

## САМООРГАНИЗУЮЩИЕСЯ MESH-СЕТИ ДЛЯ ЧАСТНОГО ИСПОЛЬЗОВАНИЯ

С.В. Гусс

преподаватель, e-mail: infoguss@gmail.com

Омский государственный университет им. Ф.М. Достоевского

**Аннотация.** В статье рассматриваются особенности самоорганизующихся mesh-сетей, приводятся необходимые определения и пояснения, выделяются главные характеристики. Дается небольшой обзор примеров существующих в мире mesh-сетей общего использования и проприетарных технологий, в основе которых лежат принципы самоорганизации. Обсуждается вопрос создания и развёртывания частной mesh-сети, проблем, возникающих с этим и способов их решения.

**Ключевые слова:** mesh-сети, самоорганизующиеся сети, 802.11s, WMN.

### Введение

**Самоорганизующаяся сеть** (ad hoc network) — беспроводная, динамическая, децентрализованная, мобильная сеть, не имеющая постоянной структуры (площадь покрытия такой сети не обязана быть постоянной; сеть с переменной топологией).

Из представленного выше определения можно выделить четыре главных свойства самоорганизующейся сети:

1. **Беспроводная.** Можно использовать существующие протоколы, стандарты и технологии беспроводной связи, такие, как, например, IEEE 802.11 Wi-Fi (для локальных и городских сетей), IEEE 802.15.1 Bluetooth (для бытовых устройств), IEEE 802.15.4 Zigbee (для датчиков).
2. **Динамическая.** Сеть настраивается сама, автоматически, без участия человека. Требуется обмен управляющей, а в некоторых случаях и статистической информацией между узлами, участвующими в организации сети приёма и передачи данных (например, для балансирования нагрузки и отправки сведений об изменении топологии сети).
3. **Децентрализованная.** В таких сетях нет единого управляющего центра. Каждое устройство сети (абонент) — активный участник процесса организации приёма и передачи данных между сетевыми узлами. В частных случаях абонент может находиться только в одном из режимов: простой клиент (station), точка доступа (access point), прямое соединение (peer-to-peer).

4. **Мобильная.** Узлы, составляющие сеть, могут перемещаться в пространстве, могут выбывать из сети, а новые устройства, в свою очередь, присоединяться к сети и участвовать в её организации. Это вполне могут быть персональные компьютеры, ноутбуки, смартфоны, планшеты, интеллектуальные датчики и другие устройства (подойдёт и Raspberry PI, пример для справки — проект Meshberry [1]), способные принимать и передавать данные, основываясь на установленных в данный момент правилах (на которые влияет структура сети, её связность).

В типичной локальной самоорганизующейся сети устройства пользователей просто подключаются к точкам доступа, которые могут перемещаться в пространстве, как и сами пользователи со своими устройствами. Естественно, что здесь не обойтись без беспроводной технологии подключения, особенно если учесть, что сами точки доступа могут находиться высоко в воздухе или постоянно перемещаться (если, например, установлены в патрульных машинах, ну, или в квадрокоптерах, доставляющих товары, заказанные в интернет-магазине). Такую сеть практически невозможно настроить статически, её топология меняется, а устройства подключаются и отключаются, причём как пользовательские, так и промежуточные сетевые, составляющие основу сети. Поэтому функции настройки и маршрутизации должны быть автоматизированы и оптимизированы по времени организации работоспособности.

Один из типов самоорганизующихся сетей — mesh-сети (ячеистые сети). Это одноранговые (P2P, peer-to-peer) распределённые сети, в которых каждый абонент соединяется со своими ближайшими соседями и может принимать на себя функцию маршрутизатора. Подобные сети способны реализовать высокую отказоустойчивость и применяются в таких областях, как [2]: военная связь, интеллектуальные транспортные системы, локальные сети, беспроводные сенсорные сети; также могут быть использованы в бизнесе, образовании, сфере развлечений, промышленности и коммерции.

В стандарте IEEE 802.11s (Wireless LAN Mesh Network Technology) представлено наиболее полное описание самоорганизующихся сетей.

## 1. Актуальность mesh-сетей

Mesh-сети могут интегрировать в себе различные сетевые и радиотехнологии. Самый распространённый на сегодняшний день стандарт беспроводного соединения устройств — Wi-Fi. Поэтому и сами mesh-сети строятся в основном на этой технологии. А используются такие сети преимущественно для организации локальных (LAN) и городских (MAN) сетей.

