

УДК 621.39

## **ПЕРЕДАЧА ВИДЕО ФАЙЛОВ ПО БЕСПРОВОДНЫМ МУЛЬТИМЕДИЙНЫМ СВЕРХШИРОКОПОЛОСНЫМ СЕНСОРНЫМ СЕТЯМ**

**Т. А. Шапошникова, А. И. Чернышов,**

**Московский физико-технический институт, факультет проблем физики и энергетики  
Институт радиотехники и электроники им. В.А. Котельникова РАН**

Получена 1 ноября 2012 г.

**Аннотация.** Рассматривается задача передачи видео файлов по беспроводным мультимедийным сверхширокополосным сенсорным сетям. Разработаны и экспериментально исследованы схемы и алгоритмы передачи видео данных в режиме точка-точка и с введением ретрансляции сигнала в таких сетях. Рассматривается вопрос сжатия и качества радиоканала, а также решена проблема, связанная с вероятностью потери пакетов.

**Ключевые слова:** беспроводные мультимедийные сенсорные сети, ППС-40, пакет, ретрансляция, радиоканал.

**Abstract:** Considering the transmission of video files via wireless multimedia sensor networks. Schemes and algorithms of video files transmission (point-to-point mode and mode of signal retransmission) have been developed and experimentally investigated. Considering the problems of file's compression, quality of radio channel and packet loss.

**Key words:** wireless multimedia sensor networks, PPS-40, packet, retransmission, radio channel.

### **1. Введение**

Интеграция маломощных беспроводных сетевых технологий таких как беспроводные сенсорные сети (БСС) и недорогих CMOS (КМОП – комплементарная логика на транзисторах металл-оксид-полупроводник) камер

и микрофонов позволило развитие так называемых БМСС (беспроводные мультимедийные сенсорные сети - WMSNs), то есть сетей беспроводных устройств, способных считывать мультимедийный контент, такой как аудио, изображения и видео. БМСС могут быть использованы для новых приложений, включая системы наблюдения и контроля, услуги по оказанию медицинской помощи и телемедицину [1].

По сравнению с обычными беспроводными сенсорными сетями, мультимедийные сенсорные сети требуют большего количества вычислительных ресурсов в своих узлах и более высоких скоростей передачи данных.

В данной работе рассматривается только часть этой проблемы, а именно способность беспроводной сенсорной сети с использованием прямохаотической сверхширокополосной схемы связи [2] состоящей из устройств ППС40 (прямохаотический приёмопередатчик сверхширокополосный) [3] обеспечить достаточную скорость и качество передачи большого объёма данных. Для этого достаточно организовать передачу мультимедийного сжатого файла с одного ПК на другой по схемам «точка-точка» и схеме с использованием ретрансляции.

Остановимся более подробно на самих устройствах ППС-40. Приёмопередатчик ППС-40 предназначен для передачи и приёма данных, поступающих от подключенного к нему источника/приёмника данных. В качестве последнего может выступать либо сенсорный узел, либо персональный компьютер, подключаемый с помощью устройства сопряжения и кабеля. Помимо режима передачи и приёма данных, поступающих через внешний интерфейс (разъём) X1, показанный на рис. 1, приёмопередатчик может работать как ретранслятор в составе сенсорной сети.



Рис. 1. Приёмопередатчик

Приёмопередатчик включает в себя (рис. 1.) генератор сверхширокополосных хаотических колебаний (Г), антенну (А), логарифмический приёмник (ЛП), ключ (К), коммутирующий сигнал между антенной, генератором и приёмником, и цифровой блок (ЦБ).

Плюсами данного устройства являются: компактность, малое потребление энергии, а также возможность работы в режиме приемника, передатчика и ретрансляции (одновременно как приемник и передатчик);

Значительным минусом в данной системе передачи является: использование com-порта, что ограничивает скорость передачи информации.

## 2. Вопрос сжатия

Немало важным вопросом, при осуществлении передачи видео в реальном времени, является фактическая скорость передачи, которая напрямую зависит от объема передаваемой информации. Уменьшение объема одного кадра с минимальной потерей качества осуществляется за счет сжатия. Но при использовании существующих эффективных форматов сжатия, таких как (.jpeg) или (.mp4), возникает проблема, связанная с воспроизведением файла после его передачи при потере хотя бы одного пакета данных, т.к. при этом невозможно декодирование данных. Поэтому от беспроводной сенсорной сети требуется передача файла, без потери данных в канале.

Для проведения экспериментов был выбран файл в формате (.avi), но хотелось бы отметить, что формат в данном случае не играет большой роли.

Главное что данные закодированы так, что потеря даже одного пакета является критической для воспроизведения на приемнике.

### 3. Алгоритм работы программы для приёма и передачи данных на микроконтроллере

Рассмотрим сначала более детально, как осуществлялась работа программы для микроконтроллера, отвечающая за передачу данных. Программа написана на языке assembler в среде AVR Studio. Её можно разбить условно на две части: в первой идёт настройка различных интерфейсов микроконтроллера, его периферии и он подготавливается для дальнейшей работы; во второй же части содержится непосредственно алгоритм работы устройства.

После этой инициализации идёт сам алгоритм работы передающего устройства, схема которого представлена на рис. 2. Сначала происходит обнуление счётчика текущего количества принятых с ПК байт (информация об этом записывается в определённый регистр). Далее если микроконтроллер получает сигнал о том, что поступили данные с UART (данные приходят от компьютера), то пришедший байт записывается в регистр специального назначения UDR, откуда затем поступает в ОЗУ микроконтроллера. После этого сравнивается значение счётчика текущего количества принятых с ПК байт с размером пакета и в случае, если оно не меньше размера пакета, формируется пакет для передачи данных в эфир. Структура пакета представлена в таблице 1.

