UG103.14: *Bluetooth*® LE Fundamentals

This volume of Silicon Labs' *Fundamentals* series provides an overview of Bluetooth low energy technology. Traditional Bluetooth technology is optimized for sending a steady stream of high quality data in a power-efficient way. Bluetooth low energy technology allows for short bursts of long-range radio connections, making it ideal for applications that depend on long battery life and don't need high throughput streaming data. This overview focuses on Bluetooth low energy technology, but also calls out some of the contrasts with traditional Bluetooth technology.

Silicon Labs' *Fundamentals* series covers topics that project managers, application designers, and developers should understand before beginning to work on an embedded networking solution using Silicon Labs chips, networking stacks such as EmberZNet PRO or Silicon Labs Thread, and associated development tools. The documents can be used as a starting place for anyone needing an introduction to developing wireless networking applications, or who is new to the Silicon Labs development environment.

KIT FEATURES

- Background
- Architecture
- Physical Layer
- Link Layer operations and network topologies
- Generic Access Profile (GAP)
- Attribute Protocol (ATT)
- Generic Attribute Profile (GATT)
- Security Manager

1. Background

Silicon Labs is developing products designed to meet the demands of customers as we move to an ever-connected world of devices in the home, what is often referred to as the IoT (Internet of Things). At a high level the goals of IoT for Silicon Labs are to:

- Connect all the devices in the home with best-in-class networking, whether with ZigBee PRO, Thread, Bluetooth low energy technology, or other emerging standards.
- Leverage the company's expertise in energy-friendly microcontrollers.
- Enhance established low-power, mixed-signal chips.
- Provide low-cost bridging to existing Ethernet and Wi-Fi devices.
- Enable cloud services and connectivity to smartphones and tablets that promote ease of use and a common user experience for customers.

Achieving all of these goals will increase adoption rates and user acceptance for IoT devices in the Connected Home.

Bluetooth technology is a core component of the IoT. Bluetooth was designed to offer a wireless alternative to cable connections by exchanging data using radio transmissions. One of the most popular applications for Bluetooth has been wireless audio. This uses a version of Bluetooth called BR/EDR (Bit Rate/Enhanced Data Rate) that is optimized for sending a steady stream of high quality data in a power-efficient way.

Bluetooth version 4.0 introduced Bluetooth with low energy functionality. Developers are now able to create sensors that can run on coin-cell batteries for months and even years. Some of these sensors are so efficient that the kinetic energy from flipping a switch can provide operating power. Bluetooth low energy technology is inherently different from BR/EDR. BR/EDR establishes a relatively short-range, continuous wireless connection, which makes it ideal for uses such as streaming audio from a smartphone to a headset. Bluetooth low energy technology allows for short bursts of long-range radio connections, making it ideal for IoT applications that depend on long battery life. Furthermore, Bluetooth low energy technology is built on an entirely new development framework using GATT (Generic Attributes). GATT profiles describes a use case, roles, and general behaviors based on the GATT functionality. Theses profiles allow developers to quickly and easily develop applications to connect devices directly to applications running on smartphones, PCs, or tablets.

Bluetooth devices can be either dual mode, supporting both BR/EDR and Bluetooth low energy technology, or single mode, supporting Bluetooth low energy technology only.

As well as ultra-low power and connectivity to smartphones, PCs, and tablets, other benefits of Bluetooth low energy technology include:

- Low cost
- Reliable and robust: AFH (Adaptive Frequency Hopping), retransmissions and 24-bit CRC (Cyclic Redundancy Checks)
- Secure: pairing, bonding, privacy, MITM (Man in the Middle) protection, and AES-128 encryption
- Supports rapid development:
  - Standardized profiles to cover key use cases (HR, HID, Glucose, Proximity, etc.)
  - Profiles can be developed as applications, supporting fast deployment
  - Vendor-specific profiles omit the need to wait for Bluetooth SIG to standardize profiles and operating system developers to integrate them
- Widely deployable: Supported by major platforms - iOS, Android 4.3 and newer, Windows 8 and 10, OSX, and Linux

The Bluetooth specification is managed by the Bluetooth SIG (special interest group). The SIG maintains a website (https://www.bluetooth.com) that contains both introductory information and links to specifications and other more technical details. In this document revisions of the specification are referred to parenthetically, where (BT5.0) means version 5.0 of the specification.

This document provides an overview of the following aspects of Bluetooth low energy:

- Bluetooth architecture overview
- Radio features
- Basic of link layer
- Explanation how device discovery and connections work
- Bluetooth security overview
- The Attribute Protocol
- The Generic Attribute Profile (GATT) and Bluetooth profiles

2. Bluetooth Low Energy Architecture

The Bluetooth low energy architecture is illustrated in the following figure:

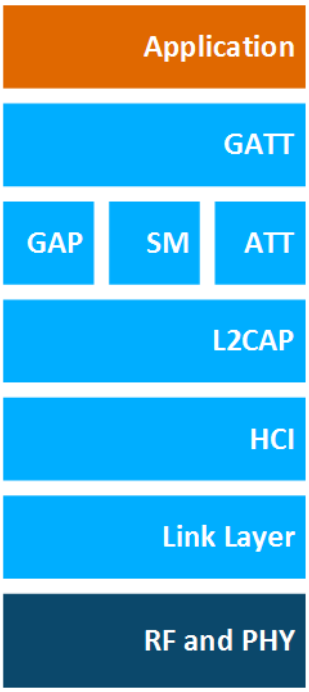


Figure 2.1. Bluetooth Low Energy Architecture

The components are as follows:

- Physical layer: Controls radio transmission/receiving.
- Link Layer: Defines packet structure, includes the state machine and radio control, and provides link layer-level encryption.

These two layers are often grouped into a Controller, with the remaining layers grouped into a host. A Host-to-Controller interface (HCI) standardizes communication between the controller and the host. The host layers are:

- L2CAP: Logical Link Control and Adaptation Protocol. L2CAP acts as a protocol multiplexer and handles segmentation and reassembly of packets. It also provides logical channels, which are multiplexed over one or more logical links. The L2CAP used in Bluetooth low energy technology is an optimized and simplified protocol based on the classic Bluetooth L2CAP. Typically, application developers do not need to care about the details of interacting with the L2CAP layer. The interaction is s handled by the Bluetooth stack, and the details of the L2CAP operation are not covered in this document.
- ATT: Attribute Protocol. The attribute protocol provides means to transmit data between Bluetooth low energy devices. It relies on a Bluetooth low energy connection and provides procedures to read, write, indicate and notify attribute values over that connection. ATT is used in most Bluetooth low energy applications and occasionally in BR/EDR applications.
- GATT: Generic Attribute Profile. The GATT is used to group individual attributes into logical services for example the Heart Rate Service, which exposes the operation of a heart rate sensor. In addition to the actual data the GATT also provides information about the attributes i.e. how they can be accessed and what security level is needed
- GAP: Generic Access Profile. The GAP layer provides means for Bluetooth low energy devices to advertise themselves or other devices, make device discovery, open and manage connections and broadcast data.
- SM: Security Manager. Provides means for bonding devices, encrypting and decrypting data and enabling device privacy.

