TUTORIAL 2: Tổng quan về công nghệ BLE và ví dụ minh họa (Phần 2)

Nội dung trình bày:

- Protocols và Profiles;
- Bộ giao thức cho BLE (BLE Protocol Stack);
- Các profiles cơ sở: GAP và GATT;
- Các profiles cho từng ứng dụng cụ thể;
- Ví dụ minh họa thiết bị theo dõi nhịp tim (Heart Rate Monitor– HRM)

1. Protocols và Profiles

Để hai thiết bị có thể giao tiếp với nhau thông qua chuẩn BLE, các thiết bị BLE cần tuân thủ một số quy định. Các quy định này được khái quát hóa thành các giao thức và cấu hình.

*** Protocol (Giao thức):** Tập các luật quy định việc định dạng gói tin, định tuyến, dồn kênh, mã hóa,… để trao đổi dữ liệu giữa các bên.

*** Profile (Cấu hình):** Định nghĩa cách mà giao thức được dùng để đạt các mục tiêu cụ thể. Có hai loại cấu hình là cấu hình chung (generic profiles) và cấu hình cụ thể theo trường hợp sử dụng (use-case profiles)

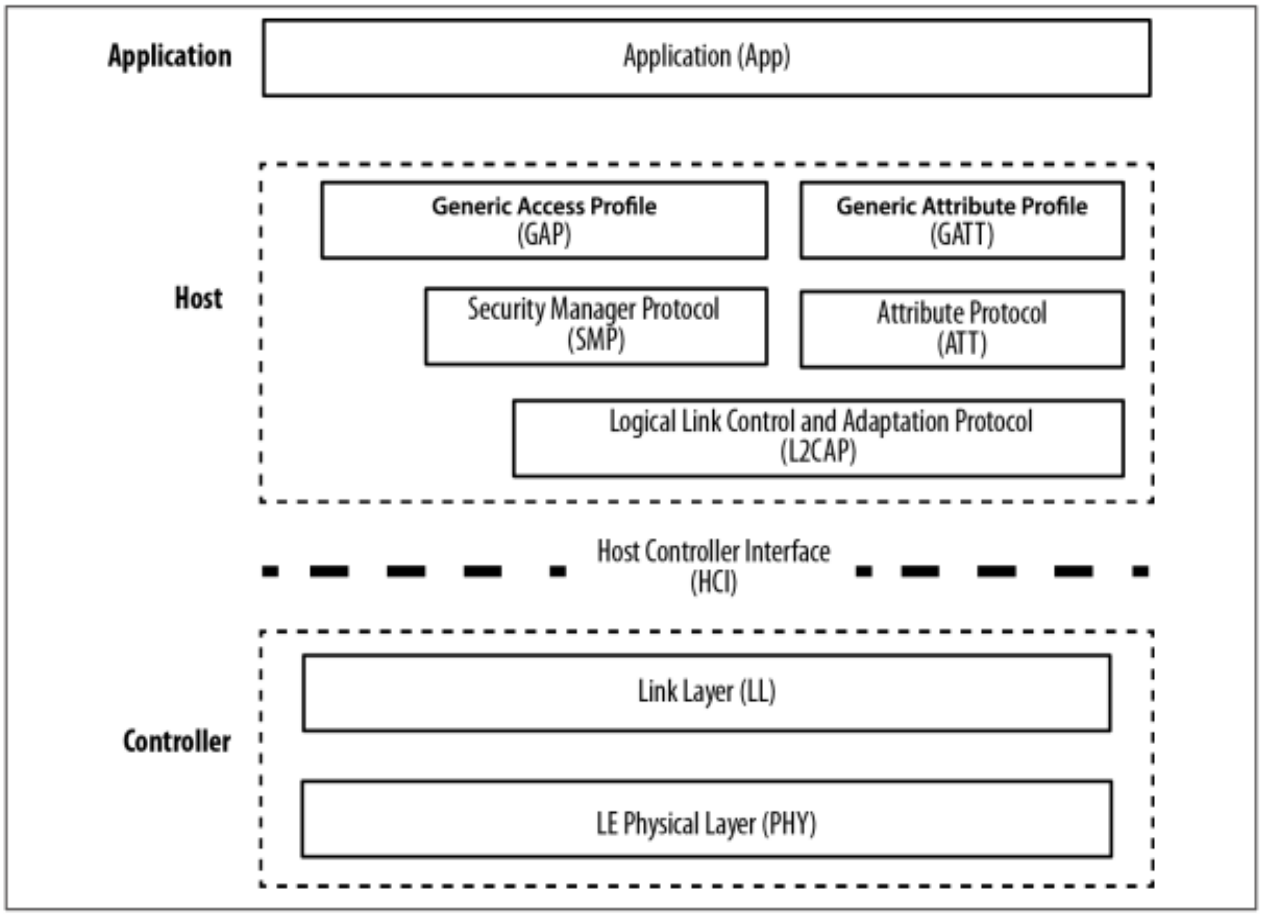
- Generic profiles: các profile cơ sở được định nghĩa trong tài liệu Bluetooth Specifications, đặc biệt là hai profiles không thể thiếu giúp các thiết bị BLE kết nối và trao đổi dữ liệu với nhau, GAP và GATT.