Таблица 1

Преамбула	Размер пакета	Тип пакета	ID получателя	ID отправителя	Data			CRC
					Data1	Data...	DataN	
1	1	1	1	1				2

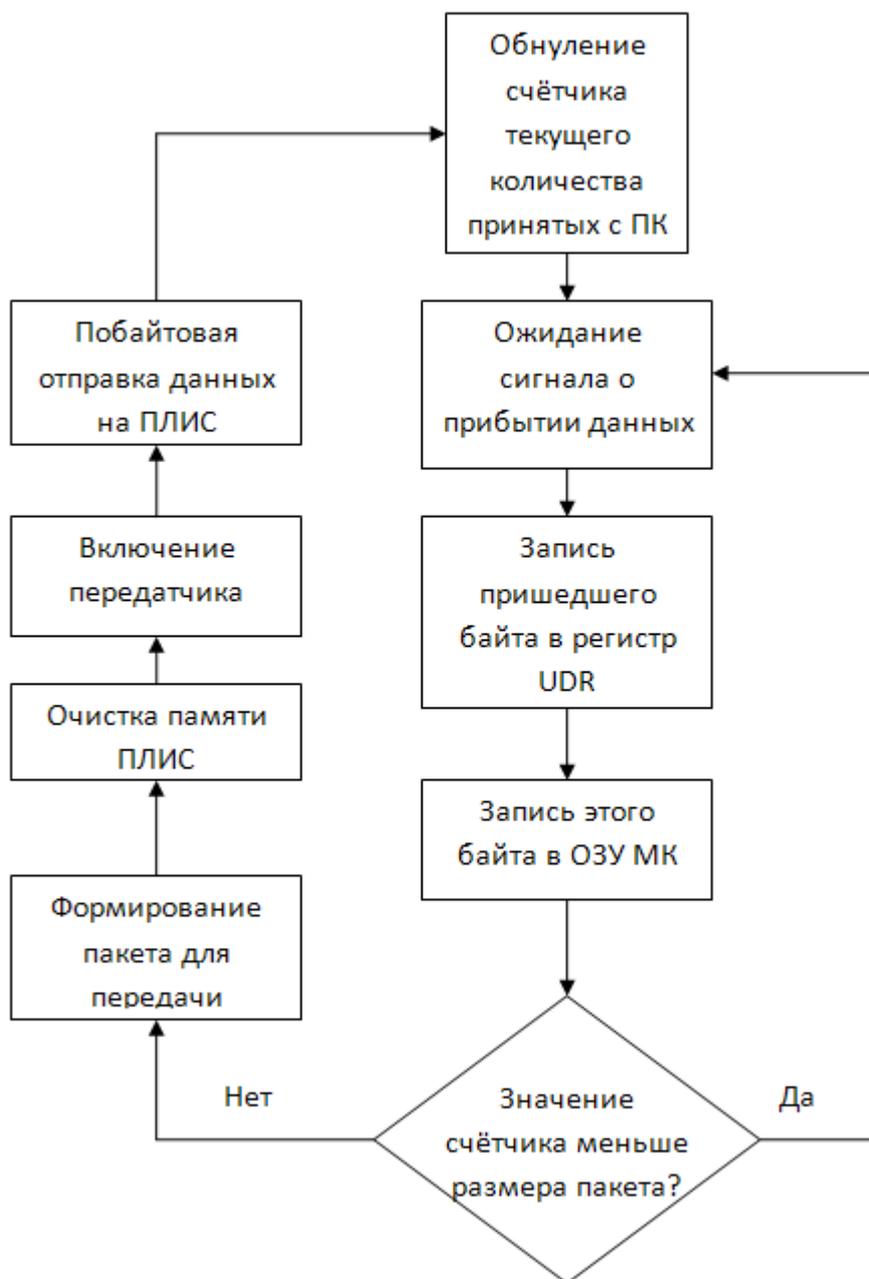


Рис. 2. Алгоритм работы микроконтроллера на передающем устройстве

Перед посылкой данных происходит очистка памяти ПЛИС и включается передатчик. Далее передаваемые данные поступают побайтно на ПЛИС и идут на аналоговую часть устройства.

Теперь опишем программу, разработанную для управления микроконтроллером и ответственную за приём данных. Часть программы,

связанная с инициализацией микроконтроллера, аналогична случаю передачи данных.

Алгоритм работы приёмного устройства: сначала включается приёмник, затем вызывается функция, отвечающая за ожидание приёма данных с ПЛИС и за отправку этих данных на ПК (алгоритм работы этой функции представлен в блок-схеме на рис. 3.). При поступлении байта данных на ПЛИС происходит запись байта данных на ОЗУ микроконтроллера. Если микроконтроллер получает сигнал на определённую «ножку» о том, что в ПЛИС поступил байт данных, то этот байт записывается в ОЗУ микроконтроллера. Первые два байта записываются таким образом в память микроконтроллера, причём второй байт сохраняется как байт, несущий информацию о размере пакета. В процессе побайтового приёма указанное количество байт записывается в буфер для последующей отправки по UART. После этого происходит подсчёт контрольной суммы для данного пакета и в случае, если она сходится с тем, что приходит в пакете, происходит отправка поля данных из буфера на UART (помимо полезных данных, в пакете содержится также служебная информация, описанная выше в структуре посылаемого пакета).

Нужно сделать замечание касательно скорости передачи данных с ПК на микроконтроллер при использовании интерфейса UART и его настроек, описанных выше. Пусть  $R$  – скорость передачи данных (единица измерения: бит/с). Тогда для передачи каждого бита будет затрачено  $t = 1/R$  (с).

Далее, при использовании стопового и стартового бита, на каждый байт данных при передаче его по UART будет уходить не 8, а 10 бит, то есть будет затрачено  $10/R$  (с). Таким образом, максимально возможная скорость передачи составляет  $R/10$  (байт/с). На практике же при передаче данных по радио эфиру используется пакетная передача данных, где при передаче каждого пакета возникает пауза в чтении данных с UART, что сказывается на скорости передачи данных. Так для длины пакета в  $L$  байт и задержки  $D$  (с) между посылкой пакетов получаем формулу для скорости передачи:

