

## СИМУЛАЦИОННА СРЕДА ЗА ИЗУЧАВАНЕ НА BLUETOOTH LOW ENERGY ТЕХНОЛОГИЯ

Айдън Хъкъ<sup>1</sup>, Йордан Йорданов<sup>1</sup>  
<sup>1</sup>Технически университет – Варна, България

## SIMULATION ENVIRONMENT FOR STUDY OF BLUETOOTH LOW ENERGY TECHNOLOGY

Aydan Haka<sup>1</sup>, Yordan Yordanov<sup>1</sup>  
<sup>1</sup>Technical University of Varnan, Bulgaria  
aydin.mehmed@tu-varna.bg

### Abstract

Nowadays, with the expansion and improvement of communication technologies, the services offered are increasing, such as broadband Internet of Things (IoT) technologies, and one of the most common IoT technologies is Bluetooth Low Energy (BLE). This report presents a simulation product for the study of BLE technology, which can also be used in education.

**Keywords:** Bluetooth, BLE, IoT, Simulator

### ВЪВЕДЕНИЕ

Социалната изолация предизвикана от кризата с коронавирус през последните 18 месеца доведе до възникването на различни предизвикателства пред дейностите, които обикновено се извършват присъствено. Особени трудности възникнаха при провеждане на образователния процес, свързани с преминаването от присъствена към отдалечена форма, като проблеми с комуникацията, невъзможност за работа с физически устройства и извършване на индивидуални задачи, и др.

Благодарение на съвременните комуникационни технологии като 4G и 5G проблемите свързани с комуникацията при провеждане на видео и гласова среща през Интернет се минимизират [1]. Проблемите, които са свързани с индивидуалната работа с физически устройства, може да се преодолеят с помощта на виртуализационните технологии. С тях може да се създадат виртуални устройства за работа по време на занятията, с което обучаващите се имат въз-

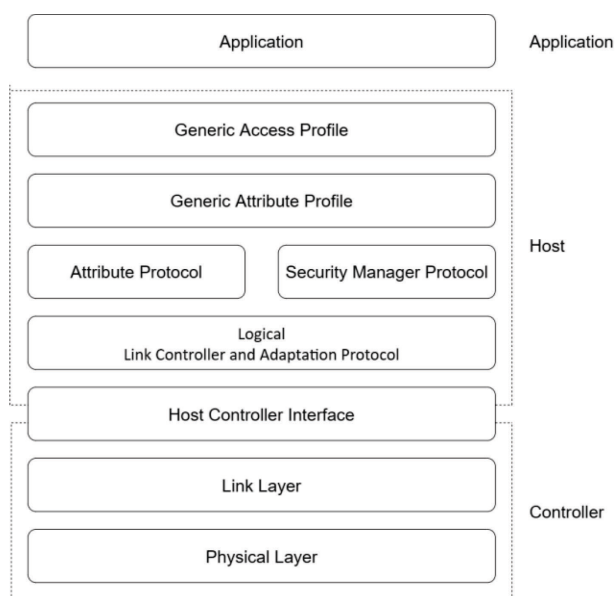
можност за опериране с интерфейса на физическото устройство. При този подход е необходимо да се осигури достатъчно ресурс на сървърна машина за работа с множество виртуални машини и потребители, а също така и обслужващ персонал. Когато това е трудно за осигуряване може да се наложи преподаващият да демонстрира работата през виртуалните машини, за да се избегне изразходването на наличния ресурс, натоварването на инфраструктурата и заделяне на време за поддръжка, или обучаващият се да изгради собствена виртуализационна инфраструктура. Също така в част от занятията може да е невъзможно осигуряване на виртуализация на използвания хардуер като при работа с различни технологии за изграждане на сензорни мрежи за Интернет на обектите (Internet of Things - IoT). При такава ситуация може да се използва симулационна среда за разглежданата технология. Симулаторите осигуряват възможност за инсталиране на отделна машина, както и индивидуална работа, при

изучаване на стандарти, алгоритми или механизми на конкретна технология.

Този доклад представя симулационна среда за изучаване на Bluetooth Low Energy (BLE) технология за целите на обучението.

