

РЕАЛИЗАЦИЯ КАНАЛА СВЯЗИ МЕЖДУ ДВУМЯ СМАРТ-УСТРОЙСТВАМИ С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ BLUETOOTH

А.С. Глебов

aleksey.glebov2010@yandex.ru

SPIN-код: 6291-7604

З.Р. Прохорова

zlata.oz@yandex.ru

SPIN-код: 3667-4011

МГТУ им. Н.Э. Баумана, Москва, Российская Федерация

Аннотация

Рассмотрена технология передачи сообщений посредством bluetooth-канала с использованием bluetooth-сокетов. Практическая значимость технологии Bluetooth обусловлена низким энергопотреблением адаптера и простотой ее использования, а также широтой распространения данного типа связи в настоящее время. После анализа технологии был разработан алгоритм обмена сообщениями между двумя конкретными устройствами (смарт-часы Samsung Gear S2). Показано, что данный алгоритм надежен с точки зрения гарантированности доставки информации и позволяет достичь уровня синхронизации, достаточного для большинства современных приложений. Задержка передачи сигнала между устройствами составляет порядка 50 мс, что в большинстве случаев является хорошим показателем.

Ключевые слова

Bluetooth, канал связи, сокет, обмен сообщениями, задержка, RFCOMM, протокол, сервер, клиент

Поступила в редакцию 28.02.2018

© МГТУ им. Н.Э. Баумана, 2018

Обзор используемой технологии передачи данных в сети Bluetooth. Обмен сообщениями. Bluetooth — технология беспроводной связи, обеспечивающая экономный (с точки зрения энергопотребления) и дешевый способ радиосвязи между электронными устройствами различных типов. Существует множество «слоев» спецификации Bluetooth, которые позволяют работать с данными различного типа [1], некоторые из них представлены на рис. 1.

Для передачи данных могут быть использованы асимметричный (721 Кб/с в одном направлении и 57,6 Кб/с в другом) и симметричный методы (432,6 Кб/с в обоих направлениях).

Алгоритм соединения bluetooth-устройств. Транспортный протокол RFCOMM [2] (см. рис. 1) реализует создание bluetooth-сокетов, которые работают аналогично сокетам потоков в интернет-протоколах [2–4].

Сокет (англ. *socket* — разъем) — название программного интерфейса для обеспечения обмена данными между процессами. Bluetooth-сокет в общем случае представляет собой набор функций, используемых для обмена данными между

двумя bluetooth-устройствами, причем каждое из устройств может играть роль как сервера (поставщик сервиса), так и клиента (пользователь сервиса).

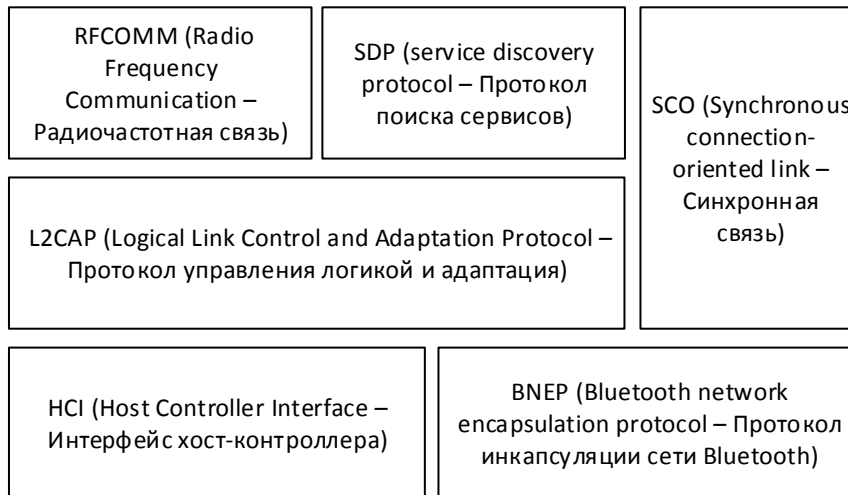


Рис. 1. Слои спецификации Bluetooth

При попытке открытия соединения с другим устройством выполняется поиск протокола обнаружения служб (SDP) и определяются протокол и канал, используемые для соединения. Если соединение установлено и сокет успешно открыт, возвращается экземпляр bluetooth-сокета с открытым состоянием. Затем этот сокет используется для обмена данными между подключенными устройствами.