Особая актуальность mesh-сетей определяется развитием микроэлектроники, появлением множества различных устройств, способных работать автономно долгое время, имеющих особенность многократной смены режима (online-нахождения в сети и offline-выхода из сети) и нуждающихся в обмене информацией со своим окружением, а возможно и с управляющим или информационным центром (поддержка концепции IoT — Internet of Things, Интернета вещей).

Одно из преимуществ mesh-сетей — независимость. Можно создать свою мобильную сеть передачи данных, которую никто не контролирует, и всё время оставаться на связи. Чем больше абонентов — тем плотнее и надёжнее сеть. Таким образом, можно всегда оставаться на связи в местах, где отсутствует сетевая инфраструктура. Это может оказаться весьма полезно в районах повышенного риска (где вынуждены работать специальные бригады), в местах дикой или неосвоенной природы (где проводят исследования учёные, археологи, геологи, туристы) и удалённых населённых пунктах, где абсолютно каждое абонентское устройство (например, смартфон местного жителя или станция, установленная в транспортном средстве участкового) может принимать участие в процессе передачи важной информации до адресата.

По замыслу, в полноценных mesh-сетях нельзя перехватить трафик и запретить распространение информации. Это, в свою очередь, может противоречить государственным законам конкретной страны или региона. В то же время, государственные структуры и военные ведомства по этим же самым причинам заинтересованы в освоении и организации подобных сетей.

Примеры общественных mesh-сетей:

1. **Guifi** [3]. Каталония, Валенсия. Создана в ответ на отсутствие «доступного» (по цене и качеству) интернет-провайдера. Есть специальные удалённые серверы доступа в Интернет, есть mesh-сети, ряд абонентов которых имеют доступ к этим серверам, а через них, в свою очередь, до Интернета могут добраться и другие участники сети.
2. **AWMN** (Athens Wireless Metropolitan Network) [4]. Греция, Афины. Для маршрутизации используются протоколы BGP (Border Gateway Protocol) и OLSR (Optimized Link-State Routing).
3. **WasabiNet** [5]. США, Сент-Луис. Один из ярких примеров городских сетей, покрывающих отдельные улицы и предлагающих доступ в Интернет как бесплатно, так и за деньги, с определёнными услугами и более высокой скоростью доступа. Для маршрутизации также используется OLSR.
4. **OLPC** (One Laptop Per Child) [6]. Страны третьего мира. Цель — возможность организовать классную/аудиторную mesh-сеть, не используя специальное коммутационное оборудование, а только ноутбуки детей/школьников, выдаваемые им в рамках проекта.
5. **Hyperboria** (прошрое название — Project Meshnet) [7]. В рамках проекта реализован протокол Cjdns. С его помощью можно построить свою инфраструктуру обмена информацией. Но самое главное, протокол решает проблему оптимального перераспределения трафика и перенаправления нагрузки [8]. В отличие от OLSR и других протоколов (например, В.А.Т.М.А.Н.) позволяет объединять отдельные сети и шифровать трафик. Проект также доступен и для русскоязычного сообщества [9].

Лидирующим протоколом на сегодняшний день является Cjdns. Иногда Wi-Fi mesh-сети называют Cjdns-сетями, а протокол Cjdns «движком маршрутизации». Не менее популярен и протокол OLSR (RFC 3626) — протокол маршрутизации для сетей MANET (Mobile Ad hoc Network, RFC 2501). MANET — полностью децентрализованная самоорганизующаяся сеть со случайным соединением узлов.

## 2. Примеры технологий на базе mesh-сетей

Далее будет представлен небольшой обзор организации и возможностей четырёх готовых, проприетарных технологий самоорганизующихся mesh-сетей для совершенно разных сфер применения, чтобы подчеркнуть универсальный характер данного подхода к построению беспроводных соединений.

### **Z-Wave** [10]

Это технология «интеллектуального дома». Каждое устройство Z-Wave сети является одновременно и приёмником, и передатчиком. Пользователь технологии имеет полный контроль над бытовыми устройствами своего дома (освещение, электроприборы, кондиционирование, отопление, видеонаблюдение). Хозяин дома, имея специальный пульт, может управлять автоматизированной системой. Сама сеть состоит из электронных бытовых приборов обслуживания, контроллеров, датчиков (движения, света, температуры, . . .) и других модулей. Есть возможность дистанционного управления через Интернет (например, по смартфону). Таким образом, если хозяин дома забыл выключить утюг или не закрыл дверь гаража, это всегда можно проверить, более того, удалённо можно даже выключить забытое устройство, закрыть дверь, спрятать что-нибудь, перекрыть кран, погасить свет (если система сама до этого «не додумалась»).

Всем управлением занимается специально разработанная микросхема Single Chip и модуль с радиочастотным трактом и антенной. В качестве процессорного ядра выступает хорошо известный и проверенный временем 8051-совместимый контроллер (вычислитель).

