

Sveprisutno računarstvo

1. Bluetooth Low Energy - BLE

- GAP
- GATT
- Services
- Characteristic

Creative Commons



<u>Sveprisutno računarstvo</u> by Hrvoje Mlinarić & Igor Čavrak, FER is licensed under CC BY-NC-SA 4.0

Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

This license requires that reusers give credit to the creator.

It allows reusers to distribute, remix, adapt, and build upon the material in any medium or format, for noncommercial purposes only.

If others modify or adapt the material, they must license the modified material under identical terms.

BY: Credit must be given to you, the creator.

NC: Only noncommercial use of your work is permitted.

SA: Adaptations must be shared under the same terms.

Bluetooth Low Energy - BLE

- light-weight subset of classic Bluetooth
- introduced as part of the Bluetooth 4.0 core specification
- There are plenty of wireless protocols
 - BLE is so interesting because it's almost certainly the easiest way to design something that can talk to any modern mobile platform out there (iOS, Android, Windows phones, etc.)
- Support for Bluetooth Low Energy (which is a subset of BT 4.0) is available on most major platforms as of the versions listed below
 - iOS5+ (iOS7+ preferred)
 - Android 4.3+ (numerous bug fixes in 4.4+)
 - Apple OS X 10.6+
 - Windows 8 (XP, Vista and 7 only support Bluetooth 2.1)
 - GNU/Linux Vanilla BlueZ 4.93+

GAP – (Generic Access Profile)

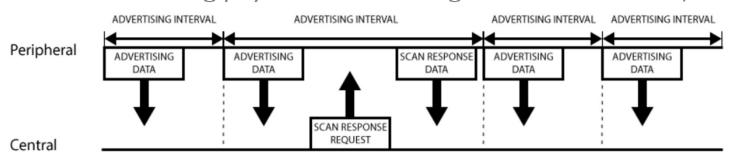
- GAP is an acronym for the Generic Access Profile, and it controls connections and advertising in Bluetooth
- GAP is what makes your device visible to the outside world, and determines how two devices can (or can't) interact with each other.
- GAP defines various roles for devices, but the two key concepts to keep in mind are Central devices and Peripheral devices.
 - Peripheral devices are small, low power, resource contrained devices that can connect to a much more powerful central device. Peripheral devices are things like a heart rate monitor, a BLE enabled proximity tag, etc.
 - Central devices are usually the mobile phone or tablet that you connect to with far more processing power and memory.

Advertising and Scan Response Data

- There are two ways to send advertising out with GAP:
 - Advertising Data payload
 - Scan Response payload.
- Both payloads are identical
 - can contain up to 31 bytes of data
- Advertising data payload is mandatory
 - constantly transmitted out from the device
 - let central devices that it exists.
- Scan response payload is an optional
 - payload that central devices can request, and allows device designers to fit a bit more information in the advertising payload such a strings for a device name, etc.

Advertising and Scan Response Data

- There are two ways to send advertising out with GAP:
 - Advertising Data payload
 - Scan Response payload.
- Both payloads are identical
 - can contain up to 31 bytes of data
- Advertising data payload is mandatory
 - constantly transmitted out from the device
 - let central devices that it exists.
- Scan response payload is an optional
 - payload that central devices can request, and allows device designers to fit a bit more information in the advertising payload such a strings for a device name, etc.



Broadcast Network Topology

- While most peripherals advertise themselves so that a connection can be established (GATT services and characteristics) there are situations where you only want to advertise data.
- By including a small amount of custom data in the 31 byte advertising or scan response payloads, you can use a low cost Bluetooth Low Energy peripheral to sent data one-way to any devices in listening range
- This is known as Broadcasting in Bluetooth Low Energy.
- This is the approach use by Apple's iBeacon, for example, which inserts a custom payload in the main advertising packet, using the Manufacturer Specific Data field.