Для установки соединения устройство-сервер регистрирует сервис и создает сокет для отслеживания и разрешения входящих запросов на соединения. В свою очередь, устройство-клиент подключается к сервису и ожидает соединения. После установки соединения оба устройства могут обмениваться данными. Когда обмен данными окончен, соединение между устройствами должно быть разорвано, а сервис должен быть закрыт [5].

Реализация алгоритма соединения bluetooth-устройств Gear S2. Для соединения двух устройств по bluetooth-каналу необходимо реализовать серверную и клиентскую части. Одно из устройств должно открыть серверный сокет, а второе — инициализировать соединение, используя MAC-адрес сервера. Сервер и клиент считаются соединенными, когда они оба имеют активный bluetooth-сокет на одном и том же RFCOMM-канале [6].

При соединении устройств одно из них должно вести себя как сервер, т. е. удерживать открытый bluetooth-сокет. Цель сервера — ждать запроса на входящее соединение, и когда оно подтверждено, создать bluetooth-сокет. Ниже описан процесс получения сокета с точки зрения сервера.

- С помощью вызова метода registerRFCOMMServiceByUUID() происходит регистрация сервиса и сервер получает bluetooth-сокет. Одним из параметров вызываемого метода является идентификационное имя сервиса. Система авто-

матически добавит его в базу протокола обнаружения служб (SDP). В качестве параметров метода также указывают стандарт UUID сервера, который используется клиентом для установления соединения.

- Начинается прослушивание запроса на соединение через метод `onconnect()`. Это блокирующий метод, который возвращает результат, когда соединение подтверждено, либо когда сгенерировано исключение. Соединение считается подтвержденным, если удаленное устройство пошлет запрос на соединение с UUID, указанным при регистрации серверного сокета. В случае успеха метод `onconnect()` возвращает настроенный на соединение сокет.

- Если возникает необходимость принять дополнительное соединение, вызывается метод `close()`. Это приведет к освобождению сокета и всех его ресурсов, но не закроет соединенный сокет. В отличие от TCP/IP, протокол RFCOMM позволяет работать только с одним клиентом на канале, поэтому в большинстве случаев имеет смысл вызывать метод `close()` сразу же после принятия сокета.

- Для инициализации соединения с удаленным серверным устройством устройству-клиенту необходимо получить объект `BluetoothDevice`, содержащий информацию о сервере. Процесс получения сокета клиентом происходит следующим образом:

- с помощью метода `connectToServiceByUUID()` инициализируется подключение к серверу и происходит запрос на создание сокета на стороне клиента. Значение параметра `UUID` должно совпадать со значением, указанным сервером при регистрации сервиса;

- вызов метода `onconnect()` происходит при установлении соединения. После вызова этого метода система будет выполнять SDP поиск на удаленном устройстве, чтобы сопоставить UUID. В случае успеха, при условии подтверждения запроса со стороны сервера, будет открыт RFCOMM канал. Это блокирующий вызов. Если по каким-то причинам соединение сорвется или произойдет ошибка времени исполнения, то будет сгенерировано исключение.

После успешного соединения каждое из соединенных устройств имеет объект `BluetoothSocket`, с помощью которого можно реализовать передачу/прием данных. Чтение и запись данных осуществляются с помощью методов `readData()` и `writeData()` соответственно [6–9].

Проблемы и решения. Анализ задержек. Основной проблемой, возникшей при реализации описываемой технологии в рамках конкретного проекта, является задержка при передаче сообщений.

Поскольку передача по протоколу RFCOMM реализована аналогично передаче по протоколу TCP, проблема потери сообщений не возникает [10], но увеличивается время доставки сообщений. Для решения данной проблемы было принято решение уменьшить размер передаваемого сообщения: в каждом сообщении передавать только начальные данные для каждой точки, а затем осуществлять одинаковую обработку на каждом устройстве. Такой подход позволил добиться минимальной задержки передачи сообщений. В самом худшем случае происходила рассинхронизация изображения на один кадр, что можно увидеть на рис. 2.