## ТЕХНОЛОГИЯ BLUETOOTH LOW ENERGY

Технологията BLE е въведена в Bluetooth стандарта във версия 4.0 и е по-известна в приложения, при които консумацията на енергия е от решаващо значение и данните за пренос са с малки количества. Тя се използва основно при фитнес устройства за следене на здравословно състояние, „умна“ система за осветление, система за локализиране в реално време, сензорни мрежи и приложения за навигация в домашна среда [2, 3]. Архитектурата на BLE (Фиг. 1) включва различни компоненти. При тази архитектура Physical и Link Layer слоевете се групират в структура наречена контролер, а останалите слоеве се групират в структура хост [4, 5]. Интерфейсът хост към контролер (Host-Controller Interface - HCI) стандартизира комуникацията между контролера и хоста.



Фиг. 1. Архитектура на BLE

Physical Layer (PHY) се отнася до радио хардуера, използван за комуникация и за модулация/демодулация на данните. BLE работи в ISM обхвата (2,4 GHz), който е

сегментиран в 40 радио-честотни канала, всеки раздалечен през 2 MHz. Три от тези канали се наричат Primary advertising (37, 38 и 39), докато останалите 37 се използват като Secondary advertising и за пренос на данни по време на връзка. Един от основните процеси, въз основа на които е възможна комуникацията между устройствата в BLE мрежата, е изпращането на съобщения за рекламиране (Advertising) по трите основни канала от крайните устройства. Това позволява на главните устройства да сканират рекламодатели, за да ги намерят и да прочетат техните рекламни данни. След това сканиращото устройство може да инициира връзка, ако рекламодателят го позволява.

Слоят Link Layer взаимодейства с физическия слой и осигурява абстракция за слоевете от по-високо ниво и начин за взаимодействие с радио предавателя чрез междинно ниво, наречено HCI. Той е отговорен за управление на състоянието на радио предавателя, както и изискванията за синхронизация, необходими за задоволяване на BLE спецификацията. Също така отговаря за управлението на хардуерни операции като Cyclic Redundancy Check (CRC), генериране на случайни числа и криптиране.

С помощта на Physical и Link Layer слоевете се осигурява комуникацията между устройствата в BLE мрежа. Когато дадено устройство рекламира, то позволява на други устройства, които сканират, да го намерят и евентуално да се свържат с него. Ако рекламиращото устройство позволява връзки и сканиращо го намери и реши да се свърже с него, и двете устройства влизат в свързано състояние. Връзките позволяват предаването на данни от приложенията по надежден начин, тъй като при BLE те използват CRC, потвърждения (Acknowledgements) и препредаване на изгубени данни, за да се осигури коректното им доставяне. Веднъж свързани, главното и подчиненото устройство обменят даннови пакети на редовни интервали, наречени събития на връзката. По време на събитие на връзка, устройствата алтернативно изпращат даннови пакети един към друг, докато нито една от страните не остане без повече данни за изпращане. Събитието на връзка

се случва периодично и непрекъснато, докато връзката се затвори или данните са загубени.

Logical Link Control and Adaptation Layer Protocol (L2CAP) действа като протоколен мултиплексор и обработва сегментирането и повторното сглобяване на пакети. Също така предоставя логически канали, които се мултиплексират по една или повече логически връзки. L2CAP, използван в BLE, е оптимизиран и опростен протокол, базиран на класическия Bluetooth L2CAP.

Attribute Protocol (ATT) осигурява средства за предаване на данни между BLE устройства. Разчита на BLE връзка и осигурява процедури за четене, запис, индикация и уведомяване на стойностите на атрибутите по тази връзка.

General Attribute Profile (GATT) използва се за групиране на отделни атрибути в логически услуги. В допълнение към действителните данни, GATT предоставя и информация за атрибутите, т.е. как могат да бъдат достъпвани и какво ниво на сигурност е необходимо.

General Access Profile (GAP) осигурява средства за BLE устройствата да рекламират себе си или други устройства, правят откриване на устройства, отварят и управляват връзки и излъчват данни.

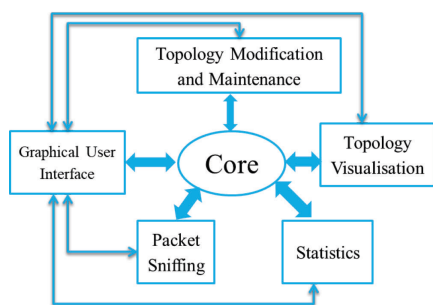
Security Manager (SM) осигурява средства за свързване на устройства, криптиране и декриптиране на данни и активиране на поверителността на устройствата.

