Bluetooth Low Energy

Architecture:

The Bluetooth architecture is divided into three parts:
1) Controller
2) Host
3) Application

The controller is a physical device which transmits and receives the radio signals. Contoller also understands the signals in order to extract the information which comes in the form of packets within them.

Host manages the communication between the connected devices and also provides services to the connected devices using radio.

Application layer uses the host and controller to implement certain function.

1. Controller:

Controller or we can also say it as a bluetooth chip or radio. Controller interacts with outside world with the antenna. Antenna are used for transmtting and receving messages to and from the devices.
Controller interfaces with host through the host controller interface.

1.1. Physical Layer:

Physical layer does the work of transmitting and receiving the bits in the form of waves to and from the devices over the 2.4GHz radio waves. Radio waves carry the information by varying the amplitude, frequency, or phase of a wave in the given frequency band. The frequency of waves is varied to allow either a zero or one to be exposed from the pattern of the waves, using a modulation scheme called Gaussian Frequency Shift Keying(GFSK).

The Frequency shift keying means that the ones and zeros are coded on to the radio by slightly shifting the frequencies up and down. Bluetooth low energy uses the spread-spectrum radio regulations for transmitting the bits.

Radio frequency transmission pattern of bits:

Transmission of radio frequencies of positive frequency deviation of more than 185kHz -> 1

Transmission of radio frequencies of negative frequency deviation of more than 185kHz -> 0

For effective working of the physical layer the 2.4 Ghz band is split into 40 separate RF channels, each 2MHz apart from one another.

1.2. Direct Test Mode:

Direct test mode is used for the testing of the physical layer. Direct test mode allows the tester to command controllers physical layer to either transmit the sequence of test packets or either receive the test packets.

1.3. Link Layer:

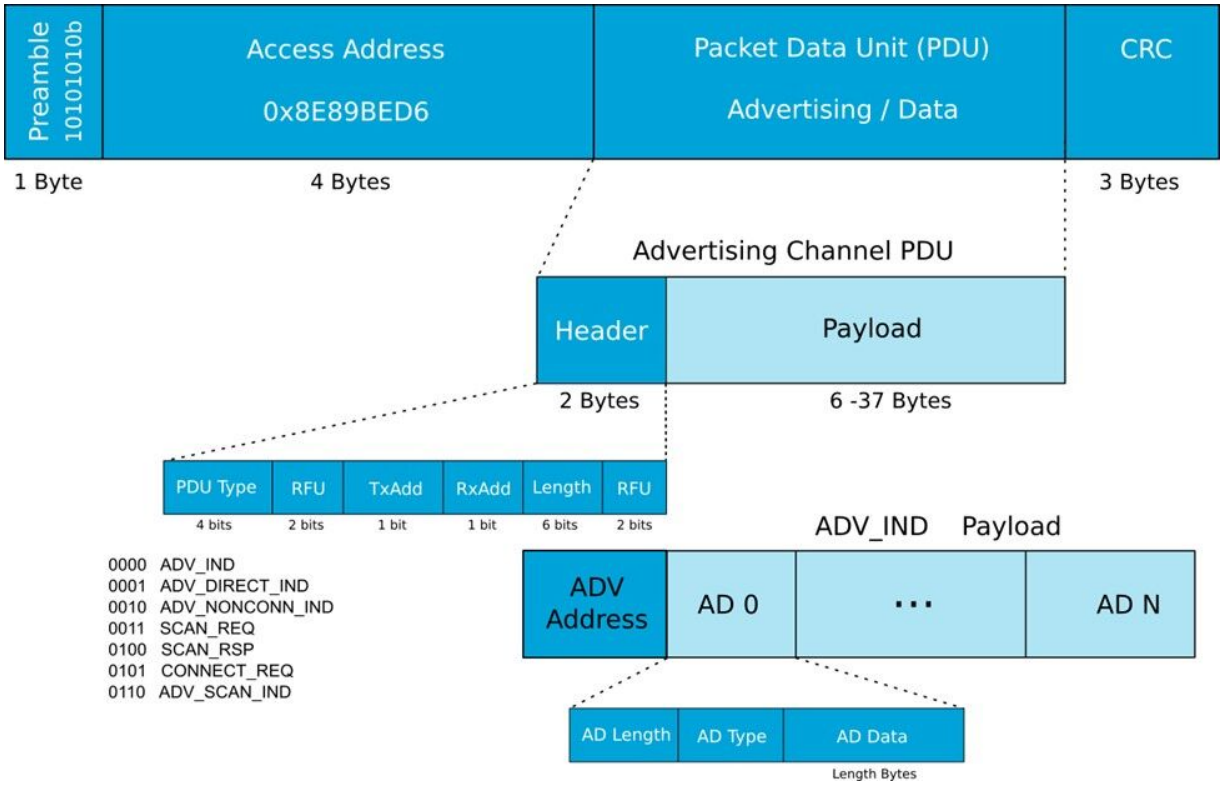
Basically as the name suggest it deals with the linking of the two devices. It is responsible for advertising, scanning and creating and maintaining connection. It also ensures the correct structure of the packets. There are two types of link layer channels: i) Advertising channels. ii) Data channels.