These components are discussed in more detail in the following sections.

3. Physical Layer

Bluetooth low energy operates in the 2.4 GHz ISM (Industrial Scientific Medical) band (2402 MHz - 2480 MHz), which is license-free in most countries. The Bluetooth 4 specification defines 40 RF channels with 2 MHz channel spacing (see the following figure). Three of the 40 channels are advertising channels (shown in green), used for device discovery, connection establishment, and broadcast. The advertising channel frequencies are selected to minimize interference from IEEE 802.11 channels 1, 6 and 11, which are commonly used in several countries.

In the Bluetooth 5 specification the three advertisement channels highlighted below are called the primary advertisement channels. The 37 remaining channels are either used as secondary advertisement channels or data channels that can be used for additional advertisement data transmission.

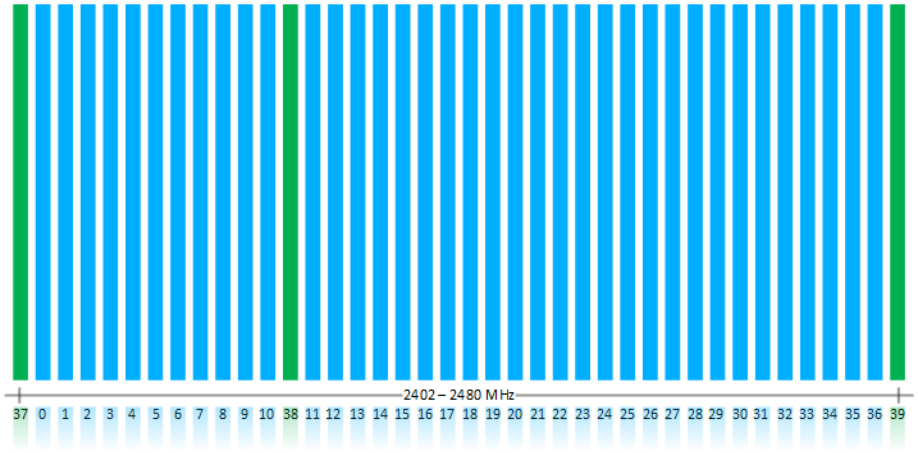


Figure 3.1. Bluetooth Low Energy Channels and Frequencies

Data channels are used for bidirectional communication between connected devices. AFH (Adaptive FHSS) is used to select a data channel for communication during a given time interval. AFH is reliable, robust, and adapts to interference.

All physical channels use GFSK (Gaussian Frequency Shift Keying) modulation, with a modulation index of 0.5, which allows reduced peak power consumption. In Bluetooth 4.0, 4.1 and 4.2 specification the physical layer data rate is 1 Mbps.

The Bluetooth 5 standard introduces an additional 2M PHY rate for faster throughput or shorter TX and RX times.

The recent changes in the Bluetooth and regulatory standards allow Bluetooth Smart devices to transmit up to 100 mW (20 dBm) transmit power. However not all countries allow the 10 0mW transmission power to be used because Bluetooth low energy radio can drop down to two RF channels when there is significant interference.

The requirements for a Bluetooth low energy radio are as follows:

Feature	Value
Minimum TX power	0.01 mW (-20 dBm)
Maximum TX power	100 mW (20 dBm)
Minimum RX sensitivity	-70 dBm (BER 0.1%)

The typical range for Bluetooth low energy radios is as follows:

TX power	RX sensitivity	Antenna gain	Range
0 dBm	-92 dBm	-5 dB	160 meters
10 dBm	-92 dBm	-5 dB	295 meters

The range to a smart phone is typically 0-50 meters due to limited RF performance of the phones.

4. Link Layer

The Bluetooth low energy link layer provides the first level of control and data structure over the raw radio operations and bit stream transmission and reception. For example, the link layer defines the following.

- Bluetooth state machine and state transitions
- Data and advertisement packet formats
- Link Layer operations
- Connections, packet timings, retransmissions
- Link layer level security

Application developers do not need to understand these in detail, but some essential concepts affect the application design, development, and the end device operation. Summaries of these concepts are provided in this section.

4.1 Link Layer Operations

This section describes the basic Bluetooth low energy link layer operations, including:

- Advertising
- Scanning
- Connection establishment

4.1.1 Advertisement

Advertisement is one of the most fundamental operations in Bluetooth low energy wireless technology. Advertisement provides a way for devices to broadcast their presence, allow connections to be established, and optionally broadcast data like the list of supported services, or the device name and TX power level.

A Bluetooth low energy device that is advertising broadcasts packets on one or multiple advertisement channels, which remote devices can then pick up.

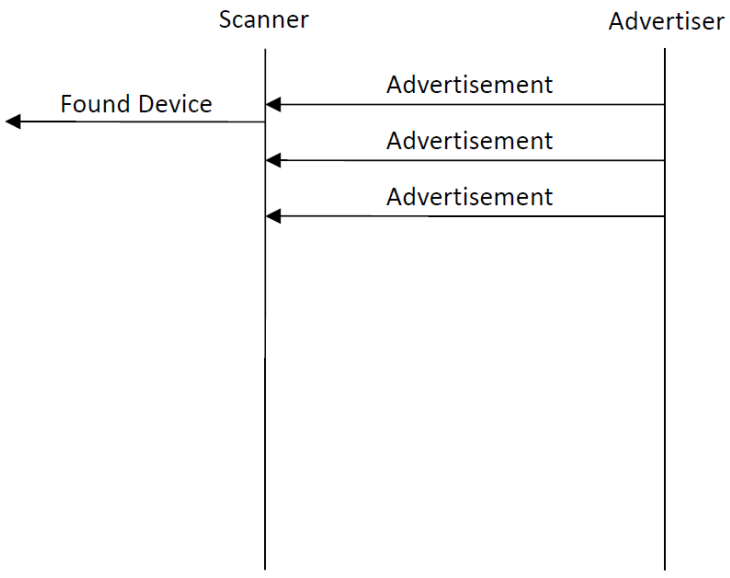


Figure 4.1. Bluetooth Low Energy Advertisement

The application typically has control of the following advertisement parameters.

