

УДК 004.855

DOI: 10.25559/SITITO.15.201903.733-745

## Исследование энергосберегающей беспроводной самоорганизующейся многопротокольной сети передачи данных интернет устройств

М. В. Ушакова, Ю. А. Ушаков\*

Оренбургский государственный университет, г. Оренбург, Россия  
460000, Россия, г. Оренбург, пр. Победы, д. 13

\*unpk@mail.ru

### Аннотация

Современный уровень развития беспроводных устройств позволяет использовать дешевый беспроводной интерфейс практически в любых областях электроники, где раньше использовались преимущественно проводные решения. Большинство устройств в последнее время – это беспроводные, мобильные или носимые устройства различной степени автономности. Большое количество сооружений, зданий и частных объектов оборудованы беспроводными системами охраны, пожарной сигнализации, беспроводными датчиками учета воды, электроэнергии, газа и т.д. Многие объекты дополнительно используют множество беспроводных устройств контроля и управления объектами, беспроводные реле, замки, видеонаблюдение. Многие устройства имеют несколько независимых каналов связи, например, Wi-Fi и Bluetooth, GSM/LTE, ZigBee, 6LoWPAN или NFC. Некоторые из этих способов связи несовместимы с другими, не используют IP адреса или имеют другие ограничения. Разработка методов и алгоритмов функционирования сетей беспроводных устройств с разными типами интерфейсов с единообразным механизмом взаимодействия и управления, а также их энергопотребление ставится целью этой работы. Рассмотрены протоколы функционирования Интернет-вещей, выбраны для исследования самые доступные и популярные аппаратные решения. Поскольку стоит задача построения произвольной сети, были рассмотрены методы реализации самоорганизующихся сетей на основе выбранных аппаратных устройств и протоколов. Основной задачей экспериментального исследования стало исследование энергопотребления устройств при использовании выбранных сетевых протоколов и методов, особенно с учетом того, что производители часто не акцентируют внимание на накладных расходах при включении устройств и модулей, предварительном согласовании подключений, которые могут быть больше потребления при полезной работе.



Контент доступен под лицензией Creative Commons Attribution 4.0 License.  
The content is available under Creative Commons Attribution 4.0 License.



**Ключевые слова:** Интернет вещей, беспроводные сети, Wi-Fi, Bluetooth, самоорганизующиеся сети, энергопотребление.

**Финансирование:** исследование выполнено при финансовой поддержке Правительства Оренбургской области и Российского фонда фундаментальных исследований в рамках научных проектов № 18-07-01446\_а «Разработка методов оптимизации систем размещения вычислительных виртуальных элементов в облачных инфраструктурах при работе с большими данными», № 16-29-09639\_офи\_м «Адаптивная система выявления угроз и защиты от кибератак в крупных корпоративных сетях», № 18-37-00485\_мол\_а «Разработка алгоритмов и моделей функционирования распределенных многопротокольных сетей беспроводных устройств с использованием виртуализации сетевых функций».

**Для цитирования:** Ушакова М. В., Ушаков Ю. А. Исследование энергосберегающей беспроводной самоорганизующейся многопротокольной сети передачи данных интернет устройств // Современные информационные технологии и ИТ-образование. 2019. Т. 15, № 3. С. 733-745. DOI: 10.25559/SITITO.15.201903.733-745

© Ушакова М. В., Ушаков Ю. А., 2019



## Research of Energy-Saving Wireless Self-Organizing Wireless Multi-Protocol Internet Device Data Transmission Network

M. V. Ushakova, Yu. A. Ushakov\*

Orenburg State University, Orenburg, Russia

13 Pobeda Av., Orenburg 460000, Russia

\*unpk@mail.ru

### Abstract

The current level of development of wireless devices allows you to use a cheap wireless interface in almost any field of electronics, where previously used mainly wired solutions. Most devices recently are wireless, mobile, or wearable devices of varying degrees of autonomy. A large number of structures, buildings and private facilities are equipped with wireless security systems, fire alarms, wireless sensors for metering water, electricity, gas, etc. Many objects additionally use many wireless devices for monitoring and controlling objects, wireless relays, locks, video surveillance. Many devices have several independent communication channels, for example, Wi-Fi and Bluetooth, GSM / LTE, ZigBee, 6LoWPAN or NFC. Some of these communication methods are incompatible with others, do not use IP addresses or have other restrictions. The development of methods and algorithms for the functioning of networks of wireless devices with different types of interfaces with a uniform interaction and control mechanism, as well as their energy consumption, is the goal of this work. The protocols of the functioning of Internet of things are considered, hardware solutions are selected the most affordable and popular for research. Since the goal is to build an arbitrary network, we considered methods for implementing self-organizing networks based on selected hardware devices and protocols. The main objective of the experimental study was to study the energy consumption of devices using the selected network protocols and methods, especially given that manufacturers often do not focus on the overhead when turning on devices and modules, preliminary coordination of connections, which can be more than consumption during useful work.

**Keywords:** Internet of things, wireless networks, Wi-Fi, Bluetooth, self-organized networks, power consumption.

**Funding:** The study was carried out with financial support from the Government of the Orenburg Region and the Russian Foundation for Basic Research as a part of the scientific project № 18-07-01446\_a "Development of Methods for Optimizing the Placement of Computing Virtual Elements in Cloud Infrastructures when Working with Big Data", № 16-29-09639\_офи\_м "Adaptive System for Threat Detection and Protection Against Cyber Attacks in Large Corporate Networks", № 18-37-00485\_мол\_а "Development of Algorithms and Operational Models of Distributed Multi-Protocol Networks of Wireless Devices Using Virtualization of Network Functions".

**For citation:** Ushakova M.V., Ushakov Yu.A. Research of Energy-saving Wireless Self-organizing Wireless Multi-protocol Internet Device Data Transmission Network. *Sovremennye informacionnye tehnologii i IT-obrazovanie* = Modern Information Technologies and IT-Education. 2019; 15(3):733-745. DOI: 10.25559/SITITO.15.201903.733-745



## Введение

В настоящее время автоматизация и системы связи проникли во все области жизни и без стабильной связи некоторые привычные уже задачи становятся недоступны. Системы Internet of Things (IoT, Интернет вещей) – это программные и аппаратные средства, соединяющие встроенные устройства для взаимодействия друг с другом и внешней средой. Обычно IoT это маломощные беспроводные устройства, которые могут подключаться к стеку TCP/IP и выходить в Интернет. Они предоставляющие такие возможности взаимодействия, как сбор, анализ и последующая обработка данных, управление исполнительными механизмами (реле, модулированные цепи, внешние линии связи) и передача данных дальше по каким либо промышленным, нестандартным или широко известным протоколам. Системы IoT могут использоваться для измерительных или контрольных датчиков в промышленности или домашних хозяйствах. Они также могут применяться для отслеживания процессов, управления техническими или промышленными средствами. Использование систем IoT получает все большее распространение, потому что благодаря им получение и обработка данных становится проще и эффективнее. Тем не менее, при современном уровне развития IoT обладает некоторыми недостатками. Одной из самых больших проблем является различие существующих стандартов связи данных устройств в зависимости от производителя.

Современные системы IoT можно разделить на два направления: «Промышленный Интернет вещей» (Industrial Internet of Things, IIoT) и «Потребительский Интернет вещей» (Consumer Internet of Things) для домашнего использования. Интернет вещей (IoT) нацелен на быт, открывая безграничные возможности по экономии и безопасности. Промышленный интернет вещей, как и следует из его названия, нацелен на производственные выгоды и позволяет увеличить рентабельность бизнеса [1]. IoT - это планомерное развитие технологии «smart home» (умный дом), в то время как IIoT – это улучшение существующих производственных телеметрических и телематических систем. Решения, построенные на базе IoT, являются низкими по стоимости. Обычно для таких решений характерно наличие малого количества датчиков. IIoT-решения имеют высокую стоимость, что обусловлено, в первую очередь, большим количеством устройств, требуемых для установки на производстве. Кроме того, IIoT-устройства являются дорогостоящими, так как к ним выставляются высокие требования в плане надёжности, безопасности и эффективности работы в различных условиях. Обычно IoT-устройства генерируют трафик, не превышающий 2-3 Мб в день. IIoT-устройства могут генерировать большие объемы трафика, до Пб (петабайт) в день.