Компания Sigma Designs предлагает комплект для разработчиков Z-Wave Development Kit для создания и программирования своей собственной беспроводной mesh-сети на базе протокола Z-Wave.

Типы узлов в Z-Wave mesh-сети: 1) controllers — могут осуществлять маршрутизацию; 2) slaves — посылают, принимают и исполняют команды, могут работать как ретрансляторы сигналов. Максимальное количество узлов — 232. Для упорядочивания устройств используются специальные идентификаторы — Home ID и Node ID. Устройства с абсолютно уникальными Node ID, где бы они не находилось, в пределах дома должны иметь одинаковый Home ID.

### **Ruckus** [11]

Это очередной пример «умной» mesh-сети для создания экономичной, высокопроизводительной, беспроводной локальной сети. Производитель основным пользователем технологии видит коммерческие предприятия и заявляет о таких свойствах, как самоорганизация, самооптимизация и самовосстановление.

Устройства сети:

1. Контроллер сети ZoneDirector. Центральный элемент, через который конфигурируется и управляется вся сеть. Через ZoneDirector администратор сети может просмотреть карту топологии сети и клиентов, внести необходимые изменения.
2. Точки доступа Root AP (корневая точка доступа соединяется с ZoneDirector) и Mesh AP (точка доступа сети, выбирающая оптимальный путь для передачи сигнала другим узлам и корневой точке доступа).

#### **AirTies Mesh** [12]

Технология решает проблему слабого сигнала в беспроводной локальной сети и ограниченного покрытия (например, в многоэтажных бетонных зданиях). Устройство подключается не к одной, а к нескольким точкам доступа, действующим в режиме репитера. Точки доступа образуют mesh-сеть, устройство пользователя подключается к этой сети через точку с самым сильным сигналом. Цель mesh-сети — предоставить оптимальный маршрут до маршрутизатора (чтобы получить доступ к другим сетям, в частности к Интернету).

#### **RAELink3 Mesh** [13]

Это «портативный модем, обеспечивающий беспроводную связь большого радиуса действия между удалёнными портативными газоанализаторами и базовой станцией компьютера для комплексного мониторинга» [14]. К модему могут подключаться до 8 газоанализаторов и до 64 газоанализаторов, оборудованных модемом. Три режима работы: 1) дистанционный — для подключения к базовой станции; 2) базовая станция — для работы в качестве главного модема; 3) ретранслятор.

### **3. Маршрутизация**

Поскольку основной характеристикой mesh-сетей является переменная топология, самый актуальный вопрос, который необходимо решить для её организации, — как эффективно построить маршрут между источником сообщений и адресатом.

Протоколы маршрутизации делят на два класса:

1. *Проактивные* (табличные). Каждый узел строит свою таблицу маршрутизации и делится информацией об изменении топологии сети со своими соседями. Примеры: OLSR, DSDV (Destination-Sequenced Distance Vector).
2. *Реактивные* (работающие по запросу). Таблицы маршрутизации не строятся, маршрут составляется по мере необходимости. Используется широковещание для определения пути отправки сообщения. Примеры: DSR (Dynamic Source Routing), AODV (Ad hoc On-Demand Distance Vector).

На практике используют гибридные протоколы. Крупная сеть делится на подсети. Внутри подсетей на узлах создаются таблицы маршрутизации. Маршрут через подсети составляется через широковещательные запросы и определение наиболее актуального в данный момент пути. Примеры гибридных протоколов: HWMP (Hybrid Wireless Mesh Protocol).

Существуют также протоколы геомаршрутизации, которые предполагают брать информацию о местонахождении участников сетевого взаимодействия через спутниковые системы (ГЛОНАСС, ГЛОНАСС/GPS). Примеры протоколов: GAF (Geographical Adaptive Fidelity), GPSR (Greedy Perimeter Stateless Routing), LAR (Location-Aided Routing).

Также как и для обычных компьютерных сетей (с постоянной топологией) протоколы маршрутизации могут делиться на протоколы вектора расстояния и протоколы состояния канала (с комплексной метрикой маршрутов). Вопрос выбора правильной метрической системы весьма актуален для создания сетей особого назначения.

Обзор актуальных протоколов маршрутизации, их преимуществ и недостатков можно найти в работе [15]. В работе [16] протоколы анализируются по определённой системе критериев, учитывающей функциональные, топологические, дистанционные, ресурсные и другие особенности организации самоорганизующейся сети. Более подробно о возможных критериях и метриках выбора оптимальных маршрутов в mesh-сетях можно узнать в работе [17]. Кратко можно отметить, что для построения маршрута могут учитываться [17]: длина пути, надёжность, задержки, пропускная способность, загрузка и стоимость передачи данных.