Технологията BLE, която е част от най-новата спецификация за Bluetooth 4.1 от Bluetooth Special Interest Group (SIG), е драстично по-малко енергозависима. Твърди се, че типичните подчинени възли, работещи с Bluetooth LE, могат да работят в продължение на месеци или дори години на батерия [6]. Комбинацията от атрибути прави Bluetooth LE привлекателна технология за дизайнери на IoT устройства и на други безжични продукти с малък обем. Технологията предоставя атрибутите: простота, ниска мощност, силен комуникационен сигнал, работа в реално време, съвместимост и др.

## СИМУЛАЦИОННА СРЕДА НА BLUETOOTH LOW ENERGY ТЕХНОЛОГИЯ

Основните процеси по работа на BLE технологията, както и при изпращане на рекламни съобщения, изграждане на връзки и прекратяването им, които се базират на стандарта са внедрени в разработената симулационна среда в катедра Компютърни науки и технологии при Технически университет – Варна. Симулаторът е разработен с технологиите Java, Apache Maven, JavaFX & Scene Builder и Launch4j. Той има модулна архитектура (Фиг. 2) и при стартиране на приложението започва изпълнение на основната функционалност на ядрото, която е добавяне на Master устройството и реализиране на неговата програмна логика за обработка на пристигащите пакети и съответния им Packet Data Unit тип, както и изчакване за добавяне на Slave устройство и проследяване на състоянието му (Standby, Advertising или Connected). Изпълнението на основната функционалност се контролира от класа “AppController”. След това ядрото се обръща към модула за определяне на поддържаната топология (Topology Modification and Maintenance). Визуализирането на топологията се осъществява от модула предназначен за това (Topology Visualisation). Когато ядрото отчете добавяне на Slave устройство, което се контролира от класа “AddDeviceController”, се обръща към модула за поддържане и модифициране на топологията, а след това към модула за визуализация. Така се актуализира мрежовата топологията. При отчитане на промяна в състоянието на Slave устройството от Standby на Advertising ядрото се обръща към модула за проследяване на трафика (Packet Sniffing), от който се получава информация за обменените съобщения между устройствата в мрежата. Това става чрез класа “MessageSequenceFactory”. За получаване на статистическа информация за времето в което крайните устройства в мрежата са били в определено състояние ядрото се обръща към модула Statistics, който се управлява от класа “DeviceStatisticsUtil”.

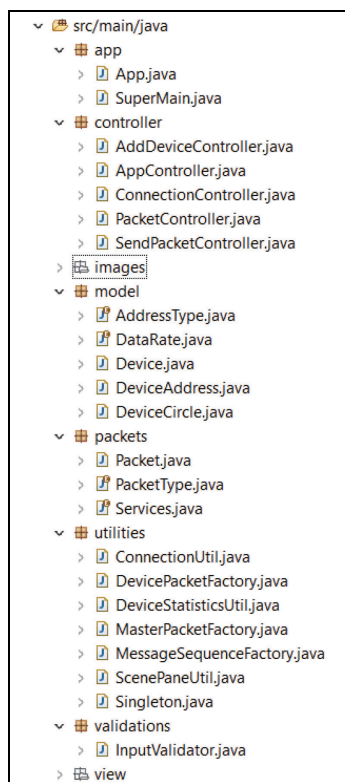
Обработената информация през различните модули се визуализира чрез изградения графичен интерфейс (Graphical User Interface).



Фиг. 2. Архитектура на BLE симулатор

Проектът на симулатора се разделя на 8 пакета, като шаблона за проектиране е MVC (Model – View - Controller). Концепцията му е да се раздели приложението на три основни логически части – модел, изглед и контролер. Моделът съдържа информация за данните, изгледът отговаря за презентацията на данните от модела, а контролерът обработва данните от модела и ги подава за визуализиране от изгледа. Всеки един от тези модули е създаден, за да управлява специфична част от софтуера (Фиг. 3):

