открыть мастер установки и выполнить ряд шагов (рисунок 26).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 26 — Мастер установки приложения

Выбрав меню «Далее», доступен выбор папки, куда будет установлено приложение, а также выбор для кого будет доступно это приложение после установки (рисунок 27).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 27 — Опции установки

После нажатие кнопки “Долее” происходит подтверждение установки (рисунок 28), мы снова выбираем “Далее” и начинается установка, после чего появляется окно завершения установки (рисунок 29).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 28 — Подтверждение установки

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 29 — Процесс установки

После завершение на компьютере пользователя появляется каталог, содержащий все необходимые данные и ярлык главного исполняемого файла (рисунок 30).

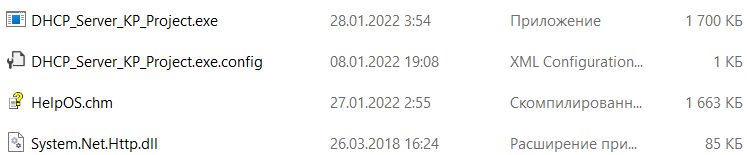


Рисунок 30 — Результат установки приложения

Тестирование программного комплекса, реализующего работу основных функций DHCP-сервера, происходило на приеме и подтверждение DHCP-сообщения от моего телефона, на рисунке 31 показано отключение DHCP-сервера роутера, настройки программы (рисунок 32) и начальная таблица резервирования IP-адресов (рисунок 33).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 31 — Отключение DHCP-сервера на роутере

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 32 — Основное окно программы с определенными настройками тестирования

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 33 — Начальная таблица IP-адресов

После запуска сервера он переходит в рабочее состояние: текст на кнопке меняется, а возможность настройки блокируется (рисунок 34). Так как телефон до этого уже имел занятый IP-адрес, но не знает IP-адрес старого DHCP-сервера, то он сразу отправляет сообщение типа **DHCPREQUEST по широковещательному каналу, на что DHCP-сервер отвечает DHCPACK со своего адреса на зарезервированный им. Их общение было перехвачено с помощью программу Wireshark (рисунок 35, 36). После этого в таблицы занятости IP-адресов статус соответствующего адреса меняется на “Занят”, устанавливается время и подписывается MAC-адрес телефона (рисунок 37). По прошествию 60 секунд таймер уменьшает время занятого IP-адрес (рисунок 38).**

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 34 — Основное окно программы после запуска сервера



Рисунок 35 — Окно программы Wireshark с перехваченными сообщениями

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 36 — Номер опции, показывающий резервируемый IP-адрес

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 37 — Таблица занятости IP-адресов после резервирования

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 38 — Таблица занятости IP-адресов по прошествию 60 секунд

ВЫВОДЫ ПО РАБОТЕ

В процессе выполнения данной курсовой работы была создана программа реализующая основные функции DHCP-сервера. В ходе разработки были выполнены следующие задачи:

* обоснование выбора инструментальных средств разработки;
* разработка структурной схемы решения задачи;
* разработка структуры данных, определение диапазона изменений;
* анализ ограничений и исключительных ситуаций;
* разработка основных алгоритмов программы;
* разработка архитектуры и дисплейных фрагментов приложения;
* создание и отладка модулей программного комплекса;
* создание справочной системы программного продукта;
* создание инсталляционного пакета программы;
* тестирование разработанного программного продукта;
* разработка эксплуатационного документа «Руководство системного программиста»;
* оформление отчетной документации по проекту.

Перспективы развития приложения предполагают внедрение полного спектра функций DHCP-сервера, а также возможности изменения маски подсети и других важных параметров.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Чеппел, Л. А. TCP/IP. Учебный курс : учебное пособие / Л.А Чеппел, Эд. Титтел — Санкт-Петербург : Питер, 2003. — 960 с. — ISBN 5-94157-315-4, 0-619-18654-2
2. Хант, К. TCP/IP. Сетевое администрирование : учебное пособие / К. Хант, В.В. Шилов — Санкт-Петербург : Питер, 2008. — 814 с. — ISBN 978-5-93286-056-4
3. Таненбаум, Э. Компьютерные сети / Э. Таненбаум, Д. Уэзеролл Пер. с англ. — Санкт-Петербург : Питер, 2016. — 960 с.
4. [Олифер, В.](http://irb.technolog.edu.ru/CGI/irbis64r_12/cgiirbis_64.exe?LNG=&Z21ID=&I21DBN=EK&P21DBN=EK&S21STN=1&S21REF=1&S21FMT=fullwebr&C21COM=S&S21CNR=20&S21P01=0&S21P02=1&S21P03=A=&S21STR=%D0%A0%D1%83%D1%81%D1%81%D0%B8%D0%BD%D0%BE%D0%B2%D0%B8%D1%87,%20%D0%9C%D0%B0%D1%80%D0%BA) Компьютерные сети. Принципы, технологии, протоколы. Учебник : учебное пособие / В. Олифер, Н. Олифер. — 5-е изд. — СПб.: Питер, 2016. — 996 с.
5. Дромс, Р. The DHCP Handbook / Р. Дромс, Т. Лемон— 2-е изд. — Индианаполис : SAMS, 2003. — 623 с.
6. Гультяев, А.К. Help. Разработка справочных систем. Учебный курс / А.К. Гультяев. Санкт-Петербург : Питер, 2004. — 272 с. — ISBN 5-94723-921-3