$$R_{packet} = \frac{L}{10L/R + D} \quad (1)$$

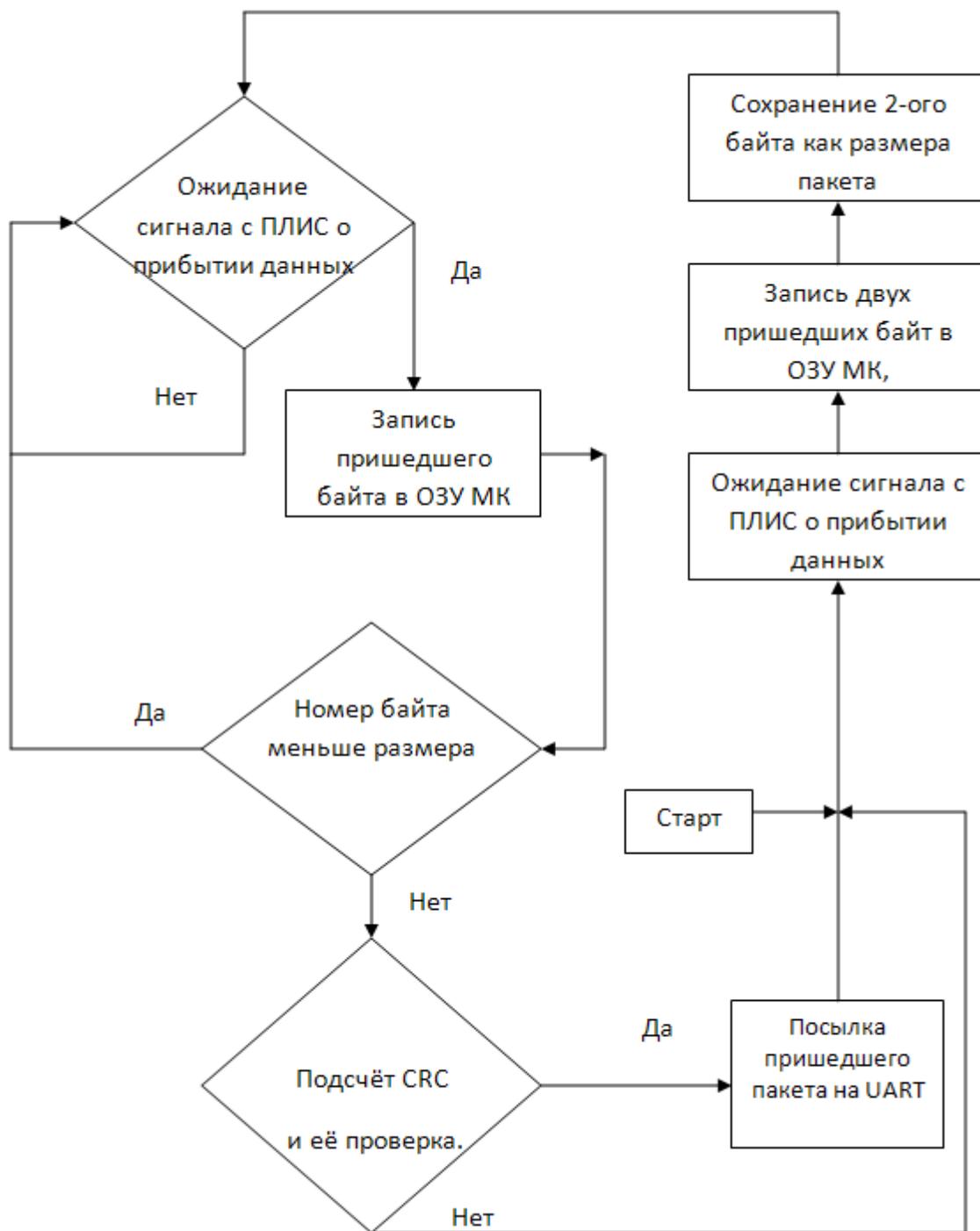


Рис. 3. Алгоритм работы микроконтроллера на принимающем устройстве.

Графики зависимости скорости передачи от задержки для длины пакета 10 и 30 байт соответственно представлены на рис. 4. Отсюда можно сделать вывод, что для сохранения скорости на, например, уровне 10000 байт/с, нужно делать короткие задержки при использовании небольших пакетов, или можно увеличить паузу между пакетами за счёт увеличения их размера. Так скорость 10000 байт/с может быть получена при 10-байтных пакетах и задержке около 120 мкс, или при 30-байтных пакетах и задержке 400 мкс.

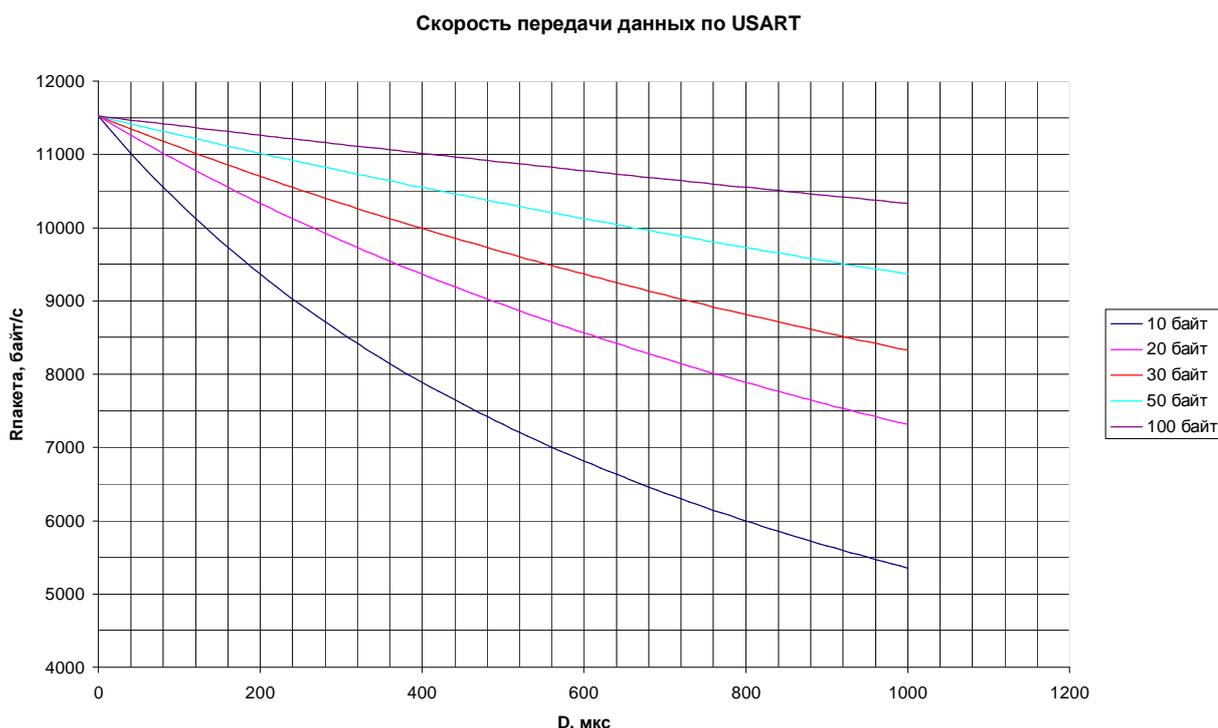


Рис. 4. Скорость передачи данных по интерфейсу UART для различных длин пакета при его настройке на скорость 115 200 бит/с в зависимости от задержки между пакетами.