- Use-case profile: Các profile cho các trường hợp sử dụng cụ thể
+ Các profile do Bluetooth Special Interest Group (SIG) định nghĩa
+ Các profile do vendor tự định nghĩa

2. The BLE Protocols Stack

Để lập trình cho thiết bị BLE, có thể chỉ cần quan tâm đến các hàm API ở lớp trên của bộ giao thức BLE (BLE protocol stack), nhưng tốt hơn hết chúng ta nên bắt đầu với một cái nhìn cơ bản về bộ giao thức cho BLE, giúp cung cấp kiến thức nền tảng để có thể nghiên cứu sâu hơn về BLE.

Hình dưới thể hiện đầy đủ về các thành phần bên trong bộ giao thức BLE cho thiết bị Bluetooth Smart.



Hình 1: BLE protocol stack

Bộ giao thức cho thiết bị BLE được chia thành 3 phần chính: controller, host và application. Mỗi phần bao gồm một hoặc nhiều lớp (layer) theo chức năng:

- Application:

Là lớp cao nhất của bộ giao thức, cung cấp giao diện người dùng, xử lý logic, và điều khiển dữ liệu của mọi thứ liên quan đến các trường hợp hoạt động của ứng dụng. Kiến trúc của application phụ thuộc nhiều vào từng bài toán cụ thể.

- Host: bao gồm các lớp sau

+ Generic Access Profile (GAP)

+ Generic Attribute Profile (GATT)

+ Attribute Protocol (ATT)

+ Security Manager (SM)

+ Logical Link Control and Adaptation Protocol (L2CAP)

+ Host Controller Interface (HCI), Host side

- Controller: bao gồm các lớp sau

+ Host Controller Interface (HCI), Controller side

+ Link Layer (LL)

+ Physical Layer (PHY)

Bảng tổng hợp chức năng của các lớp trong stack

Lớp	**Mô tả**
Controller	
Physical Layer	Là lớp thấp nhất làm nhiệm vụ truyền nhận tín hiệu - Chuyển đổi qua lại giữa tín hiệu số và tương tự - Điều chế và giải điều chế tín hiệu - Dải tần sử dụng 2.4GHz ISM (Industrial, Scientific, Medical), chia làm 40 kênh từ 2.4 GHz đến 2.4835 GHz

Lớp	**Mô tả**
Link Layer	Quản lý liên kết - Được cài đặt ở cả phần cứng và phần mềm - Các chức năng thường được cài đặt trong phần cứng: + Preamble, Access Address, and air protocol framing + CRC generation and verification + Data whitening + Random number generation + AES encryption - Link Layer định nghĩa các vai trò sau: + Advertiser: Một thiết bị gửi các gói tin quảng bá + Scanner: Một thiết bị quét các gói tin quảng bá + Master: Một thiết bị khởi tạo 1 kết nối và quản lý nó + Slave: Một thiết bị chấp nhận 1 yêu cầu kết nối và tuân theo master's timing. - Bluetooth Device Address: tương tự như địa chỉ MAC trong card mạng của PC.
Host Controller Interface (HCI), Controller side	Giao diện để kết nối giữa host và controller
Host	
L2CAP	2 chức năng chính: - Như một giao thức dồn kênh, từ nhiều giao thức lớp trên rồi đóng gói thành định dạng gói BLE chuẩn và ngược lại