ПРИЛОЖЕНИЕ А

(обязательное)

МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение   
высшего образования   
«Санкт-Петербургский государственный технологический институт  
(технический университет)»  
СПбГТИ(ТУ)

СОГЛАСОВАНО УТВЕРЖДАЮ

проф., д.т.н. \_\_\_\_\_\_\_\_\_\_\_\_\_Т.Б.Чистякова

\_\_\_\_\_\_\_\_\_\_\_\_\_

**«DHCP-сервер»**

Руководство системного программиста

ЛИСТ УТВЕРЖДЕНИЯ

460.2.030. 00001–01 34 01–ЛУ

Руководитель курсового проекта

Р. В. Макарук

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_2021

Исполнитель:

А.М. Казанцев

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_2021

2021

Продолжение приложения А

МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение   
высшего образования   
«Санкт-Петербургский государственный технологический институт  
(технический университет)»  
СПбГТИ(ТУ)

УТВЕРЖДЕН

460.2.030. 00001–01 34 01–ЛУ

**«СИСТЕМА СРАВНЕНИЯ РАЗЛИЧНЫХ АЛГОРИТМОВ ЗАМЕЩЕНИЯ СТРАНИЦ»**

Руководство системного программиста

460.2.030.00001–01 34 01

Листов 8

2021

Продолжение приложения А

2

460.2.030.00001–01 34 01

**АННОТАЦИЯ**

В данном программном документе приведено руководство системного программиста по настройке и использованию программы «DHCP\_Server\_KP\_Project.exe», предназначенной для распределения IP-адресов сети.

В данном программном документе, в разделе «Общие сведения о программе» указаны назначение и функции программы и сведения о технических и программных средствах, обеспечивающих выполнение данной программы, а также требования к персоналу.

В разделе «Структура программы» приведены сведения о структуре программы, ее составных частях, о связях между составными частями и о связях с другими программами.

В данном программном документе, в разделе «Настройка программы» приведено описание действий по настройке программы на условия конкретного применения (настройка на состав технических и программных средств, выбор функций и др.).

В разделе «Проверка программы» приведено описание способов проверки, позволяющих дать общее заключение о работоспособности программы (контрольные примеры, методы прогона, результаты).

В данном программном документе, в разделе «Сообщения системному программисту» указаны тексты сообщений, выдаваемых в ходе выполнения настройки, проверки программы, а также в ходе выполнения программы, описание их содержания и действий, которые необходимо предпринять по этим сообщениям.

Продолжение приложения А

3

460.2.030.00001–01 34 01

СОДЕРЖАНИЕ

Аннотация 2

Содержание 3

1 Общие сведения о программе 4

1.1. Назначение программы 4

1.2. Функции программы 4

1.3. Минимальный состав технических средств 5

1.4. Минимальный состав программных средств 5

1.5. Требования к персоналу (системному программисту) 5

2 Структура программы 6

2.1. Сведения о структуре программы 6

2.2. Сведения о составных частях программы 6

2.3. Сведения о связях между составными частями программы 6

2.4. Сведения о связях с другими программами 6

3 Настройка программы 7

3.1. Настройка на состав технических средств 7

3.2. Настройка на состав программных средств 7

4 Проверка программы 7

4.1. Описание способов проверки 7

4.2. Методы прогона 7

4.2.1. Проверка работоспособности программы 7

4.2.2. Проверка на сообщение об ошибке 7

5 Сообщения системному программисту 8

Продолжение приложения А

4

460.2.030.00001–01 34 01

1. **ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ**

1.1 Назначение программы

После появления интернет он довольно быстро стал очень популярен. Обмен информацией на больших расстояниях очень увлекателен. Именно межсетевой протокол IP был тем, кто объединил отдельные компьютерные сети во всемирную паутину. Неотъемлемой частью протокола является адресация сети или же IP-адрес.

Подробнее про IP можно сказать, что он объединяет сегменты сети в единую сеть, обеспечивая доставку пакетов данных между любыми узлами сети через произвольное число промежуточных узлов. Он классифицируется как протокол сетевого уровня по сетевой модели OSI. Важно, что он не гарантирует надёжной доставки пакета до адресата — в частности, пакеты могут прийти не в том порядке, в котором были отправлены, продублироваться оказаться повреждёнными или не прийти вовсе.

В современной сети Интернет используется IP четвёртой версии, также известный как IPv4. В протоколе IP этой версии каждому узлу сети ставится в соответствие IP-адрес длиной 4 байта. При этом компьютеры в подсетях объединяются общими начальными битами адреса. Количество этих бит, общее для данной подсети, называется маской подсети. Ещё существует IPv6, которое появилось ради увеличения количества узлов, а также возможности иерархической адресации.