Рис. 2. Пример рассинхронизации устройств

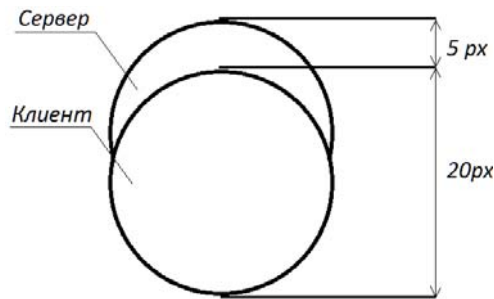


Рис. 3. Положение изображений на сервере и клиенте

Анализ ситуации, изображенной на рис. 2, позволяет оценить задержку между устройствами. При наложении изображений друг на друга получается картина, схематично представленная на рис. 3.

Каждое изображение имеет диаметр 20 пикселей. При этом разница в положении изображений на устройстве-сервере и устройстве-клиенте равна примерно четверти от

размера самого изображения, т. е. 5 пикселей. По результатам анализа исходного кода был сделан вывод, что изменение положения изображения по осям X и Y происходит таким образом, чтобы суммарный путь, пройденный изображением, был равен 5 пикселям (гипотенуза). Именно столько составляет разница между положениями изображений на двух устройствах. Поскольку данная функция вызывается 1 раз в 50 мс, можно сделать оценку задержки между двумя устройствами, которая также будет составлять порядка 50 мс.

Другой способ оценки задержки основывается на пропускной способности канала связи. В данном случае используется симметричный способ передачи, который, как было сказано ранее, позволяет передавать данные в обоих направлениях со скоростью до 432,6 Кбит/с:

$$432,6 \text{ Кб/с} = 54,075 \text{ Кб/с} = 55372,8 \text{ б/с.} \quad (1)$$

Из выражения (1) видно, что по такому каналу можно передать примерно 55,4 б информации за 1 мс. Было установлено, что средний размер передаваемого через сокет сообщения составляет 2,5 Кб, исходя из чего можно оценить время, необходимое для передачи, т. е. время задержки передачи сигнала между двумя устройствами:

$$\frac{2500 \text{ б}}{55,4 \text{ б/мс}} = 45,1 \text{ мс.} \quad (2)$$

Полученная оценка задержки (2) практически совпадает с предыдущей и составляет около 50 мс.

Заключение. Разработан работоспособный алгоритм передачи сообщений между двумя устройствами Samsung Gear S2 по bluetooth-каналу по протоколу RFCOMM с использованием сокетов. Данный алгоритм был сразу же применен при создании игры с режимом мультиплеера для данных устройств.

Литература

- [1] Weeks R., Dumbill E., Jerson B. *Linux unwired*. O'Reilly, 2004, 312 p.
- [2] RFCOMM Protocol. URL: https://www.amd.etechnik.uni-rostock.de/ma/gol/lectures/wirlec/bluetooth_info/rfcomm.html (дата обращения 09.10.2017).
- [3] Таненбаум Э., Уэзеролл Д. *Компьютерные сети*. Санкт-Петербург, Питер, 2012, 960 с.
- [4] Бройдо В.Л., Ильина О.П. *Вычислительные системы, сети и телекоммуникации*. Санкт-Петербург, Питер, 2011, 560 с.
- [5] Орлов С.А., Цилькер Б.Я. *Организация ЭВМ и систем*. Санкт-Петербург, Питер, 2011, 688 с.
- [6] Официальный сайт Bluetooth. URL: <https://www.bluetooth.com/> (дата обращения 09.10.2017).
- [7] Tizen APIs. URL: <https://developer.tizen.org/node/18032> (дата обращения 09.10.2017).
- [8] Bluetooth. URL: <http://www.tadviser.ru/index.php/Статья:Bluetooth> (дата обращения 08.10.2017).
- [9] Татарников О. Bluetooth и безопасность. URL: <http://compress.ru/article.aspx?id=10807> (дата обращения 09.10.2017).
- [10] Агафонов Н. Технологии беспроводной передачи ZidBee, BlueTooth, Wi-Fi. Беспроводные технологии, 2016, № 1. URL: http://wireless-e.ru/articles/bluetooth/2006_1_10.php.