## Цель исследования

Целью работы является исследование методов и алгоритмов функционирования сетей беспроводных устройств с разными типами интерфейсов. Основными задачами являются исследование и разработка единообразных механизмов взаимодействия и управления, исследование и разработка механизмов маршрутизации, а также изучение энергопотребления в различных режимах при передаче данных. Основной проблемой исследования является выделение приоритетных

технологий и устройств для изучения, постановка задачи оптимизации для маршрутизации в условиях энергосбережения, изучения сложной структуры энергопотребления устройств в различных режимах работы.

## Технологии создания беспроводных сетей IoT

Существует довольно много технологий беспроводного соединения. Их основными характеристиками являются расстояние связи, полоса пропускания и потребляемая мощность. Рассмотрим основные технологии, используемые сейчас в IoT среде.

Wi-Fi (Wireless Fidelity) на сегодняшний день является самым распространенным стандартом, используемым как в домашних, так промышленных IoT системах [2]. Стандарт Wi-Fi IEEE 802.11ax предоставляет техническую возможность передачи данных до 11 Гбит/с суммарно по всем каналам в диапазоне 60ГГц, 802.11ac - суммарно по всем каналам до 6.7 Гбит/с в диапазоне 5ГГц, а наиболее часто встречающийся в IoT 802.11n в частотном диапазоне 2.4ГГц – до 450 Мбит/с с технологией MIMO и 75-150 Мбит/с для малых устройств. Однако в приложениях IoT требуется в первую очередь не скорость, а надежность и высокая плотность, что стандартными методами в Wi-Fi достичь сложно. Основными недостатками являются высокое энергопотребление, малый радиус действия, низкая помехоустойчивость, низкая допустимая плотность клиентов, недоступность устройств IoT для диапазонов 5 и 60 ГГц, относительно высокая стоимость (по сравнению с Bluetooth и ZigBee).

Многие IoT используют Bluetooth LowEnergy (BLE) – особый тип Bluetooth протокола. BLE используется в основном там, где нет необходимости передавать большой объем данных. Кроме того, устройству не требуется большая встроенная батарея для связи с конечным устройством. Bluetooth Low Energy потребляет меньше энергии, чем классический Bluetooth, умеет передавать сразу нескольким устройствам, допускает создание частично-связанных сетей и широкоэвещательную рассылку. Это достигается путем отправки данных по мере необходимости с заранее определенными периодическими обновлениями. Однако BLE на самом деле не предназначен для передачи файлов и больше подходит для небольших объемов данных. Основным недостатком является то, что нам нужно какое-то промежуточное (мостовое) устройство для отправки данных в сеть Ethernet/Wi-Fi [3]. К достоинствам стандарта относятся высокий уровень стандартизации и совместимость между устройствами Bluetooth разных производителей, низкая стоимость, универсальность и большое разнообразие модулей под разные задачи.

Специально разработанный для IoT подобных устройств протокол 802.11ah (HaLow) использует полосу частот в диапазоне 900 МГц (конкретная частота будет зависеть от страны и региона). Использование предопределенных периодов пробуждения и активности оптимизирует энергопотребление и повышает возможную плотность устройств (до 8000 на одну точку доступа), радиус действия более 1500 м. Стандарт 802.11ah также поддерживает ячеистые сети mesh-сет [4]. Однако для 802.11ah требуются специализированные точки доступа и оборудование, которые на сегодняшний день на рынке представлены слабо.



Технология ZigBee наиболее распространена в промышленной среде и в брендовых устройствах IoT [5]. ZigBee PRO и ZigBee Remote Control (RF4CE) на основе IEEE802.15.4 - технологии беспроводной промышленной сети, работают на частотах 2,4 ГГц и 915 МГц, предназначены для передачи относительно небольшого количества данных при низких скоростях передачи. Максимальная скорость передачи данных в диапазоне 2,4 ГГц не превышает 250 Кбит/с (в других диапазонах она ниже), а мощность устройств не должна быть больше 10 мВт на открытой территории и 100 мВт в зданиях. В этой технологии реализована поддержка ячеистой топологии mesh, спящих и мобильных узлов, а также узлов, которые обеспечивают работу алгоритмов ретрансляции и самовосстановления. Стандарт ZigBee предусматривает криптографическую защиту данных, передаваемых по беспроводным каналам и разграничение доступа, но реализация этих механизмов часто несовместима между производителями.

Технология Z-Wave во многом похожа на ZigBee, она использует схожие механизмы шифрования данных и работает по принципу ячеистой сети mesh [6]. При этом ZigBee вместительнее (может включить более 60000 устройств, у Z-Wave около 230) и быстрее (40-250 Кбит/с против 10-100 Кбит/с у Z-Wave). С другой стороны, сигналы ZigBee слабее (около 10 м против 30 м у Z-Wave), и не все устройства ZigBee работают друг с другом из-за особенностей реализации конкретных производителей. Однако устройства Z-Wave зависимы от страны в плане частот работы, в различных поддиапазонах 800 и 900 МГц.

LoRaWAN (Lora Network) является наиболее открытым стандартом для создания сети IoT [7]. К преимуществам LoRa можно отнести открытость стандарта, большую дальность, высокую проникающую способность, низкое энергопотребление, различные нелицензируемые частоты (109 МГц, 433 МГц, 868 МГц, 915 МГц), адаптивную скорость передачи данных 0,3-50 Кбит/с. Недостатками являются отсутствие роуминга, несо-

вместимость и несовместимость устройств разных производителей.

Протокол Thread, реализованный в качестве дополнения к Wi-Fi [8]. Протокол основан на стандарте 6LoWPAN (IPv6 over Low power Wireless Personal Area Networks) — реализации протокола IPv6 поверх маломощных беспроводных сетей стандарта IEEE802.15.4. Протокол обеспечивает шифрование AES ячеистую топологию mesh, плотность - до 250 устройств на точку доступа. В технологии наиболее полно описаны спящие режимы, достигающие энергосбережения, сравнимого с LoRaWAN. Недостатки Thread заключаются в низкой помехоустойчивости, малой проникающей способности, зависимости от страны в плане частот работы за пределами зданий.

В будущем производители надеются на сети 5G в плане универсальной платформы для IoT [9]. В настоящее время есть множество устройств, использующих сотовый канал связи для связи с сервисами в Интернет, но 5G позволит использовать сеть как инфраструктуру для mesh и размещать виртуальные службы в NFV (network function virtualization) сегментах для переноса функциональности с IoT устройств в граничные вычисления (Edge-computing). Также сотовые сети позволяют использовать множество политик энергосбережения, имеют широкий переключаемый диапазон скоростей, частот и технологий доступа. Недостатком использования сотовых сетей является привязка к поставщику, плохое покрытие в труднодоступных местах, низкая проникающая способность на промышленных объектах.

Дополнение к LTE (Long-Term Evolution) существует стандарт NB-IoT для устройств с низким потреблением [10, 11]. Особенности стандарта - работа при более низких уровнях сигнала и при высоком уровне шумов, чем LTE, экономия батареи, полудуплексный режим, скорость до 100Кбит/с.

В таблице 1 приведены основные характеристики описанных выше технологий.