#### 4. Система передачи информации

Первоначально рассмотрим ситуацию, когда у нас есть одна базовая станция, собирающая информацию и отправляющая её на ПК (персональный компьютер) и один передающий узел, данные на который поступают с ПК.

Итак, рассматривается передача видео (.avi) файлов в системе точка-точка с использованием прямохаотических приёмопередатчиков ППС-40. Была

разработана программа для передачи данных на ПК, с помощью которой информация передаётся через виртуальный СОМ-порт на передающее устройство ППС-40. На нём происходит обработка принимаемых данных, формирование пакета и отправка их по радиоканалу. Пакеты принимаются вторым устройством ППС-40, из них извлекается поле, содержащее полезные данные, и они отправляются на ПК для последующей записи и воспроизведения.



Рис. 5. Общая схема передачи видео файла.

## **5. Алгоритм работы программы для приёма и передачи данных на персональном компьютере**

Приступим теперь к описанию того, как осуществляется передача файла с ПК на передающее устройство ППС-40, то есть рассмотрим, как работает на ПК программа, выполняющая эту функцию (название программы «ControlCenter»). Программа написана на языке С# в среде Microsoft Visual Studio Express.

Для удобства проведения экспериментов в меню Settings программы были вынесены некоторые параметры передачи: размер блока данных в поле СОМ port block size и паузу в миллисекундах между отправкой очередного блока данных в поле delay(ms). При каждом запуске программы настройки сбрасываются на стандартные значения: СОМ port block size – 60, delay(ms) –

10). Также в основном меню программы необходимо указывать директорию к файлу в поле Choose File. Передача файла начинается нажатием клавиши Send File.

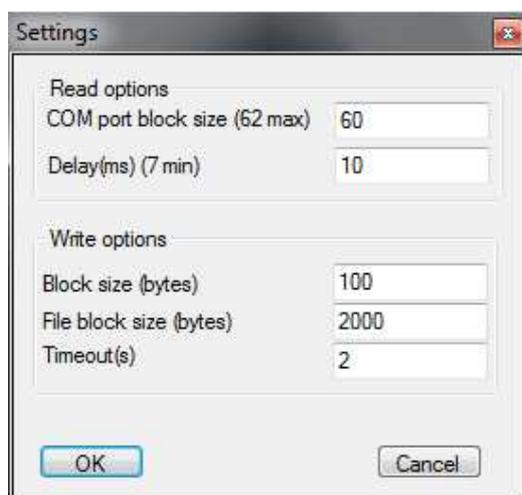


Рис. 6. Окно настройки программы «ControlCenter».

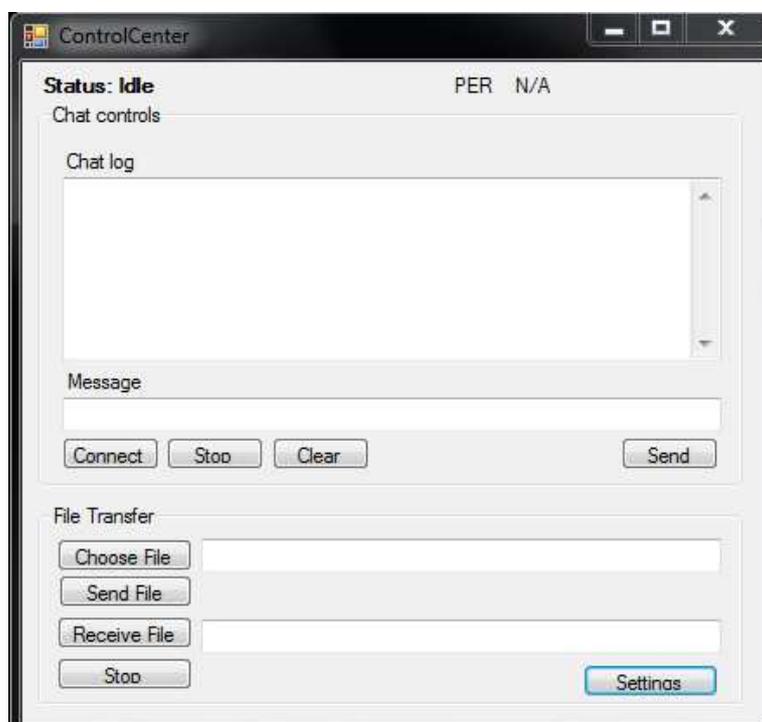


Рис. 7. Окно программы «ControlCenter».

Алгоритм работы программы представлен на рис. 8. Сначала происходит настройка основных параметров виртуального СОМ-порта, а именно:

- Скорость работы (115 200 кбит/с)
- Название виртуального СОМ-порта
- Размер буфера на приём и передачу данных (16 кбайт)

Далее, после выбора файла, который нужно передать, сохраняются и отправляются в СОМ-порт первые служебные данные: байт, содержащий в себе значение 11 (размер пакета), 2 байта, содержащие в себе значение 255 (это нужно для определения программой на приёме начала передачи файла) и 8 байт, содержащих размер передаваемого .avi файла. Затем программа входит в свой основной цикл работы - запись очередного блока данных из файла во временный массив и отправка новых данных из этого массива на СОМ-порт, если в его буфере на отправку содержится меньше данных, чем размер блока.

Перейдём к описанию получения файла на ПК от принимающего устройства ППС-40. Для этой цели на ПК также используется программа «ControlCenter».