#### **4. Создание и развёртывание mesh-сети**

Идея самоорганизующейся сети заключается в том, что если, к примеру, вы живёте в многоэтажном многоквартирном доме, и в каждой (или почти в каждой) квартире есть маршрутизатор (беспроводная точка доступа с функцией маршрутизации и выходом в Интернет, которые сегодня весьма распространены), то эти маршрутизаторы можно объединить в mesh-сеть. Преимущества сети будут состоять в том, что если у одного из пользователей пропадёт Интернет или «упадёт» скорость, можно будет воспользоваться Интернетом соседей (выбор оптимального маршрута). Или если сеть не перегружена, то большой объём информации можно загрузить одновременно через несколько каналов. Такая схема будет работать, если соседи находятся в дружеских отношениях. В реальности это можно организовать через посредника — управляющую компанию, которая сама развернёт сеть и будет предоставлять абонентские услуги доступа к оборудованию. В случае с предприятием или небольшой организацией вопрос собственности уже менее принципиален, а организацией сети может заняться специальный отдел, отвечающий за вопросы автоматизации и предоставления доступа к компьютерной сети.

Проблем становится больше, когда сеть находится в постоянном движении, например, сеть транспортных средств путешественников или просто геологов-исследователей, перемещающихся целевыми группами. В этом случае на помощь приходят компании со своим готовым сетевым оборудованием и предлагают услуги. Сами разработки (реализация технологии) являются проприетарными и не выставляются на всеобщее обозрение.

На сегодняшний день нет полной и точной информации о том, как на основе

имеющихся свободных программных и аппаратных средств создать и развернуть свою mesh-сеть, готовую предоставить полноценные сервисы самоорганизующейся частной сети своим пользователям.

Теоретически, и даже практически, вопрос о настройке существующих специальных операционных систем коммуникационного оборудования и протоколов в той или иной степени раскрыт. Можно без труда объединить в ячеистую топологию Wi-Fi маршрутизаторы и даже подключить к ним абонентов. Проблема состоит в том, что для обычных пользователей, которым нужен готовый, надёжный функционал и набор сервисов, вся эта информация практически ни о чём не говорит, за исключением тех, кто как-то связан с областью компьютерных сетей и информационных технологий. Более того, современный пользователь не хочет покупать специализированное оборудование или тратить деньги на его аренду. Необходимо задействовать то, что у него уже есть. Для жизни в современном развивающемся обществе человеку просто необходим смартфон для ориентации в городе, для связи с друзьями и близкими, для доступа к обслуживающей население информационной среде. Поэтому можно считать, что у каждого, кому понадобилась своя mesh-сеть, найдётся смартфон. Самая распространённая операционная система для смартфонов — Android. Следовательно, для удовлетворения спроса имеет смысл ориентироваться на пользователей, у которых всегда с собой есть смартфон с операционной системой Android.

Можно найти довольно много советов о том, как подключиться к «сети будущего», где нет цензуры и вся информация пользователя доступна только тем, кому он её предоставляет (адресатам). Но это всего лишь предполагаемая замена или альтернатива сети Интернет. Если требуется развернуть безопасную сеть, можно воспользоваться технологией и протоколом Sjdns. Весь трафик сети будет шифроваться автоматически. Протокол гарантирует приватность (никто не перехватит личную информацию, предмет разговора будет сокрыт), но не гарантирует анонимность (всегда можно узнать, кто и с кем общается). Для работы с сетью устройство должно поддерживать работу протокола IPv6. Проблема Sjdns состоит в том, что вы создаёте не совсем свою сеть, а становитесь частью проекта Hyperboria, так называемого независимого Интернета будущего. Тем не менее, это пример mesh-сети, стало быть, в своих разработках можно ориентироваться на огромный опыт, чтобы избежать проблем. Особого внимания заслуживает android-проект Open Garden (набор приложений) [18], позволяющий без особых проблем подключиться к mesh-сети пользователям приложения, тем не менее у приложения всё та же направленность — «общий открытый Интернет», не для частного использования.