Lớp	Mô tả
	- Phân mảnh và tái kết hợp: lấy các gói dữ liệu lớn từ các lớp trên và chia chúng thành các gói BLE 27 byte tại bên truyền. Tại bên nhận nó làm ngược lại
Attribute Protocol	ATT là một giao thức client/server phi trạng thái đơn giản dựa trên các thuộc tính được thể hiện bởi một thiết bị. Trong BLE, mỗi device là một client, một server, hoặc cả hai, không phân biệt nó là master hay slave. Một client yêu cầu dữ liệu từ 1 server, và 1 server gửi dữ liệu đến các client - Mỗi server chứa dữ liệu được tổ chức theo dạng của các thuộc tính (attributes), mỗi một thuộc tính được gắn với một handle 16bit, 1 UUID (ID duy nhất), tập giới hạn quyền, 1 giá trị. - Khi một client muốn đọc hoặc ghi các giá trị thuộc tính từ/đến một server, nó phát ra một read request hoặc write request đến server với handle. Server sẽ đáp ứng với giá trị thuộc tính hoặc một tín hiệu ACK. Trường hợp hoạt động đọc, client phân tích giá trị và hiểu kiểu dữ liệu dựa trên UUID của thuộc tính. Khi ghi, client mong đợi để cung cấp dữ liệu với kiểu thuộc tính và server sẵn sàng để nhận
Security Manager	Chuỗi các thuật toán có thể được dùng để bảo đảm an ninh cho quá trình truyền nhận dữ liệu qua BLE
Generic Attribute Profile	Dựa trên ATT và bổ sung thêm một hệ phân cấp và mô hình dữ liệu - Định nghĩa cách tổ chức dữ liệu và trao đổi dữ liệu giữa các ứng dụng. - Dữ liệu được đóng gói trong các services. Trong services gồm các characteristics

Lớp	Mô tả
Generic Access Profile	Chỉ ra cách các thiết bị thực hiện các thủ tục như tìm kiếm thiết bị, kết nối, thiết lập an ninh, các thủ tục khác để đảm bảo các hoạt động nội bộ và cho phép trao đổi dữ liệu diễn ra giữa các thiết bị của các hãng sản xuất khác nhau

Tóm lại, BLE protocol stack bao gồm nhiều lớp, mỗi lớp đảm nhiệm một vài chức năng nhất định giúp thực hiện quá trình giao tiếp giữa các thiết bị BLE với nhau.

3. Các profiles cơ sở: GAP và GATT

* GAP (Advertising and Connections)

GAP (Generic Access Profile) là nền tảng cho phép các thiết bị BLE giao tiếp với nhau. Nó cung cấp một framework mà bất cứ thiết bị BLE nào cũng phải tuân theo để có thể tìm kiếm các thiết bị BLE (Bluetooth) khác, quảng bá dữ liệu, thiết lập kết nối an ninh, thực hiện nhiều hoạt động nền tảng theo một chuẩn.

Tài liệu BLE Specifications định nghĩa các khái niệm sau khi xét đến sự tương tác giữa các thiết bị:

- Roles: Mỗi thiết bị có thể hoạt động theo một hoặc nhiều vai trò khác nhau tại cùng một thời điểm: broadcaster, observer, central, peripheral.
- Modes: Một mode là một trạng thái mà thiết bị có thể chuyển đến trong một khoảng thời gian để đạt được một mục đích cụ thể hoặc nhiều điều đặc biệt, để cho phép một peer thực hiện một thủ tục cụ thể.
- Procedures: Là các thủ tục (thường thì Link Layer điều khiển sự trao đổi gói tin) để cho phép một thiết bị đạt được một mục đích chắc chắn. Một thủ tục thường được liên kết với một mode, nên mode và procedure thường được xem xét cùng nhau.

Mode	Applicable Role(s)	Applicable Peer Procedure(s)
Broadcast	Broadcaster	Observation
Non-discoverable	Peripheral	N/A
Limited discoverable	Peripheral	Limited and General discovery
General discoverable	Peripheral	General discovery
Non-connectable	Peripheral, broadcaster, observer	N/A
Any connectable	Peripheral	Any connection establishment

Hình 2: Modes, Roles và Procedures

Procedure	Applicable Role(s)	Applicable Peer Mode(s)
Observation	Observer	Broadcast
Limited discovery	Central	Limited discoverable
General discovery	Central	Limited and General discoverable
Name discovery	Peripheral, central	N/A
Any connection establishment	Central	Any connectable
Connection parameter update	Peripheral, central	N/A
Terminate connection	Peripheral, central	N/A

Hình 3: Procedures và các modes yêu cầu

- Security: GAP xây dựng dựa trên Security Manager và Security Manager Protocol (định nghĩa các modes và procedures an ninh để xác định cách mà các thiết bị đặt mức an ninh khi trao đổi dữ liệu). Ngoài ra GAP định nghĩa thêm các tính năng an ninh cao hơn mà không gắn với modes và procedures cụ thể nào, tăng cường mức bảo vệ dữ liệu được yêu cầu bởi mỗi ứng dụng.