Таблица 1. Основные характеристики технологий для встраиваемых модулей  
Table 1. Key technology specifications for embedded modules

Технология	Стандарт	Частотный диапазон	Скорость передачи данных	Дальность действия с внешней антенной	Энергопотребление модуля при передаче	Топология
Wi-Fi	IEEE 802.11	2.4/5/60 ГГц	Более 300 Мбит/с	до 100 м	Высокое Около 250мА на канал	«Звезда», иногда ячеистая сеть, используемая для расширения диапазона
BLE	Bluetooth 5.0	2.4 ГГц	до 3 Мбит/с	>100 м	Низкое BLE-13мА, v4.0- 40мА	«Точка-точка», ячеистая сеть
HaLow	IEEE 802.11ah	900 МГц	50 кбит/с- 18Мбит/с	> 1 км	Пониженное 120мА	Ячеистая сеть
ZigBee	IEEE 802.15.4	915 МГц /2.4 ГГц	250 кбит/с	100 м	Низкое Около 50мА	Ячеистая сеть
Z-Wave	Z-Wave	433/800/900 МГц	10-100 кбит/с	30 м	Низкое 2.5 мА	Ячеистая сеть
Lora	LoRaWAN	менее 1 ГГц	0,3-50 Кбит/с	2-5 км в городе, до 45 км вне города	Низкое 45 мА	Звезда
Thread	IEEE 802.15.4	2.4 ГГц	250 кбит/с	30 м	Низкое	Ячеистая сеть
LTE	LTE	Все сотовые диапазоны	До 300 Мбит/с	От 3 до 20 км, в зависимости от частоты	До 450мА	Звезда
NB-IoT	NB-IoT	800/ 900/1800МГц	До 100Кбит/с	От 13 до 20 км	До 220 мА	Звезда



Хаб IoT (шлюз, gateway) — это устройство, преобразующее сигналы от конечных устройств (датчиков) в Wi-Fi. Многие производители конечных устройств выпускают собственные хабы, не совместимые с устройствами других производителей. Универсальные хабы не получили особого развития [12]. Кроме этого, существует возможность организации открытого хаба на базе любого микрокомпьютера с нужными интерфейсами. Это позволяет управлять Wi-Fi, Bluetooth, ZigBee устройствами, соединять устройства разных производителей в одном месте. Набор функций у открытых хабов примерно одинаков — подключение устройств через веб-интерфейс, интеграция с внешними сервисами типа IFTTT/Zapier, создание сценариев автоматизации и т.д. [13].

При установке конечного оборудования вероятно возникновение ситуации, когда разные устройства будут поддерживать разные протоколы IoT - Wi-Fi, ZigBee, Bluetooth, IEEE 802.15.4 (в том числе 6LoWPAN), LoRa. Однако каждый протокол имеет свои реализации, несовместимые с другими протоколами. Чтобы избежать проблемы с совместимостью устройств, работающих на разных протоколах, предпочтительнее использовать устройства, поддерживающие возможность использования нескольких протоколов одновременно.

В данный момент почти все широко доступные для самостоятельной разработки модули IoT поддерживают либо Wi-Fi, либо Bluetooth, иногда оба сразу. Одно из основных преимуществ стандарта Bluetooth заключается в его высоком уровне стандартизации и широчайшем распространении в составе пользовательских электронных устройств. Это позволяет в ряде случаев существенно сэкономить время и затраты на разработку при проектировании систем сбора данных, телеметрии или управления на основе Bluetooth, поскольку в качестве одной из сторон беспроводного обмена данными может выступать, например, обычный ноутбук или коммуникатор с поддержкой данной технологии. Wi-Fi при использовании совместно с Bluetooth может выступать вторым каналом передачи для длинных расстояний и условий низкой плотности.

## Выбор оборудования для построения исследовательской сети

Рассмотрим наиболее доступные аппаратные модули IoT с беспроводными технологиями Wi-Fi и Bluetooth с точки зрения доступности на рынке, энергопотребления и доступных технологий.

Настоящим прорывом в IoT стал выпуск модулей на основе Espressif ESP8266, которые могут работать в качестве самостоятельного беспроводного Wi-Fi-контролера либо выступать в роли Wi-Fi-адаптера для управляющего процессора, включая Arduino и Raspberry Pi [14]. С момента выхода на рынок и до сегодняшних дней это самый доступный IoT модуль. Имеет режимы пониженного энергопотребления и множество прошивок, включающих и поддержку различных технологий mesh. Потребление тока при напряжении питания 3.3В – до 300мА при включенной передаче, до 50мА при отключенной связи.

Семейство микросхем Espressif ESP32 поддерживает Wi-Fi и Bluetooth, также у него несколько ядер, множество режимов энергопотребления и внешних выводов [15]. ESP32 дополняет функциональность ESP8266 новыми режимами потребления и работы с беспроводными сетями. Потребление

в активном режиме зависит от работы беспроводных интерфейсов. Потребление тока при напряжении питания 5В – до 300мА при включенной передаче по двум протоколам (не одновременной), до 60мА при отключенной связи.

Arduino платформа с контроллерами ATmega очень популярна среди начинающих разработчиков IoT, несколько вариантов моделей оснащаются Wi-Fi и Bluetooth, дополнительно подключаются практически любые беспроводные модули, включая ZigBee, LoRaWAN и прочие [16]. Потребление в среднем выше, чем у семейства ESP, больше размер, но больше и количество доступной периферии.

Серия плат Raspberry Pi, а особенно Zero W — полноценный одноплатный компьютер, работающий на Linux, имеет Wi-Fi и Bluetooth и возможность подключать внешние модули, в том числе ESP. Потребление высокое, до 2А при напряжении 5В и передаче данных, около 300-500мА при отключенной связи [17].

Менее популярные модули хуже доступны обычным разработчикам, хотя имеют схожие с ESP возможности, но более дорогие. Рассмотрим их с точки зрения модулей связи:

- Tibbo серии EM – есть Ethernet, Wi-Fi, можно подключить GSM/LTE;
- Atmel серии SAMD – есть Wi-Fi, можно подключить почти любые модули связи;
- STM32 серии L – есть Wi-Fi, BLE, опционально ZigBee, Z-Wave, LTE;
- Telit серии WE – есть только Wi-Fi;
- MXCHIP EMW – есть только Wi-Fi;
- Winner Micro серии W – есть только Wi-Fi;
- SIERRA WIFI-BT-IOT-CARD – есть только Wi-Fi, BLE;
- Realtek RTL-01 – есть только Wi-Fi;
- Nordic nRF5xxx – есть только BLE.

В результате были выбраны модули ESP32, как самые доступные в регионе и с наибольшей функциональностью за меньшую стоимость полнофункциональные платы. Arduino в основном использует модули на базе ESP и nRF для обеспечения связи, поэтому энергопотребление и размер устройств будет больше.

В случае, когда оборудование IoT питается от батарей, важную роль играет сокращение энергопотребления [18]. ESP32 имеет несколько режимов сна, использование которых позволяет существенно увеличить сроки службы батарей. Спящий режим – это энергосберегающее состояние, которое сохраняется в оперативной памяти, питание отключается от неиспользуемых периферийных устройств. Существует 5 настраиваемых режимов питания:

1. Активный режим (Active Mode) – все включено, потребление при использовании Wi-Fi до 250мА, при всех модулях в пике до 800 мА.
2. Режим ожидания (Modem Sleep Mode) – отключены все радиомодули, в режиме клиента Wi-Fi каждые 102мс модуль Wi-Fi просыпается для отправки фрейма DTIM (Delivery Traffic Indication Message) для поддержания связи. При снижении частоты процессора потребление около 3мА, при полной частоте – 20мА.
3. Легкий сон (Light Sleep Mode) дополнительно отключает тактовый генератор процессора и памяти, сни-