Generic ATTribute - GATT

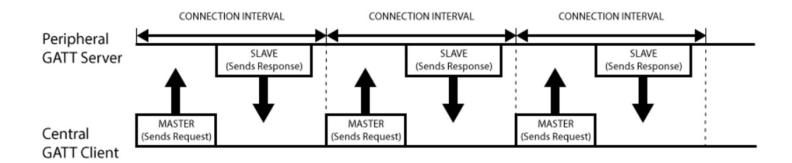
- GATT (Generic ATTribute Profile) defines the way that two Bluetooth Low Energy devices transfer data back and forth using concepts called Services and Characteristics.
- It makes use of a generic data protocol called the Attribute Protocol (ATT), which is used to store Services, Characteristics and related data in a simple lookup table using 16-bit IDs for each entry in the table.
- GATT comes into play once a dedicated connection is established between two devices, meaning that you have already gone through the advertising process governed by GAP.
- The most important thing to keep in mind with GATT and connections is that connections are exclusive.
 - is meant that a BLE peripheral can only be connected to one central device (a mobile phone, etc.) at a time.
 - As soon as a peripheral connects to a central device, it will stop advertising itself and other devices will no longer be able to see it or connect to it until the existing connection is broken.

Connected Network Topology

- Peripheral can only be connected to one central device (such as a mobile phone) at a time, but the central device can be connected to multiple peripherals.
- If data needs to be exchanged between two peripherals, a custom mailbox system will need to be implemented where all messages pass through the central device.
- Once a connection is established between a peripherals and central device, however, communication can take place in both directions, which is different than the one-way broadcasting approach using only advertising data and GAP.

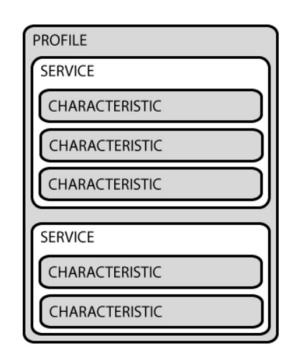
GATT Transactions

- GATT uses server/client relationship concept.
- The peripheral is known as the GATT Server, which holds the ATT lookup data and service and characteristic definitions, and the GATT Client (the phone/tablet), which sends requests to this server.
- All transactions are started by the main device, the GATT Client, which receives response from the secondary device, the GATT Server.
- When establishing a connection, the peripheral will suggest a 'Connection Interval' to the central device, and the central device will try to reconnect every connection interval to see if any new data is available
- 'Connection Interval' is just a suggestion
 - Your central device may not be able to honour the request because it's busy talking to another peripheral or the required system resources just aren't available.



Services and Characteristics

- GATT transactions in BLE are based on nested objects called Profiles, Services and Characteristics
- A Profile doesn't actually exist on the BLE peripheral itself, it's simply a pre-defined collection of Services that has been compiled by either the Bluetooth SIG (Special Interest Group) or by the peripheral designers.
 - The Heart Rate Profile, for example, combines the Heart Rate Service and the Device Information Service.



Services

- Services are used to break data up into logical entities, and contain specific chunks of data called characteristics.
- Service can have one or more characteristics
- Each service distinguishes itself from other services by means of a unique numeric ID called a UUID, which can be:
 - 16-bit (for officially adopted BLE Services)
 - 128-bit (for custom services).
- A full list of officially adopted BLE services can be seen on the Services page of the Bluetooth Developer Portal.
- If you look at the Heart Rate Service, for example, we can see that this
 officially adopted service has a 16-bit UUID of 0x180D, and contains up to 3
 characteristic, though only the first one is mandatory: Heart Rate
 Measurement, Body Sensor Location and Heart Rate Control Point.

Characteristic

- lowest level concept in GATT transactions is the Characteristic
 - encapsulates a single data point (though it may contain an array of related data, such as X/Y/Z values from a 3-axis accelerometer, etc.).
- Similarly to Services, each Characteristic distinguishes itself via a pre-defined 16-bit or 128-bit UUID, and you're free to use the standard characteristics defined by the Bluetooth SIG (which ensures interoperability across and BLEenabled HW/SW) or define your own custom characteristics which only your peripheral and SW understands.
- As an example, the Heart Rate Measurement characteristic is mandatory for the Heart Rate Service, and uses a UUID of 0x2A37. It starts with a single 8-bit value describing the HRM data format (whether the data is UINT8 or UINT16, etc.), and then goes on to include the heart rate measurement data that matches this config byte.

Characteristic

- Characteristics are the main point that you will interact with your BLE peripheral
- They are also used to send data back to the BLE peripheral
- You are also able to write to characteristic.
- You could implement a simple UART-type interface with a custom 'UART Service' and two characteristics:
 - one for the TX channel
 - one for the RX channel
- One characteristic might be configured as read only and the other would have write privileges.

Getting Started with Bluetooth Low Energy

- by Kevin Townsend, Carles Cufí, Akiba, Robert Davidson
- Released May 2014
- Publisher(s): O'Reilly Media, Inc.
- ISBN: 9781491949511