Table 4.1. Advertisement Parameters

Parameter	Values	Description
Advertisement interval	20 ms to 10240 ms	Defines the interval between the advertisement events. Each event consist of 1 to 3 advertisement packets depending on the configuration. A random 0-10 ms is added by the link layer to every advertisement interval to help avoid packet collisions.
Advertisement channels	37, 38 and 39 (primary channels) 0-10 and 11-36 (BT 5 secondary channels)	The physical RF channels used to transmit the advertisement packets. For most reliable operation all channels should be used, but reducing the number of channels used will reduce power consumption at the cost of reliability.
Discoverability mode	Not discoverable Generic Discoverable Limited Discoverable Broadcast	Defines how the advertiser is visible to other devices.
Connectability mode	Not connectable Directed Connectable Undirected connectable	Defines if the advertiser can be connected or not
Payload	0 to 31 B (primary advertisement) 0 to 255 B (BT5 secondary advertisement)	0-31 bytes of data can be included in each primary advertisement packet. 0-255 bytes of data can be included in each secondary advertisement packet (Bluetooth 5)

4.1.2 Scanning

Scanning is the operation where a scanner is listening for incoming advertisement in order to discover, discover and connect, or simply to receive the data broadcast by the advertising devices.

Two types of scanning modes are supported: passive scanning (Figure 4.2 Passive Scanning on page 7) and active scanning (Figure 4.3 Active Scanning on page 8).

In passive scanning mode the scanner simply listens for incoming advertisement packets. The scanner cycles through each advertisement channel in a round-robin fashion, listening to one channel at a time.

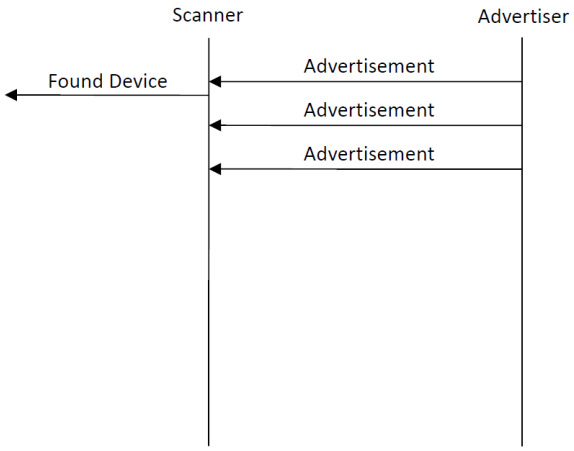


Figure 4.2. Passive Scanning

In active scanning mode the scanner listens for incoming advertisement packets and, upon receiving one, sends an additional scan request packet to the advertiser in order to learn more about it. Typically the scan response contains information like the list of supported services and friendly name, but the application has full control of the scan response data payload.

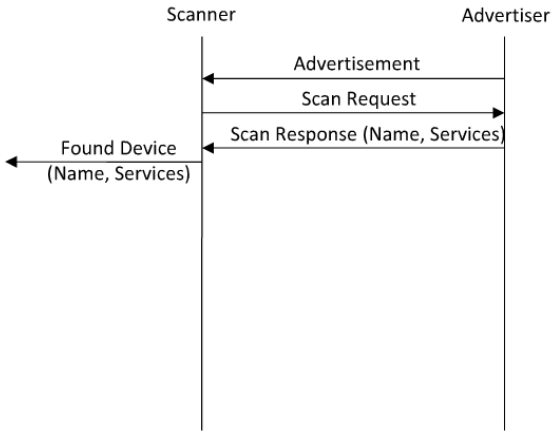


Figure 4.3. Active Scanning

The application typically controls the following scan parameters.

Table 4.2. Scan Parameters

Parameter	Values	Description
Scan interval	2.5 ms to 10240 ms	The interval is ms from the beginning of a scan event to a beginning of a consecutive scan event. Must be equal or larger than scan window.
Scan window	2.5 ms to 10240 ms	The scan window defines the duration of the listening (RX) window during a scan event.
Scan type	Limited Generic Observation	Defines which type of advertisers the scanner reports.
Scan mode	Active Passive	Defines if active or passive scanning is performed.
Connectability mode	Not connectable Directed Connectable Undirected connectable	Defines if the advertiser can be connected to or not

4.1.3 Connections

Connections allow application data to be transmitted in a reliable and robust manner, as Bluetooth low energy connections use CRCs, acknowledgements, and retransmissions of lost data to ensure correct data delivery. In addition, the Bluetooth low energy connections use Adaptive Frequency Hopping (AFH) to detect and adapt to the surrounding RF conditions and provide a reliable physical layer. Connections also support encryption and decryption of data to ensure its confidentiality.

The Bluetooth low energy connection always starts by a scanner receiving an advertisement packet that includes the fact that the advertiser allows connections. The following figure illustrates how Bluetooth low energy connection establishment happens.

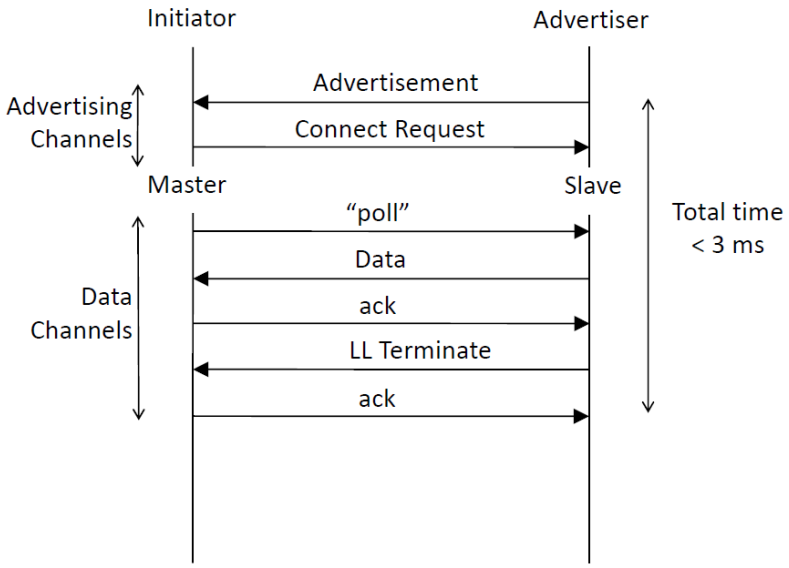


Figure 4.4. Connection Establishment, Transmission of One Packet, and Connection Termination

The application typically controls the following connection parameters.