Когда речь заходит о mesh-сетях, часто можно встретить такую технологию, как OpenWrt [19]. Немало обсуждений встречается в Интернете по поводу того, как настроить на маршрутизаторе операционную систему OpenWrt для создания mesh-сети. Более того, маршрутизаторы на OpenWrt могут соединяться с Sjdns-узлами, а это значит, что технологии разделяют общую идеологию самоорганизующейся сети. Беда состоит в том, что OpenWrt ориентирован не на мобильные, а на коммуникационные устройства, которые объединяются в mesh-сеть, а устройства пользователей пользуются преимуществами такой сети. Та-

ким образом мобильность реализуется только по отношению к пользователям, перемещающимся в радиусе охвата установленных точек доступа. Вопрос, таким образом, остаётся открытым — насколько реально организовать частную сеть группы людей (тех же геологов как пример перемещающихся по различным местам пользователей возможной сети), имеющих при себе только смартфоны (например, на базе Android, которые могут работать как маршрутизатор, обычно современные Android-смартфоны поддерживают данный функционал). Поскольку в каждом смартфоне есть Wi-Fi и Bluetooth, то рождается аналогия с проводной сетью — по Bluetooth (как замена сериального/последовательного соединения) объединить промежуточные устройства (центральные смартфоны-маршрутизаторы), а по Wi-Fi — открыть доступ участникам, абонентам сети. Или же, наоборот, если два маршрутизатора находятся далеко друг к другу, то соединить их по Wi-Fi, а подключение абонентов осуществлять по Bluetooth. Если развить тему и предположить, что группа геологов перемещается в лесу и у них есть квадрокоптер (или подобное устройство), на который можно установить Wi-Fi ретранслятор, то идея становится вполне жизнеспособной.

Что же в этом случае делать разработчикам для удовлетворения пользовательской потребности? Из вышесказанного следует, что нужно разработать некое универсальное промежуточное программное обеспечение для поддержания надёжной mesh-связи между абонентскими устройствами, которые подключаются друг к другу посредством доступных коммуникационных технологий (Wi-Fi, Bluetooth, ZigBee, IrDA). Причём необходимо учитывать, как будет происходить обмен информацией о топологии и подключениях, как вся эта работа по организации многосвязной сети будет влиять на продолжительность работы аккумуляторов пользовательских устройств. Нужен хорошо продуманный протокол, который учитывает все ограничения устройств и предполагает возможность соответствующей подстройки.

Можно предложить и более амбициозную идею. Учитывая уровень развития электроники и наличия современных программируемых микропроцессорных средств (тех же микроконтроллеров и более интегрированных решений), реализовать на низком уровне подобный представленному выше протокол. Протокол, который можно будет интегрировать в миниатюрное устройство (назовём его UIWRT — Universal Intellectual Wireless Receiver/Transmitter, универсальный беспроводной приёмо-передатчик), способное участвовать в беспроводных соединениях. Пользовательские устройства будут подключаться к умным UIWRT, а те в свою очередь организовывать сеть, mesh-сеть.

## 5. Моделирование

Моделирование работы самоорганизующейся сети остаётся актуальной задачей, поскольку далеко не все места планета покрыты общедоступными системами связи и в каждой сети могут возникать свои потребности. Одна из универсальных проблем — информирование населения (или группы людей) в условиях чрезвычайных ситуаций. В связи с чем становится обоснованной потребность разработки целого программно-аппаратного комплекса для «организа-



ции локального информационного пространства в рамках зоны чрезвычайной ситуации с возможностью взаимодействия её с внешними информационными сетями» [20], а также возможность «развернуть сеть в сроки адекватные ситуации» [20]. Такая система должна обеспечить быстрое развёртывание с минимальными расчётами, оперативное самовосстановление, доступ к мобильным устройствам.

Для построения своей частной mesh-сети можно опираться на стандарт IEEE 802.11s. Несмотря на то, что стандарт рекомендует использовать протокол маршрутизации HWMP (по умолчанию), он не ограничивает использование других протоколов. В частности, можно использовать свою собственную разработку. Поскольку возможности доступных мобильных устройств не ограничиваются технологией Wi-Fi, стоит также воспользоваться рекомендациями стандарта IEEE 802.21 для организации взаимодействия сетей разных типов (802.3, 802.11, 802.15, ...). С другой стороны и сам стандарт рекомендует использовать несколько частотных каналов. Таким образом, на устройстве может быть несколько различных интерфейсов передачи.

Ведутся также работы над открытым стандартом open80211s [21].

Роли устройств (согласно 802.11s): 1) как в стандартной 802.11 сети: а) станция (STA) — пользователь сети, не участвующий в организации инфраструктуры, просто подключается к б) точке доступа (AP); 2) в mesh-сети появляются ко всему: а) узлы mesh-сетей (MP), реализующие mesh-службы для поддержки необходимого функционала, б) точки доступа mesh-сети (MAP) — MP, поддерживающие функции AP, в) порталы mesh-сети (MPP) — это MP, открывающий доступ к внешним сетям.