Xét một cách bản chất thì GAP là lớp điều khiển cao nhất của BLE (Topmost control layer) và là cấu hình bắt buộc cho tất cả thiết bị BLE.

* GATT (Services and Characteristics)

GATT thiết lập chi tiết cách trao đổi tất cả profile và dữ liệu người dùng qua kết nối BLE. Ngược lại với GAP (định nghĩa sự tương tác mức thấp với các thiết bị), GATT chỉ trình bày các thủ tục truyền và định dạng dữ liệu thực tế

GATT sử dụng ATT và giao thức truyền của nó để trao đổi dữ liệu giữa các thiết bị. Dữ liệu này được tổ chức phân cấp thành các phần gọi là services, nó nhóm các phần khái niệm liên quan của dữ liệu người dùng gọi là characteristic. Nói một cách ngắn gọn thì dữ liệu truyền qua BLE là dữ liệu có cấu trúc, mà cụ thể là được tổ chức phân cấp thành services và characteristics.

Roles

GATT Client: tương ứng với ATT client, gửi yêu cầu đến server và nhận kết quả phản hồi. Ban đầu, GATT Client không biết server hỗ trợ những thuộc tính nào vì thế nó cần phải thực hiện service discovery.

GATT Server: tương ứng ATT server, nhận yêu cầu từ client và gửi những nội dung tương ứng.

Chú ý rằng các vai trò của GATT không phụ thuộc vào vai trò của GAP. Có nghĩa là cả GAP Central và GAP Peripheral có thể hoạt động như GATT Client hoặc GATT Server hoặc thậm chí là cả hai tại cùng một thời điểm.

UUIDs

Là một số định danh thiết bị, dài 128 bit (16 byte) duy nhất trên thế giới. Vì độ dài quá lớn, chiếm phần lớn trong gói dữ liệu, BLE Specification định nghĩa thêm 2 định dạng UUID: 16bit và 32 bit. Các định dạng ngắn này có thể chỉ được sử dụng với UUID được định nghĩa trong BT Specification.