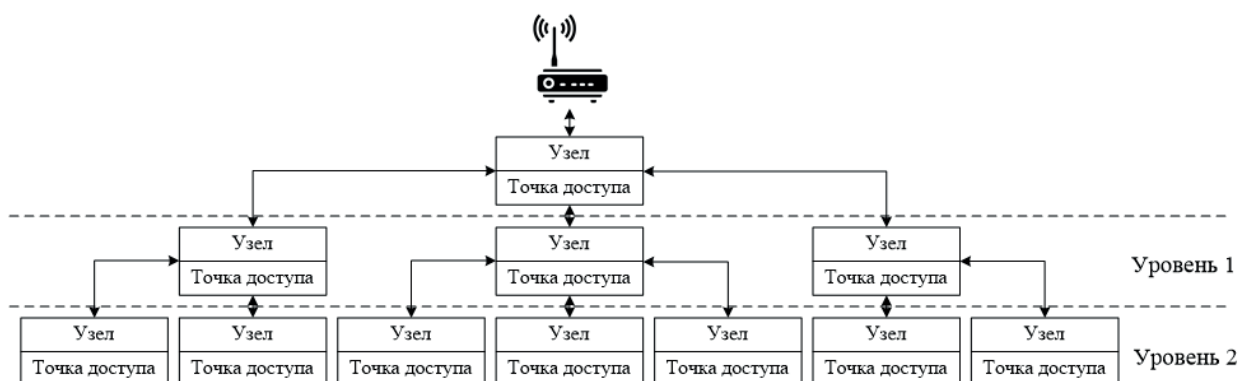
- жая потребление до 0.8 мА.
4. Глубокий сон (Deep Sleep Mode) – отключается все, кроме сопроцессора ULP (ultra low power) и модулей RTC (real time clock). Потребление 150 мкА.
  5. Гиббернация (Hibernation Mode) – отключено все, данные не сохраняются, просыпание только по таймеру или от внешнего события. Потребление 2.5 мА.

Чтобы поддерживать соединения WiFi / Bluetooth, процессор, Wi-Fi, Bluetooth и радиоприемник просыпаются через определенные интервалы. Это так называемый Association sleep pattern. Во время Association sleep pattern режим питания переключается между активным режимом и режимом ожидания модема. ESP32 может войти в режим ожидания модема, только когда он подключается к маршрутизатору в режиме станции.

## Самоорганизующаяся сеть Wi-Fi

Самоорганизующаяся сеть Wi-Fi позволяет соединять многочисленные устройства (далее называемые узлами), распределенные по большой физической области, в единую беспроводную локальную сеть. Такая сеть является самовосстанавливающейся, т.е. может строиться и поддерживается автономно [19].

Самоорганизующаяся сеть отличается от обычной сети Wi-Fi тем, что узлы не обязаны подключаться к центральному узлу. Вместо этого им разрешено соединяться с соседними узлами для передачи друг другу сообщений. Здесь каждый узел может выступать в роли и станции и точки доступа одновременно, следовательно, у него может быть несколько нисходящих соединений, и в то же время только одно восходящее соединение. Это естественным образом приводит к топологии древовидной сети с иерархией «родитель-потомок», состоящей из нескольких слоев (рисунок 1).



Р и с. 1. Топология самоорганизующейся mesh-сети  
F i g. 1. Self-organizing mesh network topology

Самоорганизующаяся сеть Wi-Fi имеет 4 типа узлов [20]:

1. Корневой узел, являющийся верхним узлом в сети. Он подключен к обычному маршрутизатору Wi-Fi и передает пакеты между внешней IP-сетью и самоорганизующейся сетью.
2. Конечные узлы, которым запрещено иметь нисходящие соединения.
3. Промежуточные узлы, имеющие одно восходящее соединение и несколько нисходящих соединений.
4. Свободные узлы, еще не присоединившиеся к сети.

Каждый узел, который способен формировать нисходящие соединения, периодически посылает кадры, чтобы передать другим узлам информацию о своем состоянии. Свободные узлы прослушивают эти кадры, чтобы сгенерировать список потенциальных промежуточных узлов, с одним из которых будет установлено восходящее соединение. Уровень сигнала потенциального восходящего соединения определяется параметром RSSI.

Обобщенно процесс построения самоорганизующейся сети Wi-Fi можно разделить на следующие этапы:

1. Выбор корневого узла.

Корневой узел может быть назначен вручную или выбран динамически в зависимости от уровня сигнала между каждым узлом и маршрутизатором. После выбора корневого

узла соединяется с маршрутизатором и начинает формировать нисходящие соединения. На рисунке 2 в качестве корневого узла выбран узел А, следовательно, он должен установить восходящее соединение с маршрутизатором.

2. Формирование второго слоя.

Как только корневой узел подключился к маршрутизатору, свободные узлы в зоне действия корневого узла соединятся с корневым узлом, образуя тем самым второй уровень сети. После подключения узлы второго уровня становятся промежуточными узлами (при условии, что максимально допустимые уровни больше 2). На рисунке 2 узлы В, С, D находятся в диапазоне корневого узла. Поэтому они образуют восходящие соединения с корневым узлом и становятся промежуточными родительскими узлами.

3. Формирование оставшихся слоев.

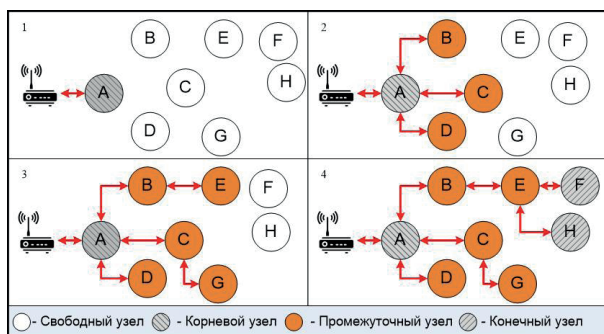
Оставшиеся свободные узлы будут соединяться с промежуточными узлами в пределах диапазона, тем самым формируя новый уровень в сети. После подключения свободные узлы становятся промежуточными или конечными в зависимости от максимально допустимых уровней сети. Этот шаг повторяется до тех пор, пока в сети больше не будет свободных узлов или пока не будет достигнут максимально допустимый уровень сети. Как показано на рисунке 2, узлы Е и G соединя-



ются с узлами В и С соответственно и сами становятся промежуточными узлами.

#### 4. Ограничение глубины дерева.

Чтобы предотвратить превышение максимально допустимого количества уровней в сети, узлы на максимальном уровне после подключения автоматически становятся конечными узлами для предотвращения формирования нового уровня.



Р и с. 2. Процесс построения самоорганизующейся сети Wi-Fi  
F i g. 2. The process of building a self-organizing Wi-Fi network

Когда у свободного узла есть несколько кандидатов в родительские узлы, он установит восходящее соединение с предпочтительным промежуточным узлом. Предпочтительным является узел, расположенный на более высоком уровне сети (включая корневой узел) по сравнению с другими кандидатами. Если на одном и том же уровне имеется несколько кандидатов в родительские узлы, предпочтительным будет родительский узел с наименьшим количеством дочерних узлов.

Каждый узел в сети поддерживает свою отдельную таблицу маршрутизации. Таблица маршрутизации конкретного узла состоит из MAC-адресов всех узлов в подсети этого узла (включая MAC-адрес самого узла). Таблица маршрутизации внутри разделена на несколько подтаблиц, каждая из которых соответствует подсети каждого дочернего узла.

Самоорганизующаяся сеть Wi-Fi использует таблицы маршрутизации, чтобы определить, следует ли пересылать пакет в восходящий или нисходящий поток на основе следующих правил [21, 22]:

1. Если MAC-адрес назначения пакета находится в таблице маршрутизации текущего узла и не является текущим узлом, выбирается подтаблица, которая содержит MAC-адрес назначения, и пакет данных перенаправляется ниже по потоку на дочерний узел, соответствующий подтаблице.

2. Если MAC-адрес назначения не находится в таблице маршрутизации текущего узла, пакет данных пересылается в восходящем направлении на родительский узел текущего узла. В результате пакет поступит в корневой узел, где таблица маршрутизации должна содержать все узлы в сети.

Автоматический выбор корневой узла происходит среди всех свободных узлов. Каждый свободный узел передает в сеть кадр со своим MAC-адресом и значением RSSI. Одновременно с этим узел сканирует кадры от других свободных узлов. Если узел обнаруживает кадр с более сильным RSSI, он начинает передавать содержимое этого кадра. В результате кадр с самым сильным RSSI будет распространен по всей сети. Узел проверяет свой процент голосов, чтобы определить, должен ли он

стать корневой. Если у узла процент голосов превышает предварительное заданное пороговое значение (по умолчанию 90%), то он станет корневой узлом.