Формат кадра остался тем же (что и в общем стандарте 802.11), только в поле данных добавляется mesh-заголовок (mesh-флаги, время жизни mesh-пакета, номер mesh-пакета в последовательности, mesh-расширение адреса).

Вопросы, требующие решения для эффективного взаимодействия: маршрутизация, балансирование нагрузки, передача служебных сообщений.

Учитывая всё разнообразие задач, встающих на пути организации эффективной mesh-сети, нужно подумать также над выбором соответствующего задаче алгоритма назначения канала. Алгоритм должен определить, какой интерфейс следует использовать для конкретного случая, учитывая характеристики канала связи. В качестве примера можно привести алгоритм Hyacinth [22, 23]. Для определения канала алгоритм сравнивает пропускную способность соединения с ожидаемой нагрузкой. Для вычисления общей пропускной способности алгоритм учитывает объём трафика и загрузку каналов на маршрутах. Пропускная способность всех соединений не должна быть меньше ожидаемой нагрузки.

Алгоритмы, работающие в mesh-сети должны быть основаны на математической модели. Пример математической модели для организации mesh-сети стандарта IEEE 802.16 (беспроводной городской сети) представлен в работе [24]. В работе рассматривается задача распределения подканалов для оценки характеристик возможных конфигураций сети. Для сравнения полученных результатов используются графы Кенига. Для построения модели сети, согласно [24], следует учитывать: неоднородность устройств и подсетей, динамическое распреде-

ление частотных ресурсов, обеспечение показателей QoS (качество обслуживания), режим управления (распределённый или централизованный), влияние интерференции между устройствами, выделение радиоканалов, протокол маршрутизации, технологические особенности (дальность связи, интенсивность трафика). Чем полнее будет математическое описание, тем проще будет создавать реализующий протокол.

## 6. Прототипирование

Естественное продолжение этапа моделирования — прототипирование или создание экспериментального устройства, реализующего модель для проверки на практике целесообразности дальнейшей разработки.

Далее будет представлен обзор существующих на рынке, готовых к непосредственному использованию модулей аппаратного обеспечения для реализации беспроводных специализированных устройств на базе технологии Wi-Fi. Модули пригодны не только для прототипирования, но также и для проектирования готовых устройств. В обзор включены недорогие по цене решения (согласно распространённому мнению сообщества разработчиков встраиваемых устройств). Более дорогой вариант — использование Raspberry Pi и подключение к ней внешних модулей. Использование Raspberry Pi не всегда целесообразно ввиду более сильного (порой значительно) расходования ресурсов электропитания проектируемого устройства по сравнению со специализированными, менее производительными, микроконтроллерными средствами.

**CC3100MOD** [25]. Специализированный сетевой MCU (Microcontroller Unit, микроконтроллер) для IoT. Это SiP модуль (System in Package). Оборудован беспроводным интерфейсом Wi-Fi 802.11 b/g/n. Скорость передачи полезных данных до 16 Мбит/с (для UDP) и до 12 Мбит/с (для TCP). Может работать в режимах station (клиент), AP (точка доступа) или Wi-Fi Direct (для независимой работы без маршрутизатора). Выполнен в виде микросхемы на базе ядра ARM. Реализует стек протоколов Wi-Fi, TCP/IP (версия 4), шифрование (WPA2, SSL 3.0, TLS 1.2), сервисы (ARP, ICMP, DHCP, DNS). Можно одновременно открыть 8 сокетов или 2 защищённых соединения. Предполагается, что встраиваемое решение на базе процессора может работать год и более от двух батарей AA. Загрузка программ возможна через UART и SPI. Для реализации всех возможностей рекомендуется подключение внешней памяти Serial Flash через SPI-интерфейс. Эта память используется для хранения пользовательских данных, файлов настройки, сертификатов. Для тактирования используются две частоты (два тактовых генератора) — 1) для часов реального времени (32768 Гц), 2) для внутреннего процессора и подсистемы WLAN (40 МГц). Поскольку микроконтроллер поставляется с управляющим драйвером, от разработчика программных приложений не требуется реализовывать с нуля стандартные протоколы или службы, а можно просто использовать высокоуровневый интерфейс команд. Полное описание процессора и принципов его работы можно найти в [26, 27].