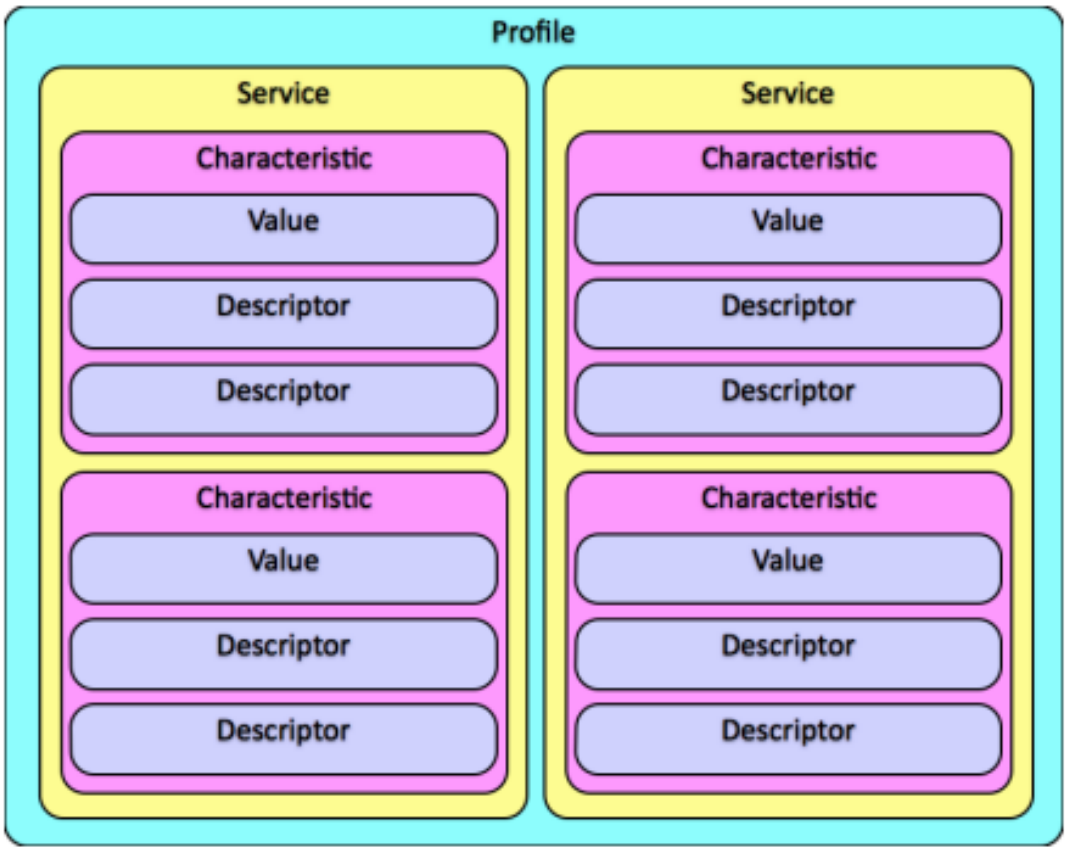
Attributes

- Là thực thể dữ liệu nhỏ nhất được định nghĩa bởi GATT (và ATT).
- Cả GATT và ATT chỉ làm việc với attributes nên để tương tác giữa client và server tất cả dữ liệu phải được tổ chức theo định dạng này.
- Mỗi attribute chứa thông tin về chính nó là dữ liệu người dùng và được mô tả như sau:

- Handle: số 16 bit duy nhất trên mỗi server để địa chỉ hóa attribute
- Type:là kiểu UUID, 16bit – 32bit – 128 bit
- Permission: xác định các ATT opertation có thể thực thi trên attribute cụ thể
- Value: chứa phần dữ liệu thực tế trong attribute, giới hạn 512 byte

Services và Characteristics

Dữ liệu trao đổi thông qua kết nối BLE là dữ liệu có cấu trúc, được tổ chức phân cấp thành các services, bản thân services lại bao gồm các characteristics.



Hình 4: Cấu trúc của một profile

4. Các profiles cho từng ứng dụng cụ thể

GAP và GATT là hai profiles nền tảng cho mọi ứng dụng BLE. Ngoài ra, tùy ứng dụng mà các thiết bị sẽ cung cấp các profile khác (dựa trên GAP và

GATT). Các services chính là các dịch vụ mà thiết bị cung cấp như: Heart Rate Monitor, Battery, Health Thermometer, HID,…

Để biết thông tin chi tiết các services, truy cập các đường dẫn sau:

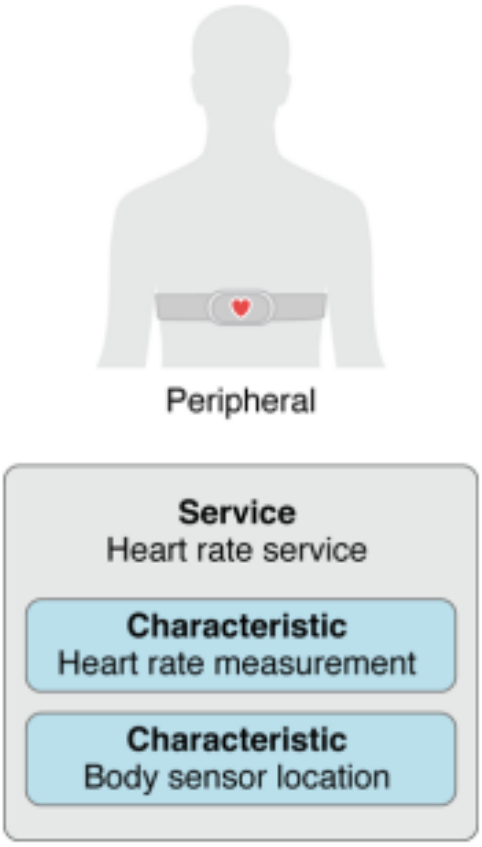
https://www.bluetooth.com/specifications/adopted-specifications

https://www.bluetooth.com/specifications/gatt/services

Các đường dẫn trên trình bày các profiles đã được tổ chức Bluetooth SIG định nghĩa. Ngoài ra các nhà sản xuất thiết bị có thể tự định nghĩa thêm các profiles khác.