Корневой узел также может быть назначен вручную, тогда при включении он сразу соединится с маршрутизатором, а процесс выбора инициализирован не будет.

Построение сети зависит от порядка включения узлов. Если некоторые узлы в сети включаются позже других, окончательная структура сети может отличаться от идеального случая, когда все узлы включаются одновременно. Включенные позже узлы придерживаются следующих правил:

1. Если корневой узел уже существует в сети, включенный позже узел не будет пытаться стать корневой, даже если у него более сильный RSSI. Он присоединится к сети, как и любой другой свободный узел. Если включенный позже узел является назначенным вручную корневой узлом, все остальные узлы в сети будут оставаться бездействующими до тех пор, пока он не включится.

2. Если включенный позже узел становится промежуточным, он также может стать новым предпочтительным родительским узлом для других узлов, т.е. другие узлы могут изменить свои восходящие соединения.

3. Свободный узел с назначенным родительским узлом, задерживающимся при включении, будет бездействовать неограниченное время до тех пор, пока его назначенный родительский узел не включится.

Если корневой узел выходит из строя, связанные с ним узлы второго уровня будут пытаться восстановить соединение. Однако после нескольких неудачных попыток узлы второго уровня будут инициализировать выбор корневой узла. Узел второго уровня с самым сильным RSSI будет выбран в качестве нового корневой узла. Оставшиеся узлы второго уровня сформируют восходящее соединение с новым корневой узлом (или соседним родительским узлом, если корневой не находится в зоне действия).

Если из строя одновременно выходят корневой узел и несколько нижестоящих уровней (например, корневой узел, второй уровень и третий уровень), самый высокий функционирующий уровень инициализирует выбор корневой узла.

Если промежуточный узел выходит из строя, отключенные дочерние узлы первоначально будут пытаться восстановить соединение с ним. После нескольких неудачных попыток переключения каждый дочерний узел начнет поиск потенциальных родительских узлов. Если потенциальных родительских узлов для конкретного дочернего узла нет, он будет оставаться свободным неограниченное время.

Чтобы инициировать переключение корневой узла, текущий корневой узел должен явно инициализировать запуск новых выборов. Если другой узел получает больше голосов, чем текущий корневой узел, будет инициировано переключение корневой узла. В противном случае корневой узел останется неизменным. Вновь выбранный корневой узел отправляет запрос на переключение текущему корневому узлу, который, в свою очередь, ответит подтверждением, означаящим, что оба узла готовы к переключению. Как только подтверждение получено, вновь выбранный корневой узел отключится от своего родителя и сформирует восходящее соединение с маршрутизатором, тем самым став новым корневой узлом сети. Предыдущий корневой узел отключится от маршрутизатора, сохранив все свои нисходящие соединения, и





перейдет в состояние ожидания. Затем он начнет сканирование на наличие потенциальных родительских узлов и выберет предпочтительный родительский узел.

Переключение дочерним узлом своего восходящего соединения на другой родительский узел более высокого уровня происходит автоматически, если становится доступен потенциальный родительский узел более высокого уровня. Все потенциальные родительские узлы периодически передают кадры, позволяя дочернему узлу сканировать наличие родительского узла более высокого уровня.

## Планирование и оптимизация энергопотребления

Для оптимизации энергопотребления mesh-сети требуется составить целевую функцию таким образом, чтобы максимально увеличить время автономной работы всей сети в целом, не снижая ее функциональность [23]. Пусть  $PE_i$  – потребление  $i$ -го устройства для передачи 1024 байт полезной нагрузки,  $KE_i$  – коэффициент (от 0 до 1) для снижения мощности антенны для передачи,  $T_i$  – средний интервал между переда-

чами,  $TE_i$  – время передачи данных,  $PD_i$  – среднее потребление в режиме отключенных интерфейсов (с учетом DTIM фрейма для Wi-Fi),  $TD_i$  – время в режиме отключенных интерфейсов,  $PC_i$  – потребление при прочих операциях с беспроводными сетями, например при соединении или инициализации сети,  $TC_i$  – длительность прочих операций,  $D_i$  – средний размер передаваемых данных в Кб. Потребление измеряется в ваттах (усредненное за интервал времени), время в секундах.

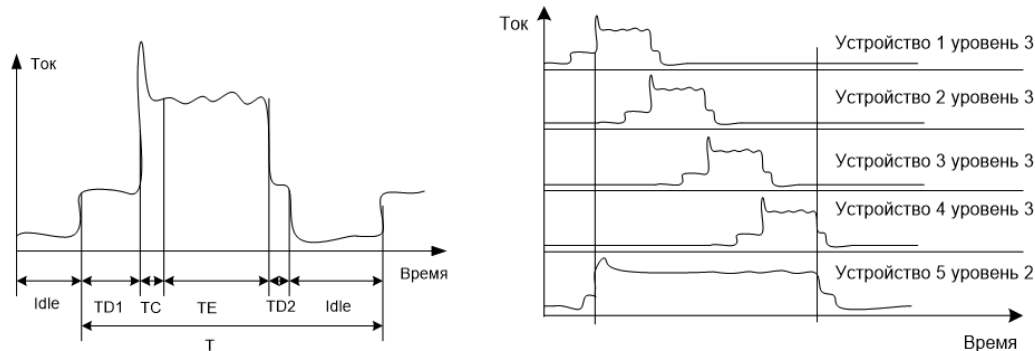
Тогда энергопотребление  $i$ -го устройства, затраченное на одну передачу, можно рассчитать по формуле

$$PO_i(D_i) = TE_i * PE_i * D_i + TD_i * PD_i + TC_i * PC_i$$

Тогда, при условии  $N_i$  передач за время  $t$   $N_i = t / T_i$ . Общее потребление устройства за время  $t$  составит

$$P_i(D_i) = (PO_i(D_i) * N_i) = PO_i * t / T_i.$$

Время простоя (в режиме глубокого сна) –  $PI_i = t - T_i * N_i = t - t / T_i$ .



Р и с. 3. График потребления одного устройства (а) и сети устройств родитель-подчиненный (б)  
Fig. 3. Consumption schedule of one device (a) and a network of devices parent-slave (b)

В случае, когда все устройства синхронизованы, можно выполнять планирование включения и выключения устройств с таким интервалом, чтобы они были друг за другом, но не пересекались и максимально плотно по времени передавали данные (рисунок 3б). Но это возможно только при условии сопоставимых интервалов передачи. Для этого родительское приемное устройство должно задавать каждому подчиненному устройству интервал для следующего просыпания. Если интервалы серьезно различаются, или они нерегулярны, требуется уведомительный подход. Конечное устройство уведомляет об интервале следующего просыпания, таким образом можно перепланировать остальные устройства для более плотного графика просыпания. Отношение  $N_i$  определяет скважность и соответственно общее потребление. При использовании Wi-Fi родительское устройство должно быть в режиме modem sleep для поддержания связи с корневым устройством, а корневое устройство должно быть всегда включено в активном режиме.

В общем случае, для низколатентной передачи данных от последнего устройства к корневому требуется включение всей цепочки коммутации, а для обратной передачи данных – планирование включения всей цепочки.

Общее время работы родителя определяется как

$$TW_i = \sum_j (T_j - TE_j - TD_j - TC_j)'$$

где  $j$  – множество подчиненных узла  $i$ .

При планировании включения промежуточных узлов, у которых нет непосредственных подчиненных, требуется учитывать все узлы вниз по дереву. Если сумма всех  $TW_i$  предпоследнего уровня сети больше доступного времени, или меньше интервалов повторения передачи конечных устройств, возникает проблема одновременной передачи данных, которая решается созданием промежуточных буферов. Суммарное потребление всей сети будет равно

$$P = P_0 + \sum_i (P_i(D_i))'$$

где  $P_0$  – потребление корневого узла[24].