Table 4.3. Connection Parameters

Parameter	Values	Description
Minimum Connection Interval	7.5 ms	Minimum allowed connection interval
Maximum Connection Interval	4000 ms	Maximum allowed connection interval
Connection (slave) latency	0 to 500 (connection intervals)	The amount of connection events the slave is allowed to skip if it has no data to send.
Supervision timeout	100 ms to 32000 ms	Defines how long the break in communications can be (for example due to out of range situation) before the connection is dropped and an error is presented to the user.

The connection parameters can be updated during the life time of a connection using a connection update message.

The connection event (Figure 4.5 Connection Timeline on page 10) starts when the master sends a packet to the slave at the defined connection interval. The slave can respond 150 µs after it has received a packet from the master. If the slave has no data to send it can skip a certain number of connection events defined by the slave latency parameter (Figure 4.5 Connection Timeline on page 10). If no packets are received by the master or slave within the time defined by the supervision timeout, the connection is terminated.

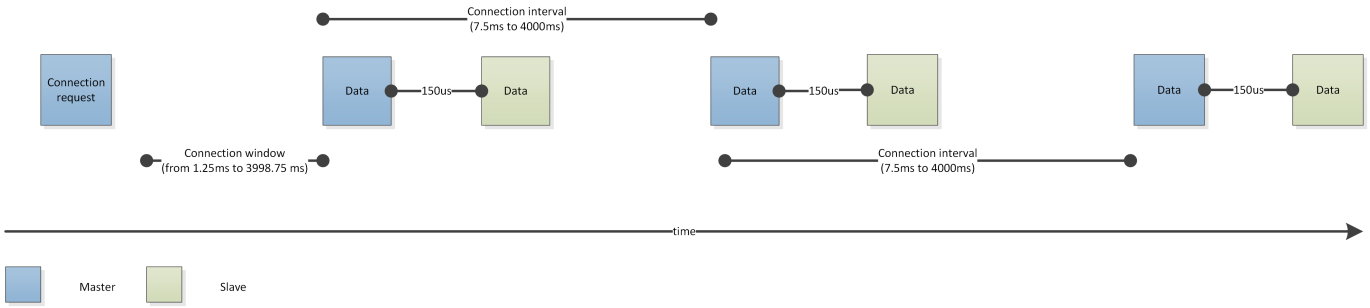


Figure 4.5. Connection Timeline

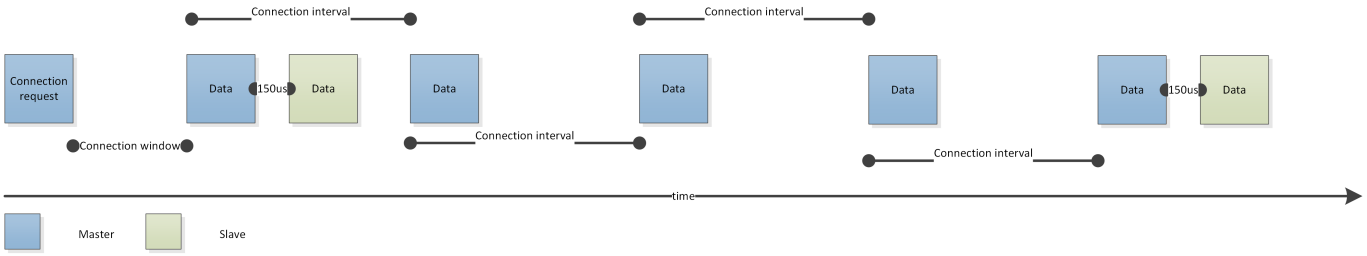


Figure 4.6. Slave Latency (latency=3)

If the slave has more data to send than can be fitted into a single packet, the connection event will automatically extend and the slave can send as many packets as there is time until the beginning of next connection interval. This can only be used with attribute protocol operations that do not require an acknowledgement.

4.2 Network Topologies

Device roles in Bluetooth low energy technology are:

- Advertiser: A device that broadcasts advertisement packets, but is not able to receive them. It can allow or disallow connections.
- Scanner: A device that only listens for advertisements. It can connect to an advertiser.
- Slave: A device connected to a single master (BT 4.0) or multiple masters (BT 4.1 and newer).
- Master: A device that is connected to one or more slaves. Theoretically a master can have an unlimited number of slave devices connected to it, but in practice the master can connect 4-20 slaves at a time.
- Hybrid: It is possible for a device to advertise and scan at the same time or be connected to a master and advertise or scan simultaneously. This is, however, vendor-specific, and the exact features that are supported should be checked with the vendor.

Examples of Bluetooth low energy topologies are shown in the following two figures.

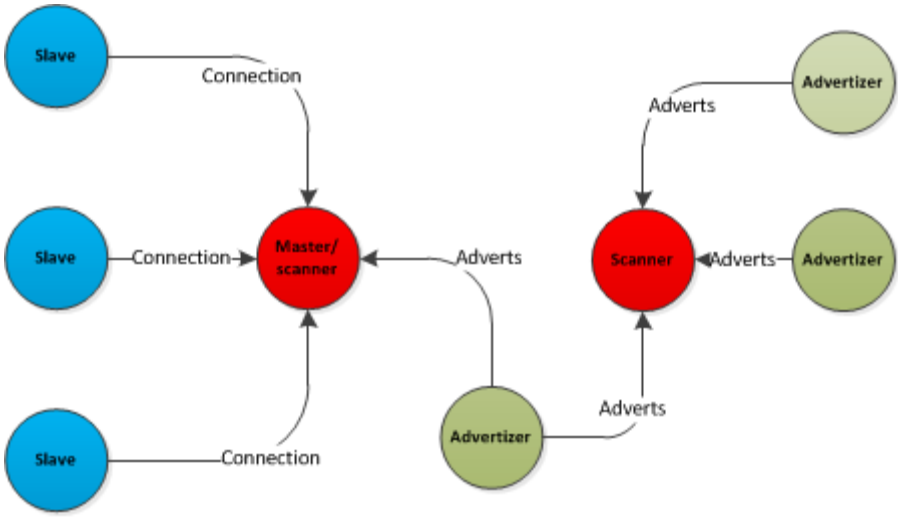


Figure 4.7. Bluetooth Low Energy Topologies

Devices can change roles and topologies, as illustrated in the following figure.