При каждом включении программы автоматически выставляются стандартные настройки приёма данных: Block size, File block size и Timeout.

Block size – 100 байт (размер блока, поступающего во временный буфер СОМ-порта)

File block size – 2000 байт (размер блока данных, используемого для записи данных в файл из буфера)

Timeout – 2 секунды (если в течение этого времени в буфере СОМ-порта накопилось мало данных, то они дописываются в файл и приём прекращается).

В поле Receive File указывается название и формат принимаемого файла (например 1.avi).

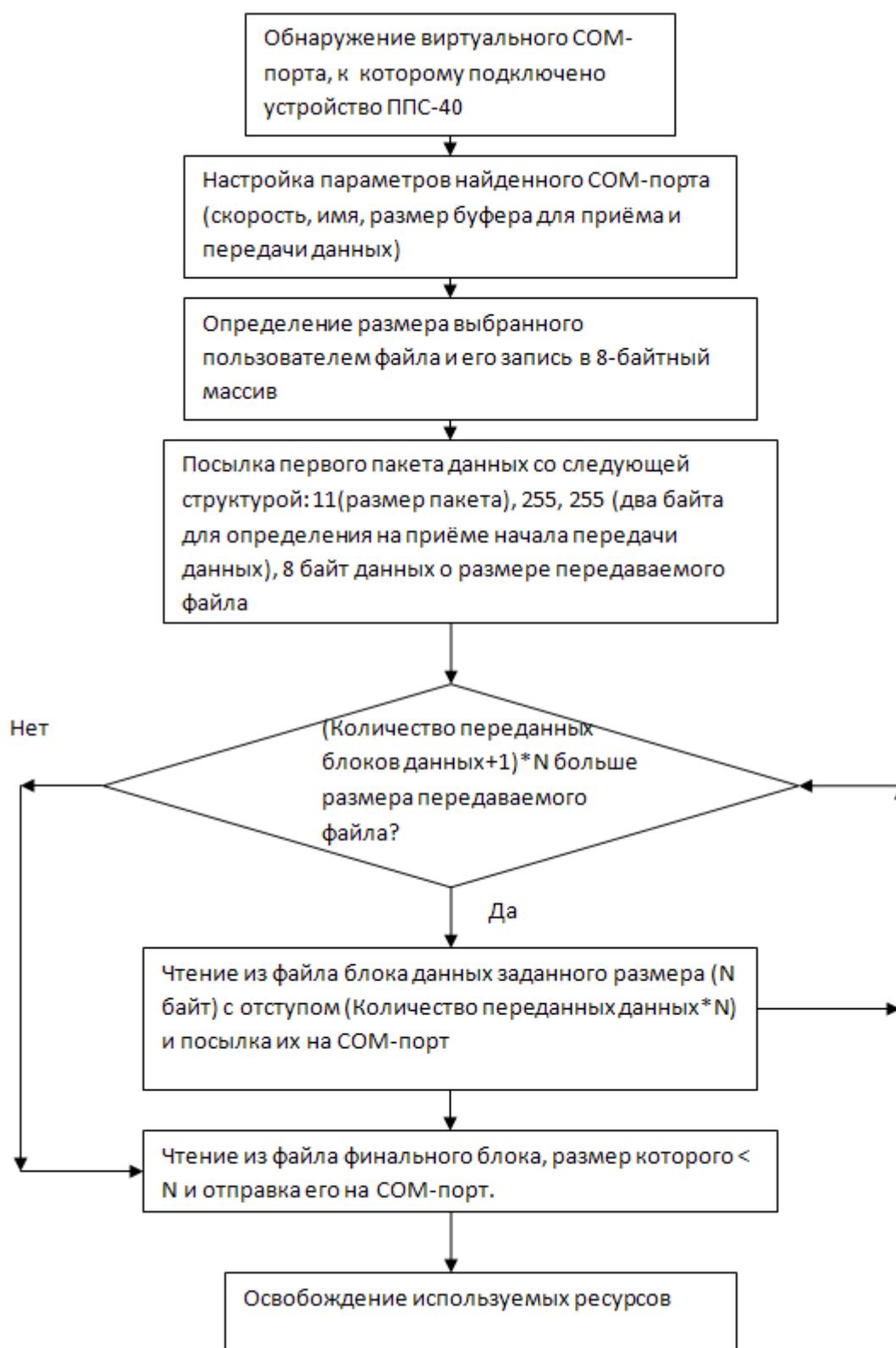


Рис. 8. Алгоритм работы программы по передаче файла.

Затем при нажатии на кнопку Receive File начинает выполняться алгоритм, представленный на рис. 9. Сначала происходит настройка основных параметров виртуального СОМ-порта аналогично случаю передачи файла. В случае если приходят 2 байта подряд со значением 255, считывается восьмибайтный размер файла и активируется таймер, причём так же сохраняется текущее время. Это нужно для критерия окончания приёма данных, который заключается в том, что если в течение определённого времени (обычно 2-3 секунды) в буфере СОМ-порта накопилось мало данных, то они дописываются в файл и приём прекращается.

После этого программа входит в цикл приёма данных, критерий выхода из которого был описан выше. Далее проверяется количество байт в буфере СОМ-порта, предназначенного для чтения. Если количество пришедших байт больше или равно размеру блока, то происходит запись нового блока данных в файл обновление таймера («отодвигается» ограничение по времени вперёд).

Если количество пришедших байт меньше размера блока, то происходит запись порции данных заданного размера в текущую позицию блока записи данных. Если данных в буфере СОМ-порта нет, то вступает в силу проверка по времени, и если оно не истекло, то идёт возвращение к более раннему этапу алгоритма (согласно рис. 9.). В случае, если лимит по времени исчерпан, происходит запись в файл всего, что накопилось к этому времени в буфере и приём файла прекращается.

Была проведена серия экспериментов по передаче видео файла (формат .avi, общий размер 452074 байт) при различных размерах блока данных. Размер блока данных указывался в настройках программы для передачи видео файла. На принимающем ПК запускалась программа «ControlCenter», и, после нажатия клавиши Receive File, программа начинала прослушивать СОМ-порт в соответствии с алгоритмом, описанным выше.