Глебов Алексей Сергеевич — студент кафедры «Компьютерные системы и сети», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Прохорова Злата Романовна — студентка кафедры «Компьютерные системы и сети», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Научный руководитель — Хохлов Сергей Александрович, ассистент кафедры «Компьютерные системы и сети», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

IMPLEMENTING THE COMMUNICATION CHANNEL BETWEEN TWO SMART DEVICES BY APPLYING THE BLUETOOTH TECHNOLOGY

A.S. Glebov

aleksey.glebov2010@yandex.ru

SPIN-code: 6291-7604

Z.R. Prokhorova

zlata.oz@yandex.ru

SPIN-code: 3667-4011

Bauman Moscow State Technical University, Moscow, Russian Federation

Abstract

The article considers the messaging technology by means of Bluetooth-channel using Bluetooth-sockets. The practical significance of the Bluetooth technology is contingent on low power consumption of the adapter and its easy handling as well as the current wide-spread use of this communication type. Having analyzed the technology we developed an algorithm for message exchange between two particular devices (smart watch SamsungGear S2). It is shown that this algorithm is reliable with regard to ensuring the delivery of information and allows reaching the point of synchronization sufficient for most modern applications. The signals transmission delay between the devices is approximately 50 ms, which is in most cases a good factor.

Keywords

Bluetooth, communication channel, socket, message exchange, delay, RFCOMM, protocol, server, client

© Bauman Moscow State Technical University, 2018

References

- [1] Weeks R., Dumbill E., Jerson B. Linux unwired. O'Reilly, 2004, 312 p.
- [2] RFCOMM Protocol. Available at: https://www.amd.technik.uni-rostock.de/ma/gol/lectures/wirlec/bluetooth_info/rfcomm.html (accessed 09.10.2017).
- [3] Tanenbaum A.S., Wetherall D.J. Computer networks. Prentice Hall, 2010, 960 p. (Russ. ed.: Komp'yuternye seti. Sankt-Peterburg, Piter publ. publ., 2012, 960 p.)
- [4] Broydo V.L., Il'ina O.P. Vychislitel'nye sistemy, seti i telekommunikatsii [Computer systems, networks and telecommunication]. Sankt-Peterburg, Piter publ., 2011, 560 p.
- [5] Orlov S.A., Tsil'ker B.Ya. Organizatsiya EVM i system [Computer and system organization]. Sankt-Peterburg, Piter publ., 2011, 688 p.
- [6] Bluetooth official website. Available at: <https://www.bluetooth.com/> (accessed 09 October 2017).
- [7] Tizen APIs. Available at: <https://developer.tizen.org/node/18032> (accessed 09 October 2017).
- [8] Bluetooth. Available at: <http://www.tadviser.ru/index.php/Stat'ya:Bluetooth> (accessed 08 October 2017).
- [9] Tatarnikov O. Bluetooth i bezopasnost' [Bluetooth and security]. Available at: <http://compress.ru/article.aspx?id=10807> (accessed 09 October 2017).
- [10] Agafonov N. Wireless data transmission technologies: ZidBee, BlueTooth, Wi-Fi. *Be-sprovodnye tekhnologii* [Wireless Technologies], 2016, no. 1. Available at: http://wirelesse.ru/articles/bluetooth/2006_1_10.php.

Implementing the communication channel between two smart devices by applying ...

Glebov A.S. — student, Department of Computer Systems and Networks, Bauman Moscow State Technical University, Moscow, Russian Federation.

Prokhorova Z.R. — student, Department of Computer Systems and Networks, Bauman Moscow State Technical University, Moscow, Russian Federation.

Scientific advisor — S.A. Khokhlov, Assistant, Department of Computer Systems and Networks, Bauman Moscow State Technical University, Moscow, Russian Federation.