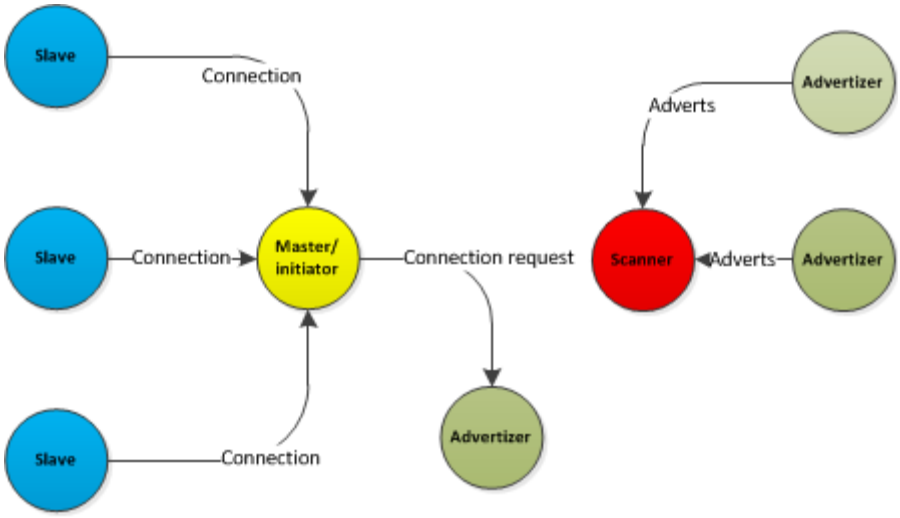


Figure 4.8. Topology and Role Change

5. Generic Access Profile (GAP)

The Generic Access Profile or the GAP is one of the first layers every Bluetooth low energy developer gets expose to. This is because the GAP is used to control how a device is visible and connectable by other devices and also how to discover and connect to remote devices.

To put this simply, the GAP provides access to the link layer operations described in section 4.1 Link Layer Operations, which are related to the device discovery, connection establishment and termination, and connection timing control.

GAP defines device roles that provide specific requirements for the underlying controller. Roles allow devices to have radios that either transmit (TX) only, receive (RX) only, or do both.

- Broadcaster (TX only): Sends advertising events and broadcast data.
- Observer (RX only): Listens for advertising events and broadcast data.
- Peripheral (RX and TX): Always slave, is connectable and advertising. Designed for a simple device using a single connection with a device in the Central role.
- Central (RX and TX): Always master, never advertises. Designed for a device that is in charge of initiating and managing multiple connections.

A device can support more than one role, but only one role can be adopted at a given time.

GAP also defines modes and procedures for discovery, connection, and bonding. The terminology is the same for Bluetooth low energy and BR/EDR, although underlying technology can differ.

Modes:

- Connectable: Can make a connection. State: Non-connectable, connectable.
- Discoverable: Can be discovered (is advertising). State: None, limited, general.
- Bondable: If connectable, will pair with connected device for a long-term connection. State: Non-bondable, bondable.

Procedures:

- Name discovery: Go into a menu and find the name of the other device. The name is shared with BR/EDR in a dual-mode device.
- Device discovery: Search for devices that are available for connection.
  - Find address and name of devices.
  - Define device role.
- Link establishment: After selecting an advertising device, connect to it.
  - Instruct Link layer to send a CONNECT_REQ.
  - Perform service discovery.
  - Request device authentication (not data authentication).
  - Request use of services.
- Service discovery: Used by devices in Central and Peripheral roles to find services available on peer devices.

6. Attribute Protocol (ATT)

Bluetooth low energy profiles expose a state of a device. The state is exposed as one or more values called attributes. The protocol to access these attributes is called the Attribute Protocol (ATT). The ATT defines the communication between two devices playing the roles of server and client, respectively, on top of a dedicated L2CAP channel. The Attribute protocol defines two roles:

- Server: The device that stores the data as one or more attributes
- Client: The device that collects the information for one or more servers

The client can access the server's attributes by sending requests, which trigger response messages from the server. For greater efficiency, a server can also send to a client two types of unsolicited messages that contain attributes: notifications, which are unconfirmed; and indications, which require the client to send a confirmation. A client may also send commands to the server in order to write attribute values. Request/response and indication/confirmation transactions follow a stop-and-wait scheme.

This section describes attributes and provides a summary of protocol methods.

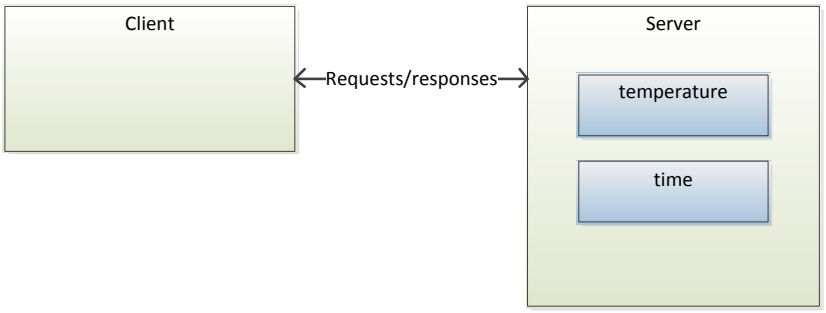


Figure 6.1. Device Roles

6.1 Attributes

Attributes are arrays that can vary from 0 to 512 bytes, as shown in the following example, and they can be fixed or variable length.

Example

Value
0x0000
0x426c7565676967612054656 3686e6f6c6f67696573

All attribute have handles, which are used to address an individual attribute, as shown in the following example. The client accesses the server's attributes using the handle.

Example

Handle	Value
0x0001	0x0000
0x0002	0x426c7565676967612054656 3686e6f6c6f6769657

Attributes also have a type, described by a UUID (Universally Unique Identifier), as shown in the following example. The UUID determines what the attribute value means.

Two types of UUIDs are used:

- Globally unique 16-bit UUID, defined in the characteristics specification (https://www.bluetooth.com/specifications/bluetooth-core-specification)
- Manufacturer-specific 128-bit UUIDs, which can be generated online (for example https://www.uuidgenerator.net/)

Example

Handle	UUID	Value	Description
0x0001	0x1804	0x0000	TX power as dBm
T0x0002	0x2a00	0x426c7565676967612054656 3686e6f6c6f6769657	Device name, UTF-8

Attributes also have permissions, which can be

- Readable / Not readable
- Writable / Not writable
- Readable and writable / Not readable and not writable

The attributes may also require the following:

- Authentication to read or write
- Authorization to read or write
- Encryption and pairing to read or write

The attribute types and handles are public information, but the permissions are not. Therefore, a read or write request may result an error, ‘Read/Write Not Permitted’ or ‘Insufficient Authentication’.