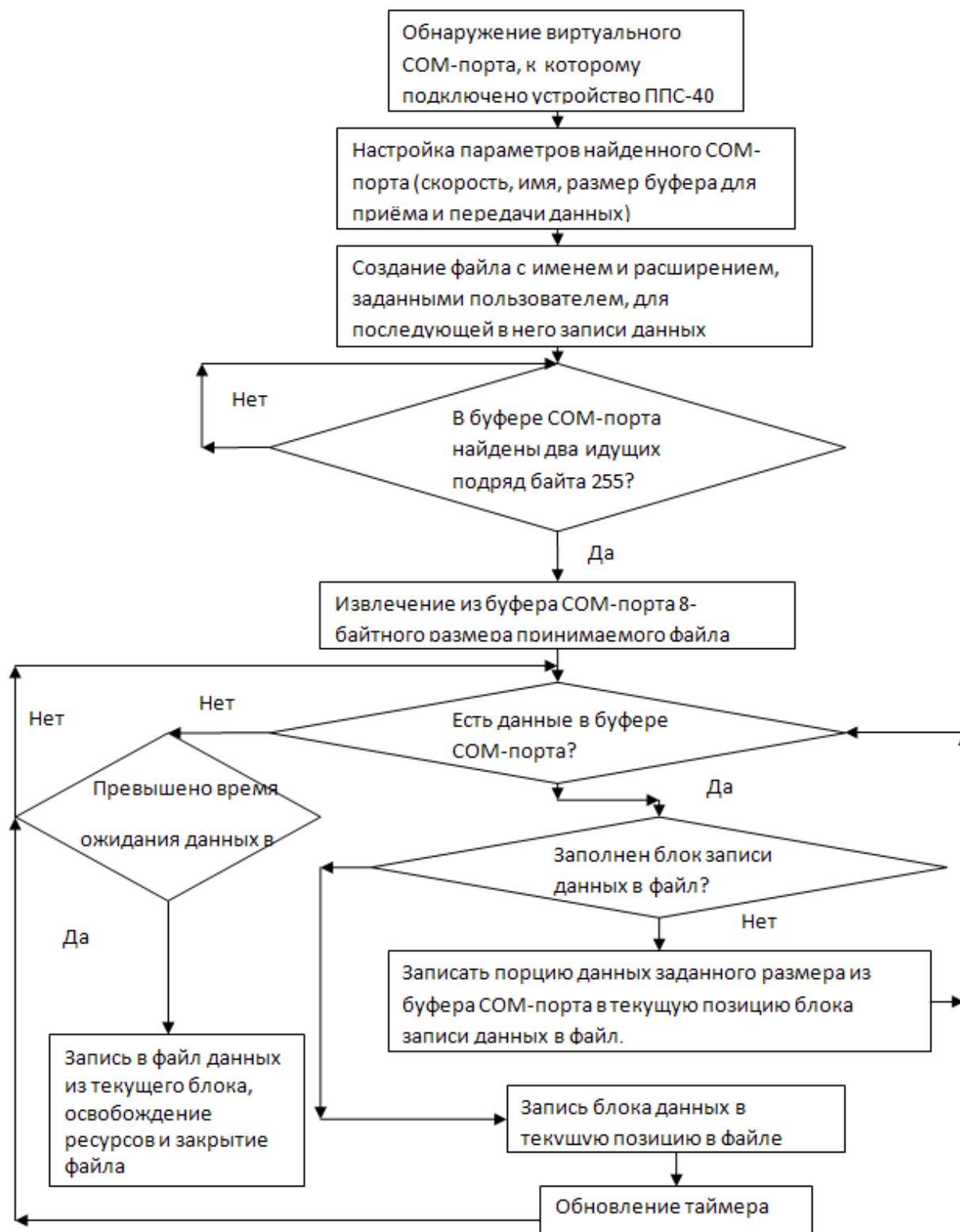


Рис. 9. Алгоритм работы программы при приёме файла.

После нажатия клавиши Send File в программе на передающем ПК, начиналась передача файла. В ходе экспериментов измерялось время, затраченное на передачу файла, и PER (Packet Error Rate). Ниже представлен график зависимости скорости передачи от размера пакета (рис. 10.). Большой вклад в скорость передачи делает пауза между пакетами. Если пересчитать скорость передачи без учета пауз, то мы получим зависимость, представленную на рис. 11.

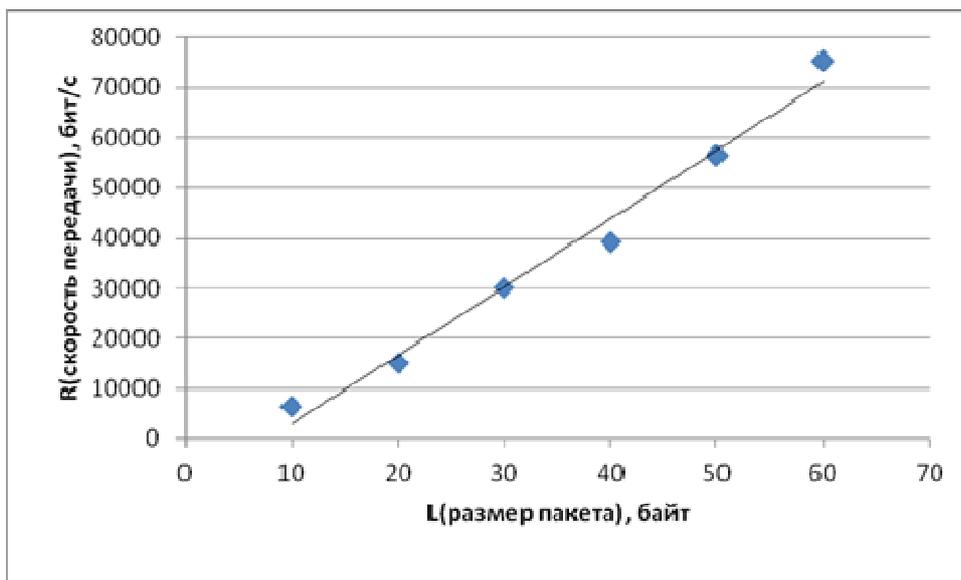


Рис. 10. Зависимость скорости передачи от размера пакета.