Таким образом получается, что очень важно уметь динамически определить свой IP-адрес, чтобы обмениваться информацией с другими. Одним из часто используемых способов являются DHCP-сервера, которые проверяют занятость адресов и арендуют для тебя один.

Данная программа реализует основные функции DHCP-сервера, а именно: приём запроса на получение IP-адреса, ответ на него.

Программа «DHCP\_Server\_KP\_Project.exe» работает под управлением ОС Windows 10.

* 1. Функции программы

Основной функцией программы «DHCP\_Server\_KP\_Project.exe» является выдача свободного IP-адреса клиенту, запрашивающего его.

Продолжение приложения А

5

460.2.030.00001–01 34 01

Основная задача - обеспечить работу прием и оперативный ответ на DHCP-сообщения.

Данные функции программы DHCP\_Server\_KP\_Project позволяют производить корректировку входных и выходных данных, мониторинг занятости IP-алресов, а также просматривать, кто занял IP-адреса.

* 1. Минимальный состав технических средств

Минимальный состав используемых технических средств:

PC на базе процессора с тактовой частотой 1 ГГц или выше;

ОЗУ более 1 Гб;

512 МБ видеопамяти и выше.

1.4. Минимальный состав программных средств

Системные программные средства, используемые программой DHCP\_Server\_KP\_Project.exe, должны быть представлены локализованной версией операционной системы Windows 10.

1.5 Требования к персоналу (системному программисту)

Системный программист должен иметь минимум среднее техническое образование. В перечень задач, выполняемых системным программистом, должны входить:

а) задача поддержания работоспособности технических средств;

б) задача установки (инсталляции) и поддержания работоспособности системных программных средств – операционной системы;

в) задача установки (инсталляции) и поддержания работоспособности программы DHCP\_Server\_KP\_Project.exe.

**2 СТРУКТУРА ПРОГРАММЫ**

2.1 Сведения о структуре программы

Программа «DHCP\_Server\_KP\_Project.exe» состоит из одной запускаемой формы.

Продолжение приложения А

6

460.2.030.00001–01 34 01

2.2 Сведения о составных частях программы

Программа «DHCP\_Server\_KP\_Project.exe» состоит из одной запускаемой формы, а также отдельно запускаемой справочной системой.

2.3 Сведения о связях между составными частями программы

Программа «DHCP\_Server\_KP\_Project.exe» способна запускать справочную систему и форму с таблицей занятости IP-адресов при нажатии определенной кнопки интерфейса.

2.4 Сведения о связях с другими программами

Программа «DHCP\_Server\_KP\_Project.exe» в ходе своей работы не требует связи с другими программами.

**3 НАСТРОЙКА ПРОГРАММЫ**

3.1 Настройка на состав технических средств

Программа «DHCP\_Server\_KP\_Project.exe» не требует каких-либо настроек на состав технических средств.

3.2 Настройка на состав программных средств

Программа «DHCP\_Server\_KP\_Project.exe» устанавливает необходимые для работы компоненты при установке программного продукта при помощи инсталляционного пакета.

Продолжение приложения А

7

460.2.030.00001–01 34 01

**4 ПРОВЕРКА ПРОГРАММЫ**

4.1 Описание способов проверки

Работоспособность программы проверяется следующим способом. Необходимо взять устройство, которое раньше уже получало IP-адрес и, отключив другие DHCP-сервера, выдать ему тот же, а по возможности и новый IP-адрес. В случае корректности работы программы в таблице занятости адресов будет отмечен MAC-адрес подключенного устройства, время, а выданный IP-адрес будет помечен как “Занят”, так же само устройство должно суметь подключиться к сети. В случае некорректности работы «DHCP\_Server\_KP\_Project.exe» выдаст сообщение об ошибке, описанное в разделе 5. «Сообщения системному программисту» данного программного руководства.

4.2 Методы прогона

4.2.1 Проверка работоспособности программы

Проверьте наличие сетевого подключения.

Проверьте наличие файла HelpOS.chm для корректного отображения файловой системы

При работе с большим набором IP-адресов проверьте наличие достаточного количества свободной оперативной памяти и процессорного времени

4.2.2 Проверка на сообщение об ошибке

В случае невозможности подключения к сети программа «DHCP\_Server\_KP\_Project.exe» должна выдать сообщение об ошибке, описанное в разделе 5. «Сообщения системному программисту» данного программного руководства.

Продолжение приложения А

8

460.2.030.00001–01 34 01

**5 СООБЩЕНИЯ СИСТЕМНОМУ ПРОГРАММИСТУ**

Программа «DHCP\_Server\_KP\_Project.exe» выдает: сообщения об ошибке, показанное на рисунке 1.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1 – Сообщение системному программисту

ПРИЧИНА. Отсутствие сетевого подключения, не возможность отправлять и получать какие-либо сообщения.

ДЕЙСТВИЯ ПРОГРАММЫ. Программа «DHCP\_Server\_KP\_Project.exe» не начинает работу основных функций DHCP-сервера.

ДЕЙСТВИЯ ПРОГРАМИСТА. Проверить и вернуть сетевое подключение.