6.2 Attribute Protocol Operations

The Attribute Protocol is a stateless sequential protocol, meaning that no state is stored in the protocol and only one operation can be performed at a time.

The available Attribute Protocol methods are described in the following table:

Table 6.1. Attribute Protocol Methods

Method	Description	Direction
Find Information (starting handle, ending handle)	Used to discover attribute handles and their types (UUIDs)	Client -> Server
Find By Type Value (starting handle, ending handle, type, value)	Returns the handles of all attributes matching the type and value	Client -> Server
Read By Group Type (starting handle, ending handle, type)	Reads the value of each attribute of a given type in a range	Client -> Server
Read By Type (starting handle, ending handle, type)	Reads the value of each attribute of a given type in a range	Client -> Server
Read (handle)	Reads the value of given handle **Maximum payload : 250 bytes**	Client -> Server
Read Blob (handle, offset)	Can be used to read long attributes larger than 250 bytes **Maximum payload: 64 kB**	Client -> Server
Read Multiple ([Handle]*)	Used to read multiple values at the same time	Client -> Server
Write (handle, value)	Writes the value to the given handle, with no response **Maximum payload: 250 bytes**	Client -> Server
Prepare Write (handle, offset, value) and Execute (exec/cancel)	Prepares a write procedure, which is queued in server until the write is executed.	Client -> Server
Handle Value Notification (handle, value)	Server notifies client of an attribute with a new value **Maximum payload: 250 bytes**	Server -> Client
Handle Value Indication (handle, value)	Server indicates to client an attribute with a new value. Client must confirm reception. **Maximum payload: 250 bytes**	Server -> Client
Error response	Any request can cause an error and error response contains information about the error	Server -> Client

6.3 Acknowledgements

ATT operations can optionally require acknowledgements (ACKs). This allows the application to know what data packets have been successfully transmitted and can be used to design extremely reliable applications.

Because the server must wait for an ACK from the client, data throughput is affected.

Non-ACKed operations can be used in applications requiring high throughput, since multiple operations can be performed within a connection interval. The Link Layer still retransmits lost packets, so reliability is not affected, but the application cannot know which packets have been transmitted successfully.

Both operations are illustrated in the following figure.

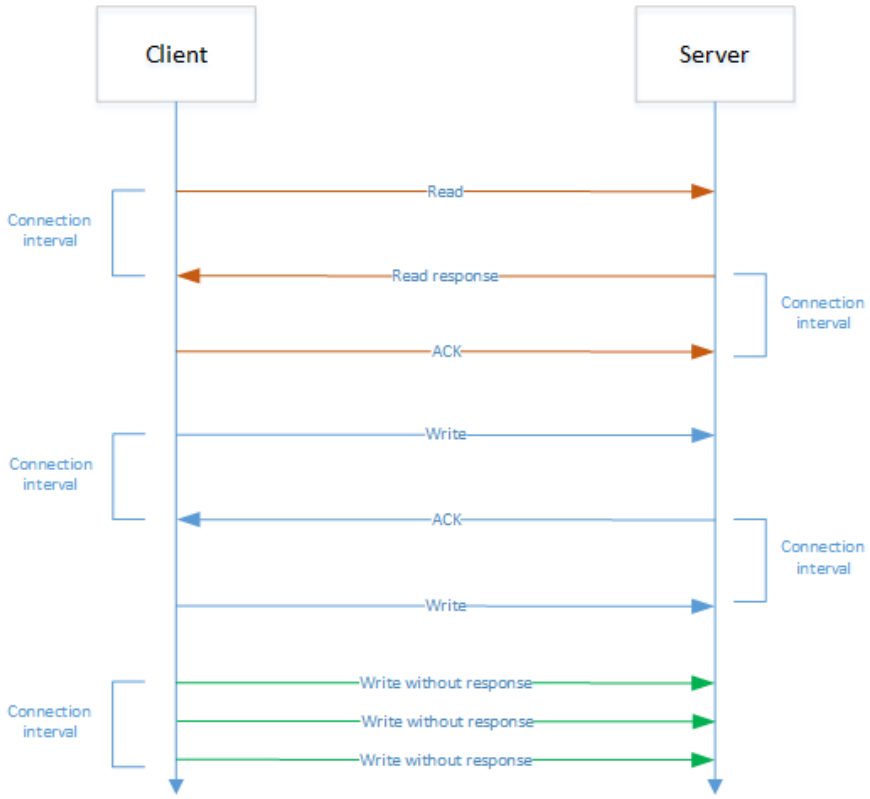


Figure 6.2. ACK and non-ACK Data Transfer

7. Generic Attribute Profile (GATT)

Generic Attribute Profile (GATT) is built on top of the Attribute Protocol (ATT) and establishes common framework for the data transported and stored by the Attribute Protocol. GATT defines two roles: Server and Client.

The GATT server stores the data transported over the Attribute Protocol and accepts ATT requests from the GATT client. The GATT server on the other hand sends responses to requests and when configured, sends indication and notifications to the GATT client when events occur on the GATT server. GATT also specifies the format of data contained on the GATT server.

Attributes, as transported by the Attribute Protocol, are formatted as services and characteristics. Services may contain a collection of characteristics. Characteristics contain a single value and any number of descriptors describing the characteristic value.

Bluetooth profiles specify the structure in which data is exchanged. The profile defines elements, such as services and characteristics, used in a profile, but it may also contain definitions for security and connection-establishment parameters. Typically a profile consists of one or more services that are needed to accomplish a high-level use case, such as heart-rate or cadence monitoring. Standardized profiles allow device and software vendors to build inter-operable devices and applications.

Bluetooth SIG standardized profiles are defined in profiles specifications. These are available at

https://developer.bluetooth.org/gatt/profiles/Pages/ProfilesHome.aspx

Services

Services are collections of data composed of one or more characteristics used to accomplish a specific function of a device, such as battery monitoring or temperature data, rather than a complete use case.

Standardized Bluetooth SIG are defined in service specifications, which are available at

https://developer.bluetooth.org/gatt/services/Pages/ServicesHome.aspx

Characteristics

A characteristic is a value used in a service, either to (1) expose and/or exchange data and/or (2) control information. Characteristics have a well-defined, known format. They also contain information about how the value can be accessed, what security requirements must be fulfilled, and, optionally, how the characteristic value is displayed or interpreted. Characteristics may also contain descriptors that describe the value or permit configuration of characteristic data indications or notifications.