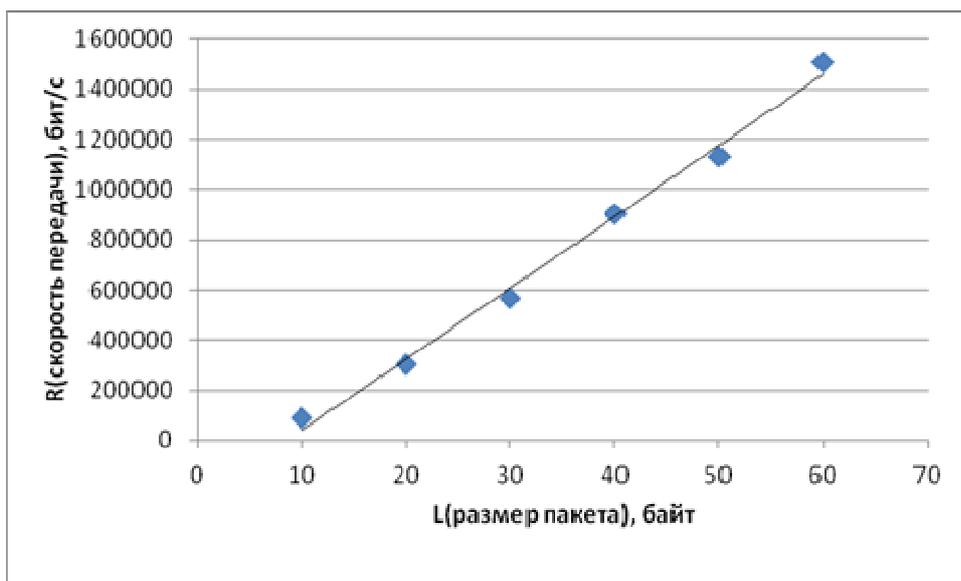


Рис. 11. Зависимость скорости передачи от размера пакета без учета пауз между пакетами.

Также исследовалась возможность передачи видео файла на различных расстояниях. Выяснилось, что на расстояниях больше 10 метров передача видео файлов невозможна, т.к. связь между устройствами обрывается.

На расстояниях больше 5м связь становится нестабильной, и некоторые пакеты могут придти с ошибками или вообще не дойти. Для отслеживания качества передачи измерялась величина PER (Packet Error Rate) по следующей формуле:

$$PER = \frac{S_1 - S_2}{S_1},$$

где  $S_1$ -определенный в начале передачи размер файла, а  $S_2$ -окончательный размер записываемого файла. В ходе экспериментов на расстояниях больше 5м величина PER менялась от 0 до величины порядка  $10^{-3}$ .

Для улучшения качества связи были введены перезапросы. После передачи каждого пакета, передатчик будет ждать сигнал от приёмника, подтверждающего доставку пакета, и только после получения сигнала будет передавать следующий пакет. В случае, если сигнала не было в течение заданного времени, передатчик будет заново отправлять тот же пакет данных. Это может повлечь за собой незначительное снижение скорости передачи (в экспериментах от 1% до 3% от скорости передачи без перезапросов), но обеспечит передачу данных без потерь.

В ходе экспериментов было показано, что представленная схема передачи данных, программное обеспечение на базе вышеуказанных алгоритмов и прямохаотические приёмопередатчики ППС-40 обеспечивают необходимое для передачи видео файла качество радиоканала. В случаях, когда  $PER=0$ , переданный видео файл успешно воспроизводился при помощи программы VLC media player.

## 6. Переход к другим конфигурациям сети и введение ретрансляции сигнала

До сих пор была представлена лишь простейшая конфигурация сенсорной сети, в которой передавались видеоданные, а именно – система точка-точка. Однако наибольший интерес представляют более сложные топологии сети, с которыми обычно и приходится иметь дело при работе с сенсорными сетями. При использовании сложных структур сенсорных сетей появляется возможность, например, для одновременной передачи видеоданных по нескольким радиоканалам. Что является принципиальным моментом, в таких сетях должна быть реализована передача данных между устройствами, расположенными далеко друг от друга (за пределами прямой радиовидимости), что влечёт за собой необходимость введения ретрансляции сигнала (рис. 12).

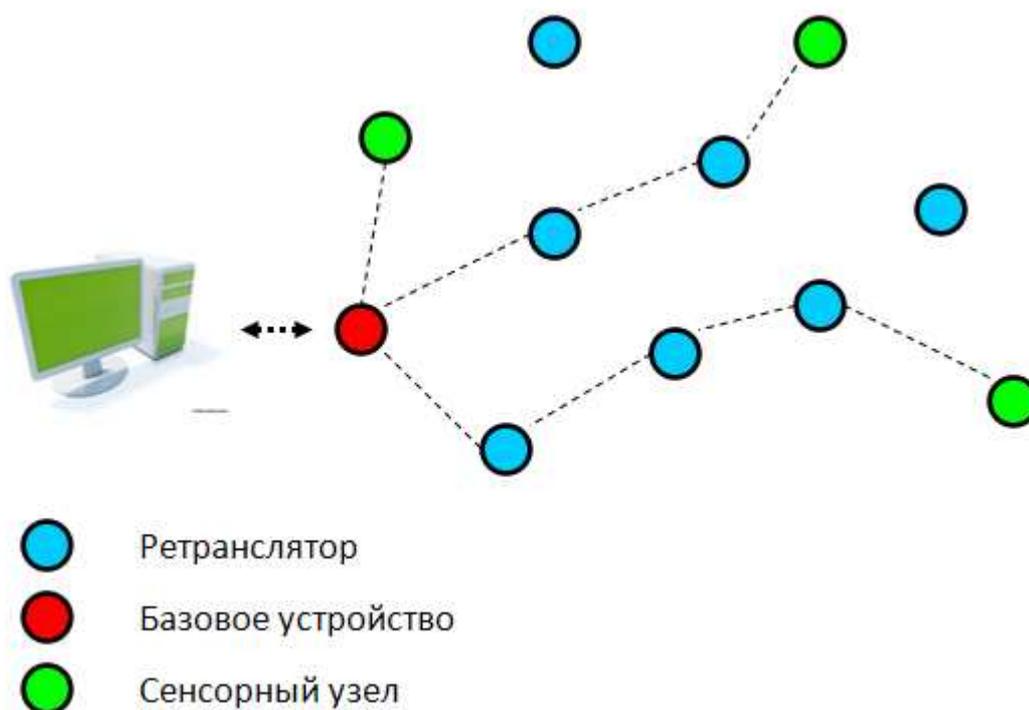


Рис. 12. Пример возможной топологии сенсорной сети

Итак, в этой части работы будет рассмотрено, как осуществляется ретрансляция сигнала в сенсорной сети. Это будет показано на системе передачи видео файла, представленной на рис. 13.