Таким образом, зная времена каждого состояния, можно оценить затраты каждого узла на передачу определенного количества данных.

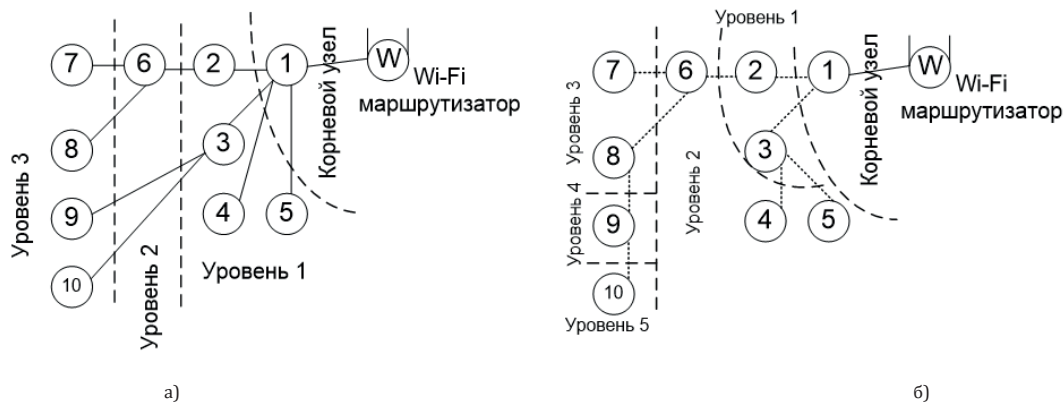


## BLE Mesh и маршрутизация

Самоорганизующаяся сеть на основе Bluetooth более сложна, так как по умолчанию там отсутствует маршрутизация особенности протокола BLE. В наиболее экономичном режиме данные посылаются односторонними посылками сенсорами, а промежуточные устройства просто передают их до корневого узла через механизм проксирования. Для согласования информации конечные устройства должны иногда переходить в активный режим и двусторонним обменом получать необходимые данные. Промежуточные устройства должны или синхронно включаться для связи или быть постоянно включенными в синхронном режиме. Потребление в режиме только BLE и включенной опции в прошивке сон модема Bluetooth (components-> bluetooth enable modem sleep) составляет около 20-30мА в активном режиме.

Для реализации самоорганизующейся сети используется связка PB-GATT прокси и протокола PB-ADV. PB-GATT исполь-

зуется для обнаружения возможностей соседей и проксирования информации о соседях. PB-ADV строит mesh сеть по принципам вещания данных на каналах рассылки рекламных фреймов (маяков, beacon). Вся служебная информация рассылается по этим каналам, и установление двусторонней связи происходит очень быстро. Передача данных по BLE требует меньше затрат энергии, но происходит медленнее (практическая скорость до 235Кбит/с), тогда как Wi-Fi на таких же платах ESP32 может достигать скоростей 23 Мбит/с для UDP. Кроме того, BLE в IoT по умолчанию не содержит расширений для IP протокола и работает только с MAC адресами, поэтому необходимо в оригинальный алгоритм добавлять механизмы сопоставления и обнаружения адресов и маршрутизации, тогда как в Wi-Fi MESH это все уже есть. На рисунке 4 показаны варианты установления связи Wi-Fi и BLE MESH. Как видно, эти сети сильно отличаются, и если вести передачу по разнородным каналам, должны быть инструменты для определения оптимальных маршрутов.



Р и с. 4. Построенная топология для: а) Wi-Fi MESH; б) BLE MESH.

Fig. 4. Topology for: a) Wi-Fi MESH; б) BLE MESH

Поскольку целью построения маршрута могут быть как сокращение общего энергопотребления, так и увеличение емкости сети и снижение задержек, каждый узел-источник должен сообщать свои приоритеты соседям, а алгоритм планирования включения и алгоритм маршрутизации должны динамически прокладывать новые оптимальные для этих целей маршруты в условиях оставшихся после выполнения подобных требований от других узлов свободных ресурсов.

Введем два коэффициента для каждого узла, генерирующего или принимающего информацию –  $bw_i$  для указания приоритета пропускной способности и  $fw_i$  для приоритета энергосбережения. Сеть представим в виде направленного мультиграфа с весами ребер в виде векторов характеристик, соответствующих значениям параметров для применения коэффициентов. Например, для введенных коэффициентов вес ребра будет определяться вектором  $M_j = (bandwidth, PE_j, P_j)$ , где  $bandwidth$  – скорость (реальная) в Кбит/с,  $PE$  – затраченная энергия на 1Кб, в Дж,  $P$  – полная затраченная энергия с учетом включения, соединения с сетью и выключения, в Дж. Тогда энергия для передачи данных каждого маршрута между точками  $k$  и  $m$  для количества данных  $D$  (в Кб) будет:

$$PER_{k,m} = \sum_{j \in J} (PE_j * D)$$

А полная энергия, затраченная на включение и передачу:

$$PPR_{k,m} = \sum_{j \in J} (PE_j * D + P_j)$$

где  $J$  – множество индексов ребер маршрута.

Когда конечный узел подключается к сети, он сообщает свои коэффициенты. Поскольку в таких сетях большинство полезного трафика идет от и до корневого узла, а сеть MESH собирается в дерево, будет существовать множество параллельных маршрутов между двумя узлами, при этом возможны изменения среды передачи. Кроме того, Wi-Fi потребляет больше и медленнее подключается к MESH сети. В результате возникает ситуация, когда для разного объема данных будут более экономны разные каналы связи, и в эксперименте нужно оценить эти значения. При условии постоянного включения BLE можно планировать маршрутизацию только через BLE каналы, а отдельно выносить временные слоты для скоростных Wi-Fi каналов и включать его только по мере необходимости.

Решение по включению, выключению, постоянному питанию некоторых узлов должен принимать планировщик и он будет являться предметом дальнейшего исследования, не включенного в данную статью.



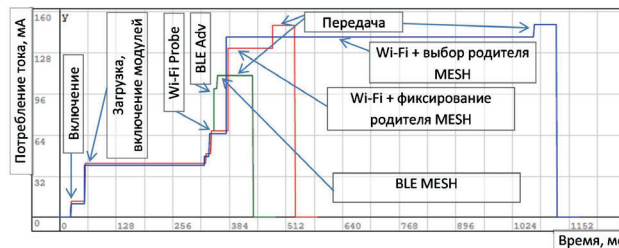
## Эксперимент

Для исследования была создана сеть из модулей ESP32-DEVKIT v1. Для реализации были созданы модули, строящие самоорганизующуюся полносвязную сеть на основе протокола ESP-MESH и собственных дополнений для обеспечения мониторинга и управлением режимами сна и энергопотреблением. Кроме того, была реализована параллельная работа Wi-Fi и BLE в режиме Mesh на основе протокола PB-ADV [25]. Для передачи данных не использовались встроенные пакеты iperf, так как они не работают с BLE Mesh, был разработан собственный сервер и клиент обмена данными.

Для первого эксперимента после запуска конечные узлы передавали серию данных от 16 до 524288 байт, оценивая время передачи, в это время пишущий осциллограф записывал показатели потребляемого тока модулей, синхронизация начала передач была выведена на GPIO (General Purpose Input/Output) контакт в виде единичного импульса платы ESP. Данные передавались сначала по Wi-Fi с использованием IP адреса Wi-Fi шлюза на Linux с запущенным tcpdump с параметрами «-w file» для записи дампа, также в сети присутствовал снифер на основе AirPcap и адаптера TP-Link TL-WN722N в режиме «Monitor Mode». Эксперимент повторялся 10 раз, значения времени и тока усреднялись. Расстояние между узлами не превышало 5 метров. Во второй части эксперимента данные передавались по протоколу BLE на групповой адрес, который был

ассоциирован с корневым узлом, и на нем информация пересылалась на шлюз. Частота ядер в активном режиме была установлена в 160МГц.