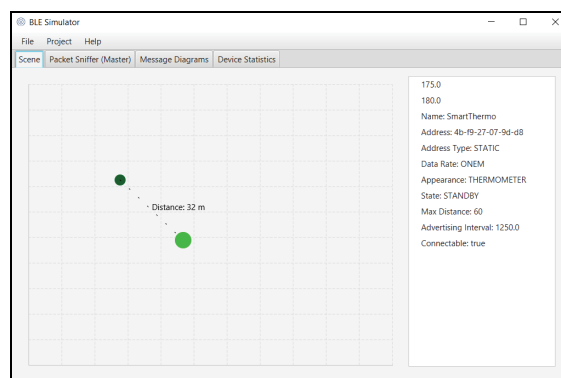
- “app” – съдържа стартовата точка на програмата “SuperMain”, която служи за извикване на стартиращата функция от класа “App”;
- „controller“ – съдържа контролерите на различните fxml файлове;
- “images” – съдържа изображения необходими за програмата;
- “model” – съдържа модел на обекта за устройствата;
- “packets” – съдържа клас “Packet”, който описва пакетните обекти;
- “utilities” – съдържа помощни класове, които извършват различни функционалности;
- “validations” – съдържа клас “InputValidator” съдържа статични функции, които служат за валидация при попълване на полета в програмата;
- “view” – съдържа fxml файловете, които са базирани на xml и служат за визуализиране на дизайна. Също така в този пакет се намират и езиковите файлове за български и английски.



Фиг. 3. Архитектура на пакетите на проекта

## ПОТРЕБИТЕЛСКИ ИНТЕРФЕЙС

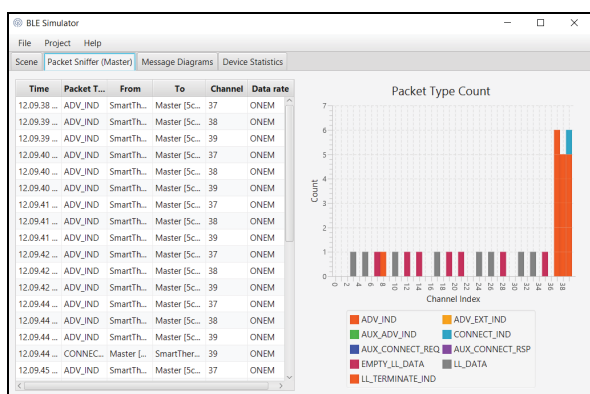
Главният прозорец на програмата (Фиг. 4) се визуализира при стартирането ѝ. В него са разположени меню лентата и лента за табове, като всеки таб изобразява различна функционалност на програмата.



Фиг. 4. Начален прозорец на симулатора

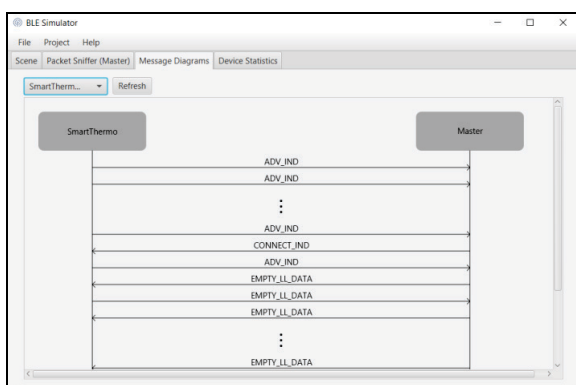
В таб Scene се разполагат елементите за визуализиране на топологията и поле за информация на избраното устройство. При избор на графичен елемент на устройство, в полето отляво се изобразява информация за него, като име, адрес, максимално разстояние и д.р. При първоначално стартиране се визуализира Master устройството в мрежата, а след това може да се добавят Slave. Добавянето на Slave устройства в мрежата

става през допълнителен прозорец, в който се задават характеристиките му. Табът Packet Sniffer (Master) (Фиг. 5) показва информация за това какви пакети постъпват на главното устройство и какво той изпраща, както и диаграма за визуализиране на бройката на различните видове пакети, които са показани в таблицата. В таблицата за пакетите са показани данни за часът и дата на изпращане на пакета, типът на пакета, от кого и до кого е, на кой канал е изпратен и какъв е неговия data rate.



Фиг. 5. Изглед на таб Packet Sniffer (Master)

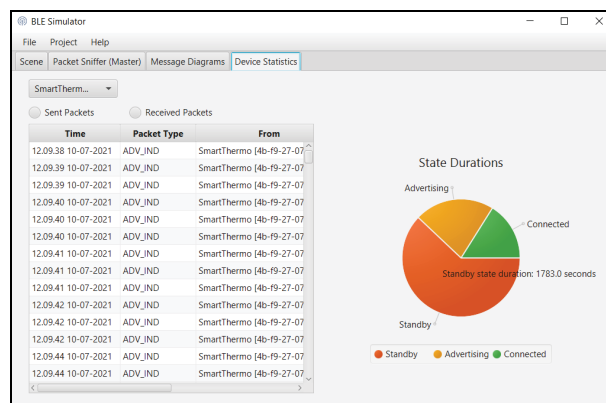
Табът Message Diagrams (Фиг. 6) служи за представяне на диаграма за последователността на съобщенията, като се показва какъв тип пакет се изпраща от кое за кое устройство, в удобен за четене вид. Представената последователност описва процеса за обмен на съобщения при изграждане на връзка, изпращане на данни и термиране на връзката между Master и Slave устройства в BLE мрежа.