Standardized characteristics are defined in the Characteristic Specification, which are available at

https://developer.bluetooth.org/gatt/characteristics/Pages/CharacteristicsHome.aspx

The figure below illustrates the relationship between GATT client, GATT server, services, characteristics and characteristics declaration, data and descriptors.

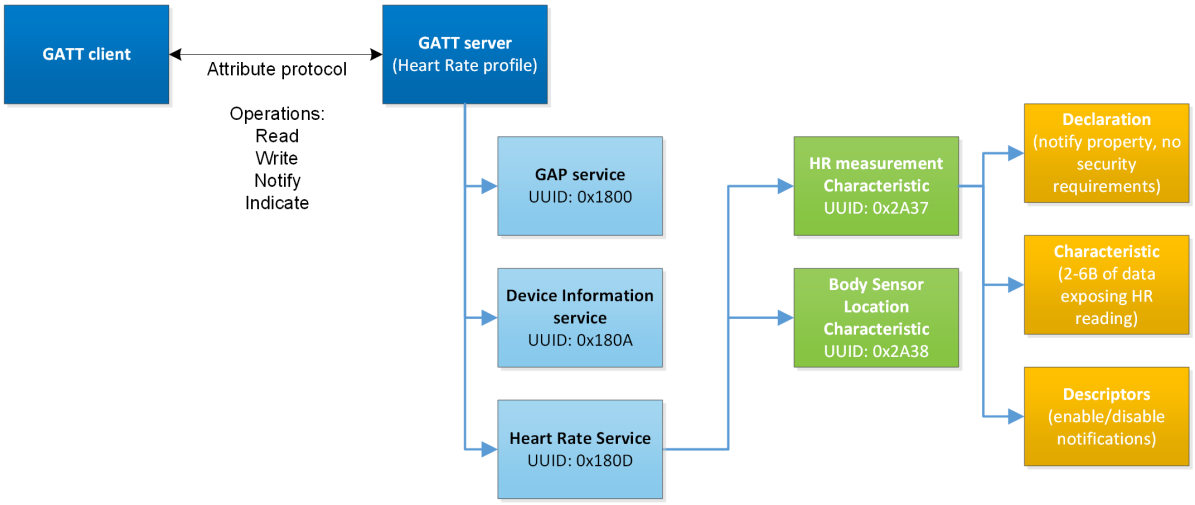


Figure 7.1. GATT Data Structure and Operation

8. Bluetooth Low Energy Security

To make sure the communication over Bluetooth low energy technology is always secure and protected, the technology provides several features to ensure the trust, integrity, privacy and encryption of the data.

The first section provides an overview of Bluetooth low energy security technology. Subsequent sections discuss the following topics in more detail:

- Pairing
- Encryption
- Privacy

8.1 What Protection Does Bluetooth Security Provide?

The Bluetooth specification defines security features to protect the user's data and identity. The security features used by Bluetooth low energy technology are either NIST compliant or FIPS approved.

Bluetooth low energy technology provides three basic security services:

- Authentication and Authorization: Establishing trusted relationships between devices
- Encryption and Data Protection: Protecting data integrity and confidentiality
- Privacy and Confidentiality: Preventing device tracking

The Bluetooth security model includes five security features:

- Pairing: the process for creating shared secret keys
- Bonding: storing the keys created during pairing so they can be used later
- Device authentication: verification of stored keys
- Encryption: data confidentiality
- Message integrity: protection against data alteration

The Security Manager is responsible for:

- Pairing
- Key distribution
- Generating hashes and short term keys

The Link Layer, on the other hand, is responsible for data encryption and decryption

The Bluetooth low energy security features provide protection against the following common threats in wireless communications.

Man-in-the-Middle Protection

A Man-in-the-Middle (MITM) attack requires the ability to monitor, alter or inject messages into a communications. This can for example be done with active eavesdropping where the attacker listens and relays messages between two parties who think are directly communicating with each other's over a private connection which is actually fully controlled by the attacker.

Bluetooth low energy technology provides protection against MITM attacks if the devices are paired either by using the passkey entry or out-of-band pairing method. Alternatively LE Secure connections and the numeric comparison pairing method can be used in devices that use Bluetooth 4.2 or newer standards.

Protection against Passive Eavesdropping

Passive Eavesdropping means that someone is passively listening (for example by using a sniffer) to the communication of others. To protect against passive eavesdropping LE Secure Connection uses ECDH public key cryptography, which provides a very high degree of strength against passive eavesdropping attacks as it allows the key exchange over unsecured channels.

Privacy Protection

Since most Bluetooth low energy devices have an address associated to them and the address is carried in the advertisement packets, it is possible to associate the address to devices in order to track them. The privacy feature in Bluetooth low energy technology and the frequently changing address can be used to protect against tracking.

8.2 Pairing and Bonding

The Bluetooth pairing is the process where the parties involved exchange their identity information in order to set up a trusted relationship and generate encryption keys used for data exchange. The Bluetooth low energy technology provides multiple options for pairing, depending on the security requirements of the application.

Bluetooth low energy standard versions 4.0 and 4.1 use the Secure Simple Pairing model, in which users can choose one method from Just Works, Passkey Entry and Out-of-Band mechanisms based on the input/output capability of the devices.

In Bluetooth low energy standard version 4.2 security is enhanced by the new LE Secure Connections pairing model, by adding a numeric comparison method, and by introducing the Elliptical Curve Diffie-Hellman (ECDH) key exchange algorithm.

The tables below summarize the association models that can be used between both parties depending on their supported I/O capabilities (BLUETOOTH SPECIFICATION Version 4.2 [Vol 1, Part A] 5.4.1 Association Models).

The term pairing means the generation and exchange of security keys, and the term bonding refers to the storage of the security keys so they can be used later.