Измерение на осциллографе показало неравномерность потребления и длительности в разных режимах, особенно при поиске нового родителя в MESH (рисунок 5).



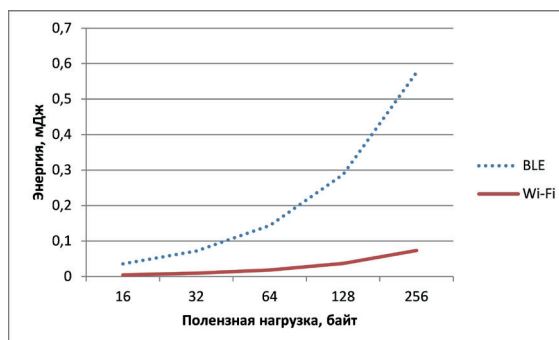
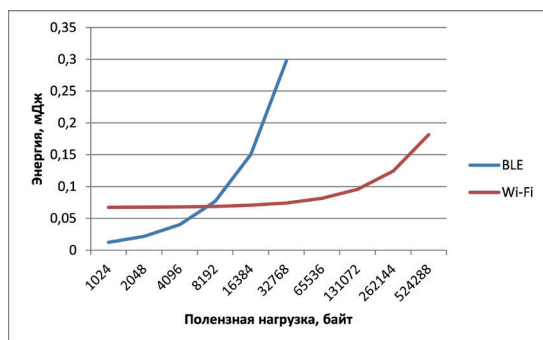
Р и с. 5. Среднее потребление и тайминг при различных режимах работы  
F i g. 5. Average consumption and timing for various operating modes

Оценим затраченную энергию для средних значений скоростей и потребления энергии BLE и Wi-Fi (таблица 2).

Проведенный эксперимент показал поведение потребляемой энергии для разного количества передаваемых байт (рисунок). Так, при значениях менее 10150 байт выгоднее использовать BLE, при больших значениях – Wi-Fi (рисунок 6).

Таблица 2. Оценка затраченной энергии BLE и Wi-Fi  
Table 2. Estimation for consumed BLE and Wi-Fi energy

Параметр	BLE	Wi-Fi
Средний потребляемый ток при передаче в ESP32	40мА + 40мА на CPU	150мА + 40мА на CPU
Реальная скорость	0,235 Мбит/с	23Мбит/с
Время передачи 1024 байт	34,85957 мс	0,356174 мс
Потребляемая энергия на 1Кб при питании 3.3В	9.20 мДж	0,223мДж
Потребляемая энергия на 1Мб при питании 3.3В	9423,79 мДж	228,68 мДж
Время восстановления соединения с соседом после включения	6мс для AES	43мс для WPA2 и 7мс для повторной ассоциации
Время установления MESH соседства	6мс	При подключении к прежнему узлу – 100мс
Потребляемая энергия для полного переключения к сети при питании 3.3В	3,16 мДж	92,16 мДж
Количество байт, когда энергия на соединение и передачу у обоих протоколов одинакова	10150 байт, 94,38 мДж	

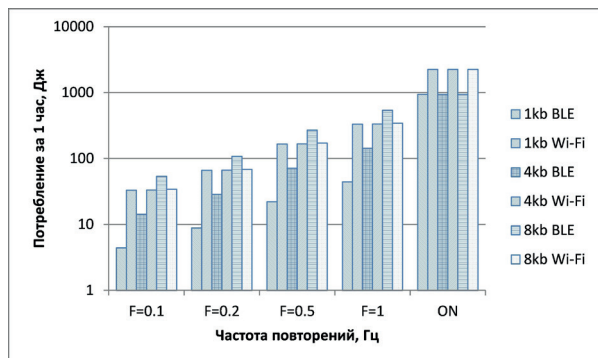


Р и с. 6. Сравнение энергопотребления при передаче разного объема данных  
а) только на передачу; б) на полный цикл включения и передачи.

F i g. 6. Power consumption comparison when transferring different amounts of data  
а) transmission only; б) for a full inclusion and transmission cycle.



Для сравнения полного потребления за период времени при постоянном включении и периодического просыпания требуется скажность посылок данных. Сравним при размере данных 1, 4 и 8 Кб (рисунок 7).



Р и с. 7. Сравнение энергопотребление за 1 час при различной частоте посылок S и в режиме постоянного включения (ON)

Fig. 7. Power consumption comparison for 1 hour at different sampling frequencies S and in constant on mode (ON)

Как видно из графика, при увеличении количества повторений посылки данных зависимость от объема передаваемых данных существенно вырастает, поэтому в будущих исследованиях имеет смысл маршрутизацию проводить еще и в зависимости от размера пакета данных. Разница при одном включении и посылке в секунду и при постоянном включении при передаче 1024 байт для Wi-Fi составляет 6.7 раз, BLE – 21,3 раз, а при передаче 16Кб 1.7 и 6.5 раз соответственно, что говорит в пользу планирования включения вместо постоянной работы узлов, особенно при малых размерах посылок. При использовании широко распространенных аккумуляторов форм-фактора 18650 с емкостью 2200мАч и напряжением 3.7В время работы изученных модулей в режимах периодической передачи для частоты 0.1 Гц (1 раз в 10 секунд, часто использующийся интервал для различных датчиков) составит с учетом ограничений разряда около 270 суток для BLE и 36 суток для Wi-Fi, а в режиме «постоянно включен» - 30 и 12 часов соответственно.

## Заключение

Проведенное изучение возможностей построения самоорганизующейся сети и вопросов энергопотребления позволяет в полной мере выявить недостаточную исследованность вопроса оптимизации таких сетей и, в особенности, ставит под сомнение возможности использования широко распространенных протоколов Wi-Fi и BLE в автономных устройствах в качестве промежуточных узлов MESH сети. Высокое потребление радиомодулей и недостаточно проработанные режимы сна радиомодулей не позволяют использовать их в качестве энергоэффективных точек доступа сети MESH, а только в качестве конечных клиентов. Устройства BLE можно использовать как конечные устройства с хорошей эффективностью даже с модулями ESP32, которые долго включаются. В дальнейшем необходимо рассмотреть вопросы эффективной по потреблению энергии маршрутизации и, особенно, планировщик включения всех узлов для обеспечения связности сети с учётом энергопотребления.