**ESP8266** [28]. Микроконтроллер, оборудованный беспроводным интерфей-

сом Wi-Fi 802.11 b/g/n. Режимы: Wi-Fi Direct, soft-AP (software enabled Access Point, программно реализованная точка доступа). Поддерживаются WEP и WPA/WPA2. Управление модулем осуществляется через UART посредством набора AT-команд. Модуль можно перепрограммировать (перепрошивать). Интегрированный стек TCP/IP. Встроенный датчик температуры. Программы выполняются из внешней памяти (микроконтроллер не имеет энергонезависимой памяти программ), подключаемой через SPI. Присутствует 10-битный АЦП. На базе микроконтроллера выполнен модуль ESP8285 со встроенной флэш-памятью в 1 Мбайт. ESP8266 построен на 32-битном ядре Tensilica Xtensa L106 на 80 МГц. Дальнейшее развитие — **ESP32** [29] (двухядерный процессор Tensilica LX108 с частотой до 240 МГц, поддержка Bluetooth).

**Particle Photon** [30]. Модуль представлен в виде небольшой платы. Плата включает в себя 1) микроконтроллер STM32 ARM Cortex M3 на 120 МГц, 2) Wi-Fi модуль Broadcom BCM43362, реализующий стандарты 802.11b/g/n, 3) 1 Мбайт флэш-памяти. Работает на базе операционной системы FreeRTOS. Поддерживает режим soft-AP. Интерфейсы: SPI, UART, CAN. Для программирования загрузчика или прошивки можно использовать JTAG.

**Omega2** [31]. Микрокомпьютер на базе Linux. Wi-Fi-модуль реализует стандарты 802.11b/g/n. Имеются два приёмопередатчика UART и интерфейс SPI. Процессор работает на частоте 580 МГц. 16 Мбайт памяти для хранения программ.

Из современных популярных модулей можно также выделить Oak [32], C.H.I.P. [33], Electric Imp [34].

Все рассмотренные модули выполняют примерно одни и те же функции, характерные для IoT. Разработчику доступны готовые контроллерные платы со всем необходимым микропрограммным обеспечением, предоставляющим свой API. Для быстрых экспериментов вполне могут подойти Particle Photon, Omega2 или решения на подобие Oak или Electric Imp. Для большей свободы реализации стоит использовать микроконтроллеры серии CC3100, ESP8266 или ESP32, распространённые и доступные для свободного приобретения на территории Российской Федерации.

## 7. Заключение

В статье кратко была рассмотрена проблема организации частной самоорганизующейся mesh-сети. Дано общее определение самоорганизующейся сети и особенностей mesh-сетей, поддерживающих идеи самоорганизации. Было показано, что проблема построения частной беспроводной mesh-сети всё ещё актуальна и ближайшее время будут появляться новые задачи, возможные решения и готовые технологии. Проведённый анализ задач организации частной mesh-сети установил необходимость создания оптимальной математической модели самоорганизующейся сети, созданной с учётом имеющихся в данной области стандартов, руководств и рекомендаций. В конце статьи был представлен обзор возможных аппаратных решений для прототипирования промежуточного сетевого оборудования для расширения охвата и увеличения возможностей проек-

тируемой mesh-сети.

## ЛИТЕРАТУРА

1. NYC Mesh. A community-owned resilient network. URL: <https://nycmesh.net/meshberry/> (Дата обращения: 24.09.2016).
2. Беспроводные самоорганизующиеся сети. URL: <http://crossgroup.solutions/adhoc.html> (Дата обращения: 24.09.2016).
3. Xarxa de Telecomunicacions de Comunitat Oberta, Lliure i Neutral. URL: <https://guifi.net> (Дата обращения: 24.09.2016).
4. Athens Wireless Metropolitan Network (AWMN). URL: <http://www.awmn.net> (Дата обращения: 24.09.2016).
5. WasabiNet. URL: <http://gowasabi.net/sitemap> (Дата обращения: 24.09.2016).
6. One Laptop per Child. URL: <http://one.laptop.org> (Дата обращения: 24.09.2016).
7. Hyperboria. URL: <https://hyperboria.net> (Дата обращения: 24.09.2016).
8. Project Meshnet Documentation. URL: <https://docs.meshwith.me/project-goals-ru.html> (Дата обращения: 24.09.2016).
9. Cjdroute.net Hyperboria ex CJDNS.RU. URL: <https://cjdroute.net> (Дата обращения: 24.09.2016).
10. Z-Wave Russia — Умный дом. URL: <http://z-wave.ru> (Дата обращения: 24.09.2016).
11. Ruckus Wireless — лучшие беспроводные WiFi решения. URL: <http://ruckus-wireless.ru> (Дата обращения: 24.09.2016).
12. AirTies Wireless Networks — Technology. URL: <http://www.airties.com/technology.html> (Дата обращения: 24.09.2016).
13. RAELink 3 Mesh. Portable wireless transmitter with integrated GPS. URL: <http://www.raesystems.com/products/raelink-3-mesh> (Дата обращения: 24.09.2016).
14. Руководство пользователя RAELink3 Mesh. URL: [http://www.raesystems.com/sites/default/files/content/resources/Manual\\_RAELink3\\_Mesh\\_Manual\\_RevA\\_3\\_RU.pdf](http://www.raesystems.com/sites/default/files/content/resources/Manual_RAELink3_Mesh_Manual_RevA_3_RU.pdf) (Дата обращения: 24.09.2016).
15. Павлов А.А., Датьев И.О. Протоколы маршрутизации в беспроводных сетях // Труды Кольского научного центра РАН. 2014. № 5(24). С. 64–75.
16. Карманов М.Л. Протокол маршрутизации для ad-hoc сетей // Вестник Южно-Уральского государственного университета. 2009. № 26(159). С. 47–51.
17. Вишневецкий В., Лаконцев Д., Сафонов А., Шпилев С. Mesh-сети стандарта IEEE 802.11s: протоколы маршрутизации // Первая миля. 2009. Том 10, № 1. С. 16–21.
18. Open Garden — Connecting the next billion mobile devices. URL: <http://www.opengarden.com> (Дата обращения: 24.09.2016).
19. OpenWrt. URL: <https://openwrt.org> (Дата обращения: 24.09.2016).
20. Мельников М.И., Ковтун А.С. Самоорганизующаяся сеть оперативного взаимодействия для нужд населения и специальных служб // Доклады Томского государственного университета систем управления и радиоэлектроники. 2014. № 2(32). С. 281–286.