5. Ví dụ minh họa thiết bị theo dõi nhịp tim (Heart Rate Monitor– HRM)

Hình dưới thể hiện cấu trúc của Heart Rate Service (HRS)



Hình 5: Thành phần chính của Heart Rate Service

Ta thấy, HRS bao gồm 2 characteristics giúp cung cấp thông tin về nhịp tim và vị trí gắn cảm biến trên cơ thể. Chi tiết có thể xem tại đường dẫn sau:

https://www.bluetooth.com/specifications/gatt/viewer?attributeXmlFile=org.bluetooth.service.heart_rate.xml

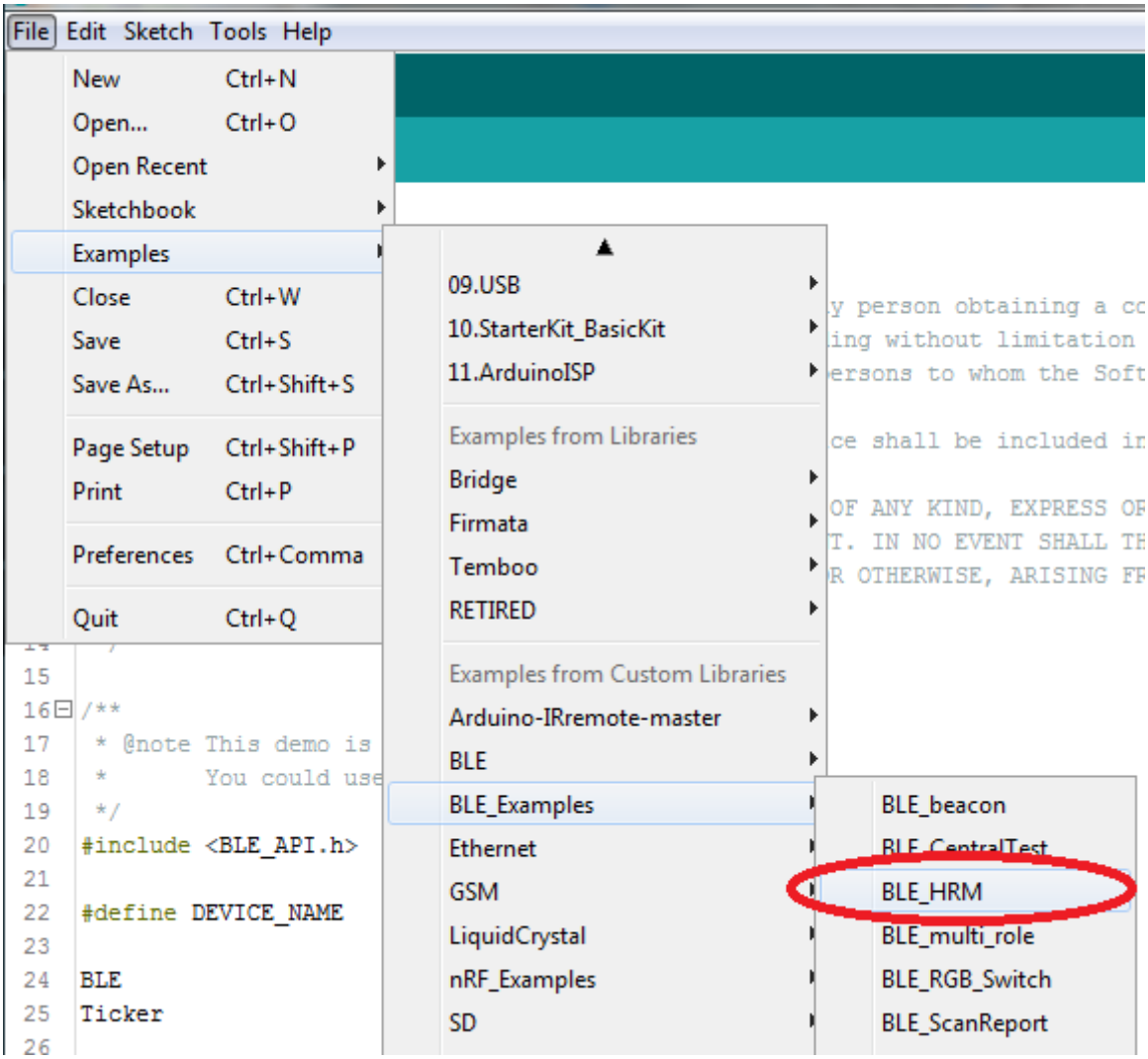
Ví dụ sau minh họa một thiết bị BLE có chức năng đo nhịp tim và truyền về BLE Central (Smartphones, tablets, PC). Vì không có cảm biến hỗ trợ tính năng này nên trong ví dụ này mình chỉ tạo giả tín hiệu theo mô tả sau:

+ Giá trị nhịp tim thay đổi tuyến tính từ 100 đến 175 với độ tăng là 1

+ Vị trí cảm biến trên cơ thể: Finger (mã = 0x03)

Thực hiện:

- Mở ví dụ HRM: File => Examples => BLE_Examples => BLE_HRM

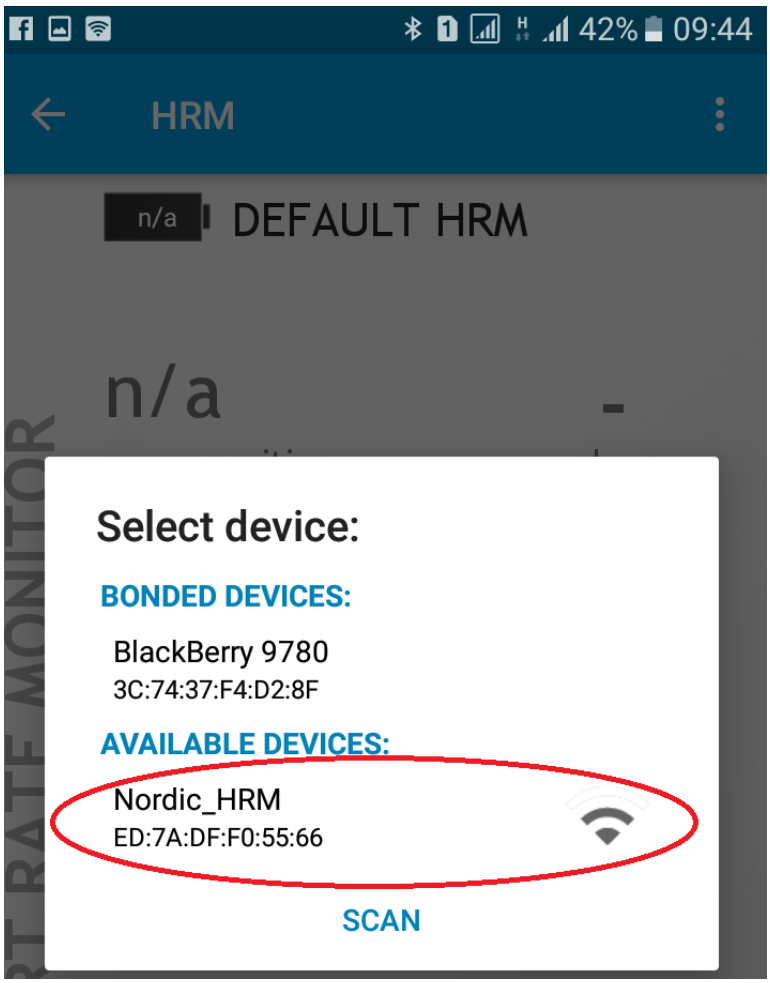


Hình 6: Chương trình ví dụ thiết bị HRM

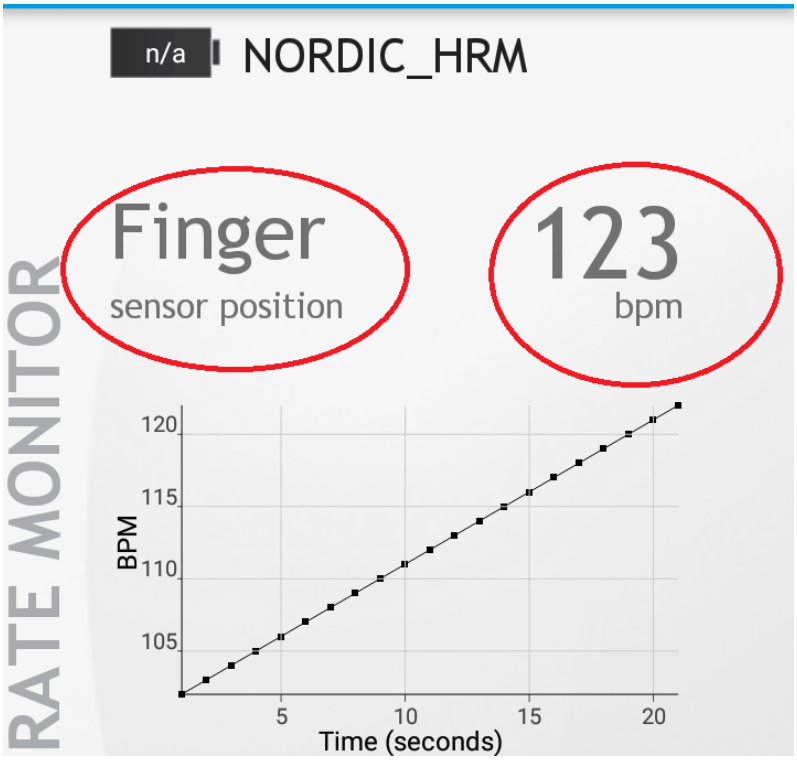
- Tiến hành biên dịch và upload xuống bo mạch VBLUno, chuyển switch bootloader-application sang vị trí để chạy application. Nếu thành công bạn đã có một thiết bị BLE minh họa tính năng theo dõi nhịp tim.