## References

- [1] Sisinni E., Saifullah A., Han S., Jennehag U., Gidlund M. Industrial Internet of Things: Challenges, Opportunities, and Directions. *IEEE Transactions on Industrial Informatics*. 2018; 14(11):4724-4734. (In Eng.) DOI: 10.1109/TII.2018.2852491
- [2] Saloni S., Hegde A. WiFi-aware as a connectivity solution for IoT pairing IoT with WiFi aware technology: Enabling new proximity based services. In: *2016 International Conference on Internet of Things and Applications (IOTA)*, Pune, 2016, pp. 137-142. (In Eng.) DOI: 10.1109/IOTA.2016.7562710
- [3] Jeon K.E., She J., Soonsawad P., Ng P.C. BLE Beacons for Internet of Things Applications: Survey, Challenges, and Opportunities. *IEEE Internet of Things Journal*. 2018; 5(2):811-828. (In Eng.) DOI: 10.1109/JIOT.2017.2788449
- [4] Akeela R., Elziq Y. Design and verification of IEEE 802.11ah for IoT and M2M applications. In: *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Kona, HI, 2017, pp. 491-496. (In Eng.) DOI: 10.1109/PERCOMW.2017.7917612
- [5] Koren A., Šimunić D. Modelling an energy-efficient ZigBee (IEEE 802.15.4) body area network in IoT-based smart homes. In: *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, 2018, pp. 0356-0360. (In Eng.) DOI: 10.23919/MIPRO.2018.8400068
- [6] Yassein M.B., Mardini W., Khalil A. Smart homes automation using Z-wave protocol. In: *2016 International Conference on Engineering & MIS (ICEMIS)*, Agadir, 2016, pp. 1-6. (In Eng.) DOI: 10.1109/ICEMIS.2016.7745306
- [7] Phung K., Tran H., Nguyen Q., Huong T.T., Nguyen T. Analysis and assessment of LoRaWAN. In: *2018 2nd International Conference on Recent Advances in Signal Processing, Telecommunications & Computing (SigTelCom)*, Ho Chi Minh City, 2018, pp. 241-246. (In Eng.) DOI: 10.1109/SIGTELCOM.2018.8325799
- [8] Unwala I., Taqvi Z., Lu J. IoT Security: ZWave and Thread. In: *2018 IEEE Green Technologies Conference (GreenTech)*, Austin, TX, 2018, pp. 176-182. (In Eng.) DOI: 10.1109/GreenTech.2018.00040
- [9] Liu X., Zhang X. Rate and Energy Efficiency Improvements for 5G-Based IoT With Simultaneous Transfer. *IEEE Internet of Things Journal*. 2009; 6(4):5971-5980. (In Eng.) DOI: 10.1109/JIOT.2018.2863267
- [10] Samsiar A., Ibrahim I., Aulia A.L., Sari R.F. Performance comparison of Wi-Fi and LTE for Internet of Things on named data networking. In: *2017 4th International Conference on New Media Studies (CONMEDIA)*, Yogyakarta, 2017, pp. 1-5. (In Eng.) DOI: 10.1109/CONMEDIA.2017.8266021
- [11] Chen M., Miao Y., Hao Y., Hwang K. Narrow Band Internet of Things. *IEEE Access*. 2017; 5:20557-20577. (In Eng.) DOI: 10.1109/ACCESS.2017.2751586
- [12] Amir Alavi S., Rahimian A., Mehran K., Mehr Ardestani J. An IoT-Based Data Collection Platform for Situational Awareness-Centric Microgrids. In: *2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE)*, Quebec City, QC, 2018, pp. 1-4. (In Eng.) DOI: 10.1109/CCECE.2018.8447718
- [13] Bastys I., Balliu M., Sabelfeld A. If This Then What? Con-



- trolling Flows in IoT Apps. In: *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS'18)*. Association for Computing Machinery, New York, NY, USA, 2018, pp. 1102-1119. (In Eng.) DOI: 10.1145/3243734.3243841
- [14] Kodali R.K., Naikoti A. ECDH based security model for IoT using ESP8266. In: *2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, Kumaracoil, 2016, pp. 629-633. (In Eng.) DOI: 10.1109/ICCICCT.2016.7988026
- [15] Babiuch M., Foltýnek P., Smutný P. Using the ESP32 Microcontroller for Data Processing. In: *2019 20th International Carpathian Control Conference (ICCC)*, Krakow-Wieliczka, Poland, 2019, pp. 1-6. (In Eng.) DOI: 10.1109/CarpathianCC.2019.8765944
- [16] Kumar N.S., Vuayalakshmi B., Prarthana R.J., Shankar A. IOT based smart garbage alert system using Arduino UNO. In: *2016 IEEE Region 10 Conference (TENCON)*, Singapore, 2016, pp. 1028-1034. (In Eng.) DOI: 10.1109/TENCON.2016.7848162
- [17] Gao C., Ling Z., Chen B., Fu X., Zhao W. SecT: A Lightweight Secure Thing-Centered IoT Communication System. In: *2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, Chengdu, 2018, pp. 46-54. (In Eng.) DOI: 10.1109/MASS.2018.00018
- [18] Ushakov Yu.A., Ushakova M.V. Research Work of Wireless Internet Devices with Different Mesh Routing Protocols. *Sovremennye informacionnye tehnologii i IT-obrazovanie = Modern Information Technologies and IT-Education*. 2018; 14(3):686-691. (In Russ., abstract in Eng.) DOI: 10.25559/SITITO.14.201803.686-691
- [19] Ushakov Y.A., Ushakova M.V., Shukhman A.E., Polezhaev P.N., Legashev L.V. Service-oriented routing in distributed self-organizing software-defined networks. In: *2018 IEEE 12th International Conference on Application of Information and Communication Technologies (AICT)*, Almaty, Kazakhstan, 2018, pp. 1-5. (In Eng.) DOI: 10.1109/ICAICT.2018.8747111
- [20] Ushakova M.V., Ushakov Y.A., Polezhaev P.N., Tarasov V.N. Model of the Distributed Self-Organizing Network of IoT Sensors. In: *2018 International Scientific and Technical Conference Modern Computer Network Technologies (MoNeTeC)*, Moscow, 2018, pp. 1-6. (In Eng.) DOI: 10.1109/MoNeTeC.2018.8572076
- [21] Ushakov Yu.A., Polezhaev P.N., Shukhman A.E., Ushakova M.V. Distribution of the Neural Network Between Mobile Device and Cloud Infrastructure Services. *Sovremennye informacionnye tehnologii i IT-obrazovanie = Modern Information Technologies and IT-Education*. 2018; 14(4):903-910. (In Russ., abstract in Eng.) DOI: 10.25559/SITITO.14.201804.903-910
- [22] Ushakov Y.A., Polezhaev P.N., Shukhman A.E., Ushakova M.V., Nadezhda M.V. Split Neural Networks for Mobile Devices. In: *2018 26th Telecommunications Forum (TELFOR)*, Belgrade, 2018, pp. 420-425. (In Eng.) DOI: 10.1109/TELFOR.2018.8612133
- [23] Lutui P.R., Cusack B., Maeakafa G. Energy efficiency for IoT devices in home environments. In: *2018 IEEE International Conference on Environmental Engineering (EE)*, Milan, 2018, pp. 1-6. (In Eng.) DOI: 10.1109/EE1.2018.8385277
- [24] Zhang K., Ahn J.H., Lee T., Zhao P. AP scheduling protocol for power beacon with directional antenna in Energy Harvesting Networks. In: *2017 International Conference on Applied System Innovation (ICASI)*, Sapporo, 2017, pp. 906-909. (In Eng.) DOI: 10.1109/ICASI.2017.7988586
- [25] ESP-BLE-MESH Is Now Fully Certified by Bluetooth-SIG. *ESPRESSIF*. Shanghai, China, Oct. 31, 2019. Available at: [https://www.espressif.com/en/news/ESP\\_BLE\\_MESH\\_SIG\\_Certified](https://www.espressif.com/en/news/ESP_BLE_MESH_SIG_Certified) (accessed 02.05.2019). (In Eng.)

Поступила 02.05.2019; принята к публикации 15.08.2019;  
опубликована онлайн 30.09.2019.  
Submitted 02.05.2019; revised 15.08.2019;  
published online 30.09.2019.

#### Об авторах:

**Ушакова Маргарита Викторовна**, старший преподаватель кафедры геометрии и компьютерных наук, Оренбургский государственный университет (460000, Россия, г. Оренбург, пр. Победы, д. 13), ORCID: <http://orcid.org/0000-0003-4462-9946>, m.v.ushakova@mail.ru

**Ушаков Юрий Александрович**, доцент кафедры геометрии и компьютерных наук, Оренбургский государственный университет (460000, Россия, г. Оренбург, пр. Победы, д. 13), кандидат технических наук, ORCID: <http://orcid.org/0000-0002-0474-8919>, unpk@mail.ru

Все авторы прочитали и одобрили окончательный вариант рукописи.

#### About the authors:

**Margarita V. Ushakova**, Lecturer of the Department of Geometry and Computer Science, Orenburg State University (13 Pobeda Av., Orenburg 460000, Russia), ORCID: <http://orcid.org/0000-0003-4462-9946>, m.v.ushakova@mail.ru

**Yury A. Ushakov**, Associate Professor of the Department of Geometry and Computer Science, Orenburg State University (13 Pobeda Av., Orenburg 460000, Russia), Ph.D. (Engineering), ORCID: <http://orcid.org/0000-0002-0474-8919>, unpk@mail.ru

All authors have read and approved the final manuscript.