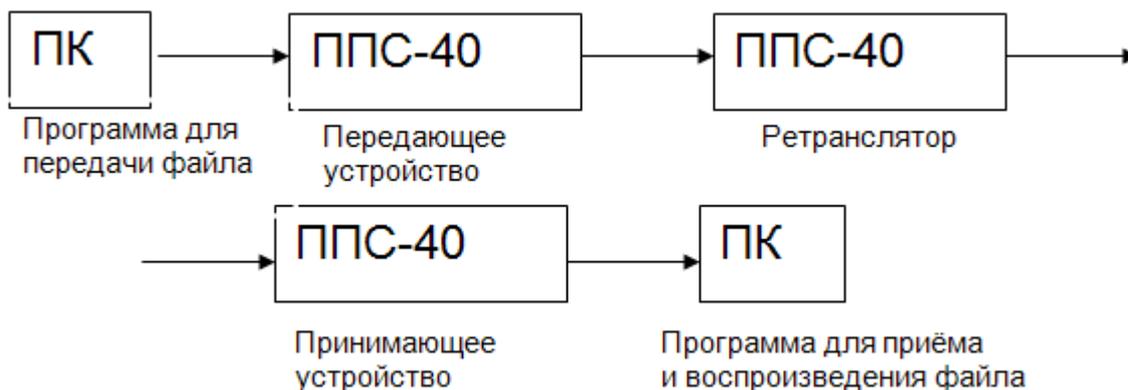


Рис. 13. Общая схема передачи видео файла с использованием ретрансляции

Программная часть для ПК, микроконтроллера передающего устройства и микроконтроллера принимающего устройства остаётся точно такой же, как и в предыдущем пункте. Отметим, что в программы заранее была встроена возможность проверки адреса отправителя данных, что важно при ретрансляции сигнала, так как в сенсорной сети с большим количеством устройств приниматься и ретранслироваться должны пакеты только от определённых отправителей данных.

Что касается алгоритма работы программы для микроконтроллера в ретрансляторе, то он представляет собой фактически алгоритм приёма данных базовым устройством, где после проверки адреса получателя идёт формирование пакета для ретрансляции с заменой адреса получателя согласно таблице маршрутизации, после чего очищается память ПЛИС и включается передатчик.

В соответствии со схемой, представленной на рис. 13 был проведён эксперимент по передаче видео файла в беспроводной сенсорной сети с использованием ретрансляции. Целью эксперимента является проверка того, что подобный канал связи предоставляет возможность для корректной

передачи видео данных. В устройства ППС-40 были прошиты соответствующие программы для сенсорного узла, ретранслятора и базового устройства. Маршрутизация осуществлялась следующим образом. У каждого устройства имеется свой адрес (состоит из двух байт данных). Устройством, получающим очередной пакет данных, проверяется адрес получателя. Если он совпадает, то пакет или ретранслируется (в случае, если речь идёт об устройстве-ретрансляторе), или отправляется на СОМ-порт ПК, к которому подключено данное устройство ППС-40 (в случае базовой станции).

В ходе экспериментов измерялось время, затраченное на передачу видео файла и PER. Было зафиксировано незначительное снижение скорости передачи (5-6% от скорости передачи без ретрансляции). Величина PER менялась от 0 до величины порядка 10<sup>-3</sup>. При PER=0 видео файл был успешно воспроизведён программой VLC media player.

## **7. Заключение**

По результатам экспериментов можно сказать, что рассмотренная система беспроводных сенсорных сетей позволяет обеспечить качественную передачу больших объемов данных. Дальнейшее развитие подобных систем позволит осуществлять передачу на больших расстояниях и скоростях. Для этого, в первую очередь, необходимо обратить внимание на используемый интерфейс UART, значительно ограничивающий скорость передачи. Также мы убедились, что какое бы сжатие мы не использовали, проблема с потерей пакетов может быть решена введением перезапросов, что позволит всегда поддерживать высокое качество передачи данных.

Работа выполнена при частичной финансовой поддержке ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007 - 2013 годы» (Гос. контракт № 11.519.11.4027) и гранта РФФИ 12-07-33106 мол\_a\_вед.

## Литература

[1] A. Redondi, M. Tagliasacchi, M. Cesana, L. Borsani, P. Tarrío, and F. Salice, "LAURA - LocAlization and Ubiquitous monitoRing of pAtients for health care support," in Advances in Positioning and LocationEnabled Communications, 2010. APLEC '10., IEEE International Workshop on, 2010

[2] А.С. Дмитриев, Б.Е. Кяргинский, А.И. Панас, С.О. Старков, "Прямохаотические схемы передачи информации в сверхвысокочастотном диапазоне", Радиотехника и электроника, 2001, Т. 46, № 2, 224-233.

[3] Дмитриев А.С., Клецов А.В., Лактюшкин А.М., Панас А.И., Синякин В.Ю. "Технологическая платформа для создания приемопередатчиков на основе хаотических сигналов", Успехи современной радиоэлектроники, 2008, №1, с. 77–83.

[4] ГОСТ 17657—79 «Передача данных. Термины и определения

[5] FCC News Release. Feb. 14. 2002. <http://www.fcc.gov/headlines2002.html>

[6] Дмитриев А.С., Клецов А.В., Лактюшкин А.М., Панас А.И., Старков С.О., Сверхширокополосная беспроводная связь на основе динамического хаоса, Радиотехника и электроника, 2006, Т. 51, №10, с. 1193-1209.

[7] Mitola J. III. // Cognitive Radio. An Integrated Agent Architecture for Software Defined Radio. Doctor of Technology Dissertation. Kista: Royal Institute of Technology, 2000. 313 p.