	Initiator				
Responder	**DisplayOnly**	**Display YesNo**	**Keyboard Only**	**NoInput NoOutput**	**Keyboard Display**
Display Only	Just Works Unauthenticated	Just Works Unauthenticated	Passkey Entry: responder displays, initiator inputs Authenticated	Just Works Unauthenticated	Passkey Entry: responder displays, initiator inputs Authenticated
Display YesNo	Just Works Unauthenticated	Just Works (For LE Legacy Pairing) Unauthenticated	Passkey Entry: responder displays, initiator inputs Authenticated	Just Works Unauthenticated	Passkey Entry (For LE Legacy Pairing): responder displays, initiator inputs Authenticated
		Numeric Comparison (For LE Secure Connections) Authenticated			Numeric Comparison (For LE Secure Connections) Authenticated

	Initiator				
Responder	**DisplayOnly**	**Display YesNo**	**Keyboard Only**	**NoInput NoOutput**	**Keyboard Display**
Keyboard Only	Passkey Entry: initiator displays, responder inputs Authenticated	Passkey Entry: initiator displays, responder inputs Authenticated	Passkey Entry: initiator and responder inputs Authenticated	Just Works Unauthenticated	Passkey Entry: initiator displays, responder inputs Authenticated
NoInput NoOutput	Just Works Unauthenticated	Just Works Unauthenticated	Just Works Unauthenticated	Just Works Unauthenticated	Just Works Unauthenticated
Keyboard Display	Passkey Entry: initiator displays, responder inputs Authenticated	Passkey Entry (For LE Legacy Pairing): initiator displays, responder inputs Authenticated Numeric Comparison (For LE Secure Connections) Authenticated	Passkey Entry: responder displays, initiator inputs Authenticated	Just Works Unauthenticated	Passkey Entry (For LE Legacy Pairing): initiator displays, responder inputs Authenticated Numeric Comparison (For LE Secure Connections) Authenticated

8.3 Encryption

Bluetooth low energy technology uses AES-CCM cryptography for encryption and the encryption is performed by the Bluetooth low energy controller. This encryption function generates 128-bit encrypted using the AES-128-bit block cypher as defined in FIPS-1971.

8.4 Privacy

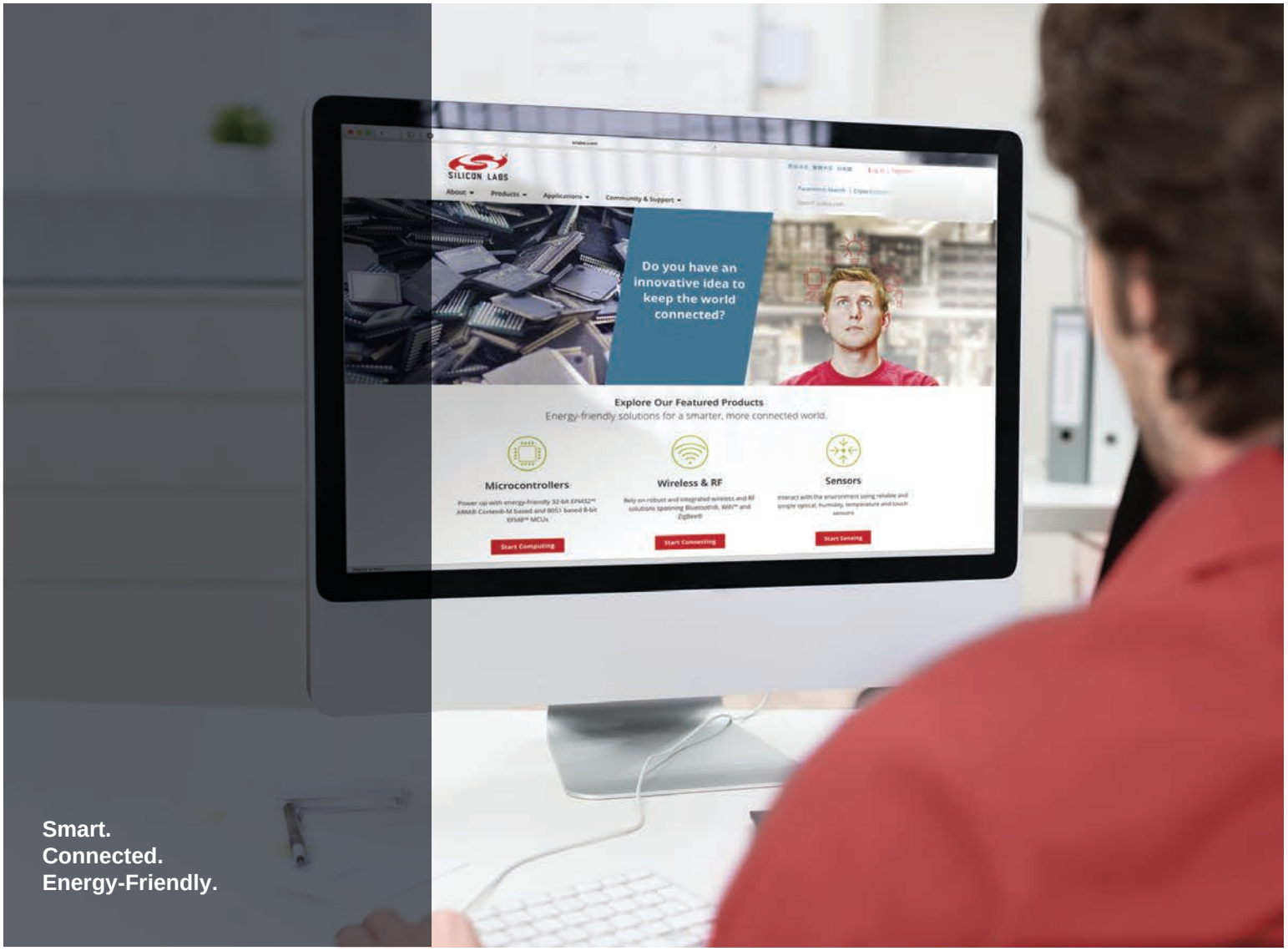
Bluetooth low energy technology also supports a feature that reduces the ability to track a Bluetooth low energy device over a period of time. This is achieved by changing the Bluetooth device address on a frequent basis. The changing address is called the public address and the bonded devices are able to resolve the private (non-changing) address from the public address.

In order to resolve the private address the devices need to be previously bonded. The public address is generated using the device's IRK (Identity Resolving Key).exchanged during the previous pairing or bonding procedure.

In Bluetooth 4.0 and 4.1 standards the private addresses are resolved and generated at the host. In Bluetooth 4.2 and newer standards the private addresses are resolved and generated at the controller without involving the host.

9. Next Steps

See *QSG139: Bluetooth Development with Simplicity Studio* for instructions on how to install the necessary tools and SDKs to get started with development using the Simplicity Studio IDE and other tools.



Products
www.silabs.com/products

Quality
www.silabs.com/quality

Support and Community
community.silabs.com

SILICON LABS

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

http://www.silabs.com