21. open80211s. URL: <http://www.o11s.org> (Дата обращения: 24.09.2016).
22. Легков К.Е., Донченко А.А. Беспроводные mesh-сети специального назначения // T-Comm — Телекоммуникации и Транспорт. 2009. № 2. С. 36–37.
23. Ashish Raniwala, Tzi Chiueh. Architecture and Algorithms for an IEEE 802.11-Based Multi-Channel Wireless Mesh Network. URL: <http://www.ecsl.cs.sunysb.edu/tr/hyacinth-infocom.pdf> (Дата обращения: 24.09.2016).
24. Гаркуша С.В., Гаркуша Е.В., Еременко А.С. Модель распределения подканалов в беспроводной mesh-сети стандарта IEEE 802.16, представленной в виде гиперграфа // Збірник наукових праць Харківського університету Повітряних Сил. 2015. Вип. 2(43). С. 33–38.
25. CC3100MOD. SimpleLink Certified Wi-Fi Network Processor, Internet-of-Things Module Solution for MCU Application. URL: <http://www.ti.com/product/CC3100MOD> (Дата обращения: 24.09.2016).
26. CC3100 — сетевой процессор для «Интернета вещей». Часть I. URL: <http://www.compel.ru/lib/ne/2014/10/2-cc3100-setevoy-protssessor-dlya-interneta-veshhey-chast-i> (Дата обращения: 24.09.2016).
27. CC3100 — сетевой процессор для «Интернета вещей». Часть II. URL: <http://www.compel.ru/lib/ne/2015/2/7-cc3100-setevoy-protssessor-dlya-interneta-veshhey-chast-ii> (Дата обращения: 24.09.2016).
28. Everything ESP8266. URL: <http://www.esp8266.com> (Дата обращения: 24.09.2016).
29. The Internet of Things with ESP32. URL: <http://esp32.net> (Дата обращения: 24.09.2016).
30. Particle: Ship your IoT product. URL: <https://www.particle.io> (Дата обращения: 24.09.2016).
31. Omega2: \$5 Linux Computer with Wi-Fi, Made for IoT. URL: <https://www.indiegogo.com/projects/omega2-5-linux-computer-with-wi-fi-made-for-iot#/> (Дата обращения: 24.09.2016).
32. Oak by Digistump. URL: <http://digistump.com/oak/> (Дата обращения: 24.09.2016).
33. C.H.I.P.s — Get CHIP — The World's First Nine Dollar Computer. URL: <https://getchip.com/pages/chip/> (Дата обращения: 24.09.2016).
34. The Electric Imp Platform. URL: <https://electricimp.com/platform/> (Дата обращения: 24.09.2016).

## PRIVATE WIRELESS MESH NETWORKS

**S.V. Guss**

Tutor, e-mail: infoguss@gmail.com

Dostoevsky Omsk State University

**Abstract.** This paper is about private wireless mesh networks. There are definitions and comments for main features of ad hoc networks, survey of public mesh networks and proprietary technologies, routing, modeling and prototyping.

**Keywords:** mesh networks, ad hoc networks, 802.11s, WMN.

*Дата поступления в редакцию: 26.09.2016*