As discussed earlier the 2.4 GHz frequency is divided into 40 channels. Out of which 3 are advertising channels and 37 are data channels.

Process:

1) The 3 channels are used for sending advertising packets so that the central device will know the advertising devices(Peripheral Devices).
2) After establishing connection the data are sent over the data channels encapsulated in the packet .

Both the data packets and advertising packets are having similar structure. But the payload differs for both advertising packets and data packets. The packet structure is a below:



Host / Controller Interface:

Consider a situation when the Host is running on main CPU and the Controller is located on the separate hardware chip connected via a **UART(Universal Asynchronous Receiver/Transmitter)** or **USB.** In this case if Host wants to Communicate with Controller then it will require some Interface. The Host / Controller Interface serves as the Interface. This was most commonly used in **Bluetooth Classic.**

Host:

The Host Layer consist of :

- **Logical Link Control and Adaption Protocol(multiplexing layer)**
- **Security Manager(Authentication and setting up secure connections)**
- **Attribute Protocol(Exposes state data of a device)**
- **Generic Attribute Profile(Defines how the Attribute Protocol is used to expose data)**
- **Generic Access Profile(defines how devices found and connect with each other)**

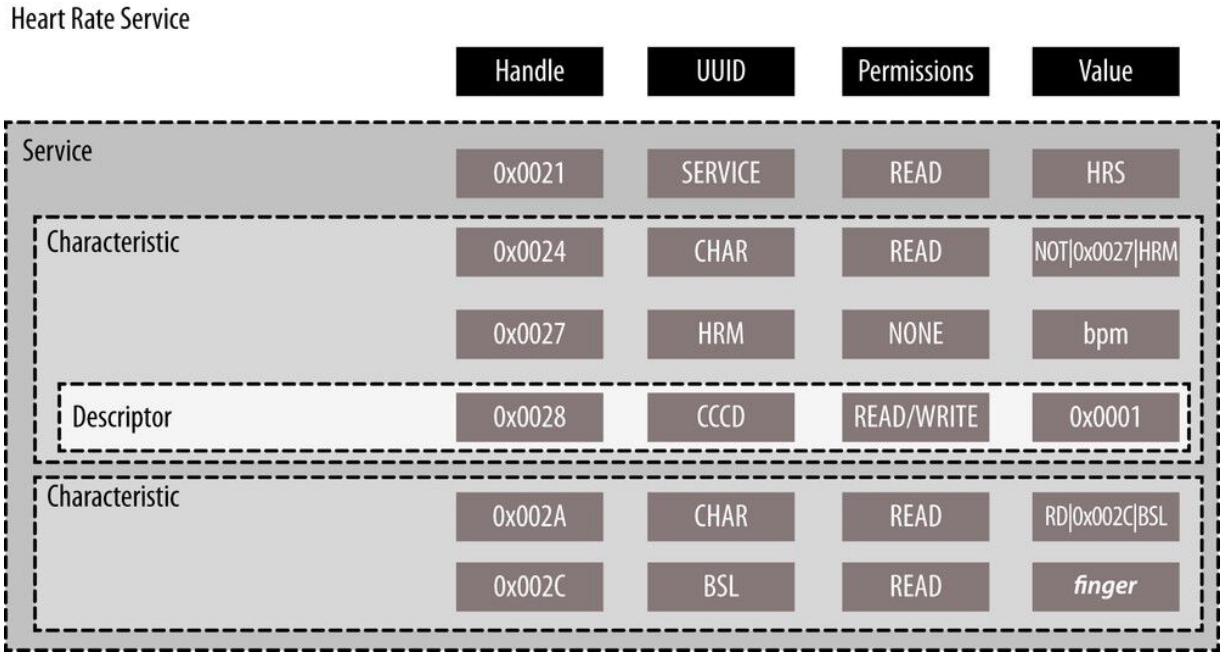
Logical Link Control and Adaption Protocol:

- It takes multiple protocols from the upper layers and encapsulates them in standard BLE Packet format.(Multiplexing)
- Fragmentation and Recombination

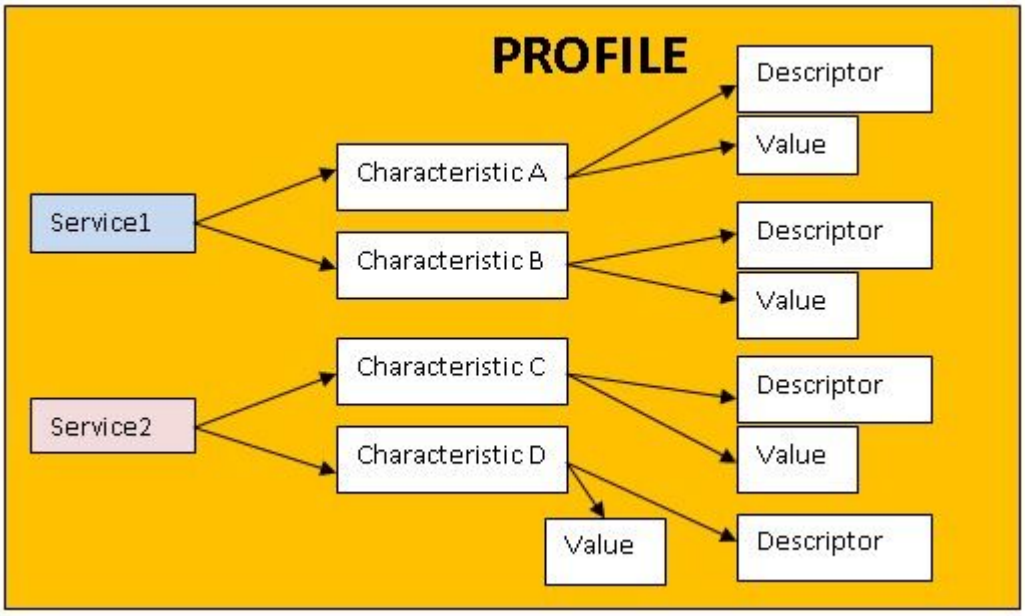
The L2CAP layer is used for routing the protocols like ATT(Attribute Protocol) and SMP(Security Manager Protocol)

Attribute Protocol:

Attribute protocol is used to define set of rules for accessing data on peer devices. Attribute protocol is used by client to access data stored on Attribute server in the form of '**Attributes'.** Each attribute is having a **Attribute handle** which is simply an identifier used for accessing the attribute value and universally unique identifier(UUID), set of permissions and value. UUID is used to specify the type and the nature of the data in the value.



GATT(Generic Attribute Profile):



GATT is on the top of ATT. It adds data model and hierarchy. The Data in the GATT is organized in the form of **Services.** Each Service contains one or more characteristics and each characteristics consist of data along with some metadata(descriptive information about the data). GATT services are organized in something we call as GATT Profile. Each GATT Profile can have multiple services. Each service is having 16-bit UUID in the same way characteristics are also having unique UUID.

GAP(Generic Access Profile):

Before discussing GAP we will look in to the roles a single device can support(**One device can have multiple roles at some instance**).

- **Broadcaster(Transmitter only) Eg., iBeacons**
- **Observer(Receiver only)**
- **Peripheral(Supports slave role)**
- **Central(Support master role, support multiple connections, initiates connection to peripherals)**

The Generic Access Profile is used for device discovery, connection and present useful information to user. Advertising in GAP can be in two ways:

- **Advertising Data Packet**(Mandatory to Send by Peripheral devices)
- **Scan Response Payload**(Only sent when requested by Central Devices)