- Để kết nối đến thiết bị này, bạn cần một BLE Central (Smart phone, tablet, PC). Ở đây mình dùng phần mềm nRF Toolbox của Nordic và phần mềm BLE Tool để minh họa.

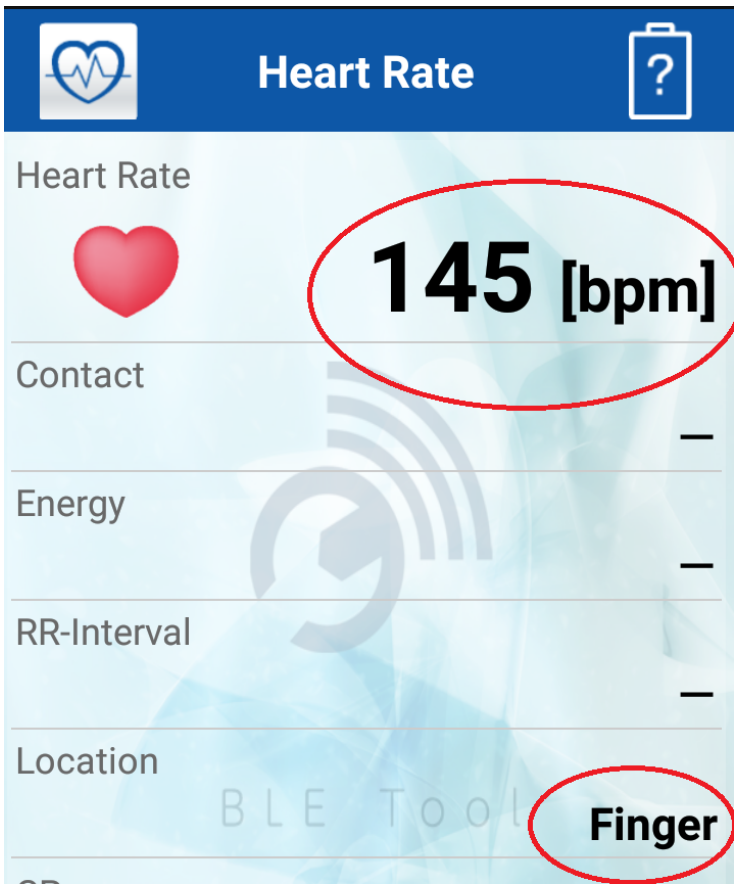
Trên điện thoại Android hoặc iPhone, chạy phần mềm nRF Toolbox, chọn mục HRM, nhấn Connect để tìm kiếm thiết bị HRM và yêu cầu kết nối.



Hình 7: Tìm thấy thiết bị HRM với tên Nordic_HRM



Hình 8: Chương trình kết nối với thiết bị HRM, nhịp tim: 123, vị trí: Finger



Hình 9: Minh họa trên phần mềm BLE Tool cũng tương tự

Mã nguồn của chương trình ví dụ khá đơn giản, bạn có thể tự đọc và tìm hiểu. Nếu có thắc mắc bạn có thể liên hệ với nhóm tác giả.

Tutorial 2 đã trình bày các kiến thức cơ bản nhất về BLE. Các kiến thức quan trọng nhất cần nắm đó là: mô hình mạng, các vai trò của thiết bị, các profiles quan trọng như GAP và GATT, services và characteristics. Bài viết cũng minh họa bằng một ví dụ tiêu biểu – Heart Rate Monitor.

Trong bài viết tới, chúng tôi sẽ trình bày thêm một số ứng dụng tiêu biểu của BLE.

-----The END-----