Фиг. 6. Последователност от съобщения между избрано и главно устройство

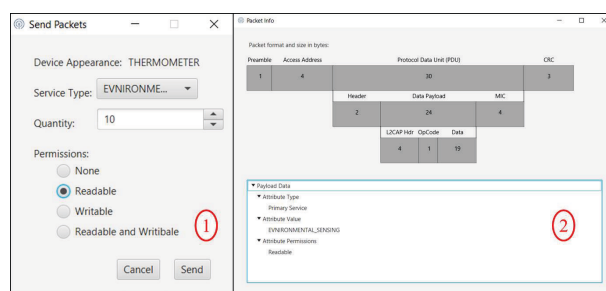
Табът Device Statistics (Фиг. 7) показва информация за периферните устройства, като получени и изпратени пакети и също

така визуализира диаграма за време на работа във състоянията му на рекламиране, в режим на готовност и на връзка.



Фиг. 7. Статистика за Slave устройства

При изпращането на пакети се изобразява какъв е изгледът на устройството и от това зависи какъв вид пакет на услугите ще може да бъде изпратен от тези въведени в програмата (Фиг. 8-1). Друга информация, която се въвежда е бройката на тези пакети и какви са разрешенията за работа с данните от пакетите, които ще се изпращат. При изобразяването на допълнителна информация на пакетите, се генерира формата на пакета и размера на отделните му елементи (Фиг. 8-2). Формата за всеки тип пакети е различен, според дефиницията му в Bluetooth класификацията.



Фиг. 8. Изпращане на даннови пакети и структура на даннов пакет

## ЗАКЛЮЧЕНИЕ

Този доклад представя симулационен продукт за BLE технология. Разработеният симулатор позволява изучаване на различни аспекти, свързани с работата на BLE стандарта, без да е необходимо наличието на физически устройства. Предоставената функционалност показва, че симулаторът може да се използва за представяне на

акцентите от комуникацията между главно и подчинено устройство, дава възможност за извеждане на статистическа информация за времето, в което крайните устройства са били в определено състояние, както и изследване съдържанието на фреймовете, предавани между комуникаращите устройства. Възможностите, които предоставя разработеният симулационен продукт дават повод за твърдение, че е подходящ за използване в обучението както при присъствена, така и при онлайн форма.

## ПРИЗНАТЕЛНОСТ

Изследванията, резултатите от които са представени в настоящия доклад, са проведени по научен проект на ТУ-Варна „Интегриране на виртуализационни и мрежови технологии за целите на дистанционно обучение в условията на covid-19“, който се финансира от държавния бюджет.

## ЛИТЕРАТУРА

- [1] Ericsson mobility report, June 2021. <https://www.ericsson.com/en/mobility-report>. Дата на последно посещаване 03.08.2021
- [2] Wireless Connectivity Options for IoT Applications – Technology Comparison. <https://www.bluetooth.com/blog/wireless-connectivity-options-for-iot-applications-technology-comparison/>. Дата на последно посещаване 03.08.2021
- [3] 2021 Bluetooth Market Update. <https://www.bluetooth.com/bluetooth-resources/> 2021-bmu/?utm\_campaign=bmu&utm\_source=internal&utm\_medium=web&utm\_content=2021bmu-resourcepopup. Дата на последно посещаване 03.08.2021
- [4] UG103.14: Bluetooth® LE Fundamentals, Silicon Labs, UG103.14: Bluetooth® LE Fundamentals
- [5] M. Afaneh. Intro to Bluetooth Low Energy: The easiest way to learn BLE Kindle Edition. Independently Published, 2018, ISBN: 9781790198153, <https://www.novelbits.io/introduction-to-bluetooth-low-energy-book/>
- [6] The pros and cons of Bluetooth Low Energy. <https://www.electronicsweekly.com/news/design/communications/pros-cons-bluetooth-low-energy-2014-10/>. Дата на последно посещаване 03.08.2021