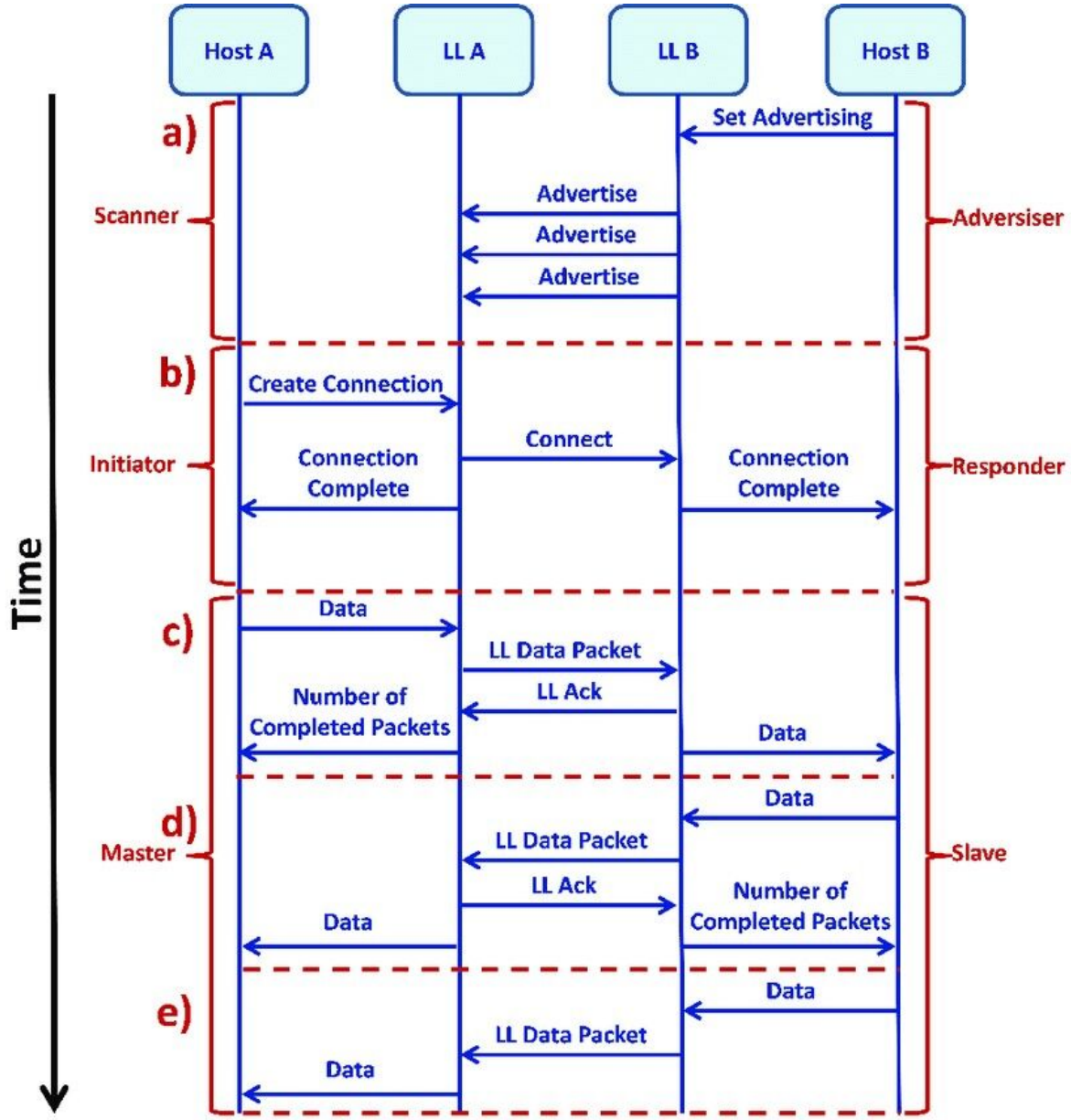
The Advertising Packet format is same as discussed above in link layer. The Peripheral devices advertise and the Central devices scan. The central devices on

the basis of the advertising information decides the appropriate peripheral device. Now, the central device will send the connection parameters to the peripheral.

Connection Parameters:

- **Connection Interval**
- **Slave Latency**
- **Supervision Timeout**

Connection Interval is the time interval in which the data will be sent.
Slave Latency is the measure of the ability of the slave to skip connection event. If the slave is not having any data to send then the connection request from the client will be refused.
Supervision Timeout is the maximum time between two successful connection events. If there occurs no successful connection event in this interval then the connection is considered lost



Communication in BLE