ModusToolbox™ Bluetooth® Configurator user guide

ModusToolbox™ tools package version 3.9
Bluetooth® Configurator version 3.70.0

About this document

A newer revision of this document may be available on the web here.

Scope and purpose

The Bluetooth® Configurator settings include options to specify what services and profiles to use and what features to offer by creating the required databases in generated code.

Intended audience

This document helps application developers understand how to use the Bluetooth® Configurator as part of creating a ModusToolbox™ application.

Document conventions

Convention	Explanation
Bold	Emphasizes heading levels, column headings, menus and sub-menus
Italics	Denotes file names and paths.
`Monospace`	Denotes APIs, functions, interrupt handlers, events, data types, error handlers, file/folder names, directories, command line inputs, code snippets
File > New	Indicates that a cascading sub-menu opens when you select a menu item

Abbreviations and definitions

The following define the abbreviations and terms used in this document:

- Application – One or more projects related to each other
- Configurator – A GUI-based tool used to configure a resource
- GAP – generic access profile
- GATT – generic attribute profile
- L2CAP – logical link control and adaptation protocol
- LE – low energy
- MCU – microcontroller unit
- Resource – Includes peripherals, pins, clocks, etc. used in an application
- SDK – software development kit
- SIG – special interest group

Reference documents

Refer to the following documents for more information as needed:

- Eclipse IDE for ModusToolbox™ user guide
- VS Code for ModusToolbox™ user guide
- Device Configurator user guide
- AIROC™ Bluetooth® SDK documentation
- PSoC™ 6 Bluetooth® Middleware API reference guide
- Device datasheets
- Device technical reference manuals

Table of contents

About this document . . . 1

Table of contents . . . 2

1 Overview . . . 3

1.1 Supported devices and libraries . . . 3

1.2 SIG-adopted and custom profiles and services . . . 4

2 Launch the Bluetooth® Configurator . . . 5

2.1 make command . . . 5

2.2 VS Code and Eclipse . . . 5

2.3 Executable (GUI) . . . 5

2.4 Executable (CLI) . . . 6

3 Quick start . . . 8

4 Code generation . . . 9

4.1 Option1 . . . 9

4.2 Option2 . . . 9

4.3 Option3 . . . 9

4.4 Option4 . . . 9

5 GUI description . . . 10

5.1 Main toolbar . . . 10

5.2 Tabs and tab components . . . 10

5.3 Notice List . . . 11

5.4 Status bar . . . 11

6 Parameters configuration . . . 12

6.1 General tab (Option1, Option3) . . . 12

6.2 General tab (Option4) . . . 14

6.3 General LE tab (Option2, Option3) . . . 15

6.4 GATT Settings tab (Option1..4) . . . 17

6.5 GAP Settings tab (Option2, Option3, Option4) . . . 31

6.6 L2CAP Settings tab (Option2, Option3, Option4) . . . 48

6.7 Link Layer Settings tab (Option4) . . . 49

6.8 Service Discovery Settings tab (Option1, Option3) (Beta) . . . 50

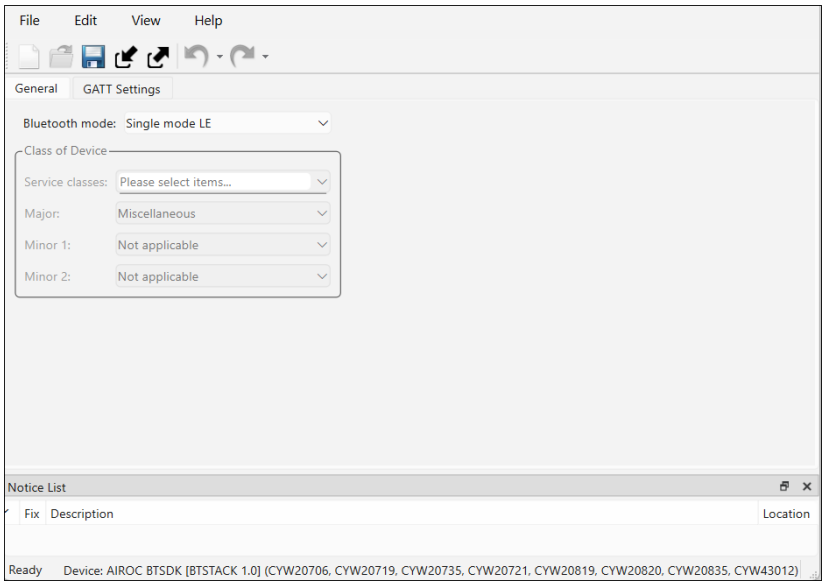
7 Version changes . . . 52

Revision history . . . 56

Disclaimer . . . 57

1 Overview

The Bluetooth® Configurator is a stand-alone graphical tool included with the ModusToolbox™ software. This configurator helps generate code for Bluetooth® applications, including the generic attribute profile (GATT) database, service discovery protocol (SDP) database (provided at Beta-level support), generic access profile (GAP) configuration, logical link control and adaption protocol (L2CAP), and link layer parameters.



1.1 Supported devices and libraries

The Bluetooth® Configurator supports various devices and libraries (also known as middleware) – the following table summarizes these options. The Configurator generates different code for each of the supported devices.

Device type	Library name	Source	Documentation	Parameter(--stack command line option)
CYW20706 CYW20719 CYW20721 CYW20735 CYW20819 CYW20820 CYW20835 CYW43012	AIROC™ Bluetooth® SDK (BTSTACK 1.0)	CYW20706 CYW20719 CYW20721 CYW20735 CYW20819 CYW20820 CYW20835 CYW43012	BTSDK Doc	Option1
CYW20829 PSoC™ 6 MCU with CYW43xxx Connectivity PSoC™ 63	AIROC™ BTSTACK with Bluetooth® LE only (Updatable Stack)	BTSTACK Bluetooth® FREERTOS	Bluetooth® LE API Doc Dual Mode API Doc	Option2

Device type	Library name	Source	Documentation	Parameter(--stack command line option)
CYW43022 CYW5551x CYW5591x CYW5557x	AIROC™ Bluetooth® SDK (BTSTACK 3.x)	CYW43022 BTSDK CYW55572 BTSDK	CYW43022 BTSDK Doc CYW55572 BTSDK Doc	Option3
PSoC™ 63	PSoC™ 63 Bluetooth® LE only Legacy Stack	Bluetooth® LE Middleware	Bluetooth® LE Middleware Doc	Option4

Parameter column options – **Option1**, **Option2**, **Option3**, **Option4** – are used throughout this document as references to corresponding Middleware (Library) types.

***Note**: Starting from version 2.70 (this document), **Option4** support is deprecated. New configurations for the PSoC™ 63 devices are based on AIROC™ BTSTACK with the Bluetooth® LE only library, the same as the CYW20829 devices. But you can still open, edit, and save the existing **Option4** configurations and perform code generation.*

***Note**: For command line options description, refer to Command line options for GUI executable and Command line options for CLI executable.*

1.2 SIG-adopted and custom profiles and services

The Bluetooth® Configurator supports numerous SIG-adopted profiles and services. Also, the Bluetooth® Configurator generates all necessary code for a particular profile/service operation in accordance with its configuration.

Also, you can create custom Bluetooth® Low Energy (LE) Profiles that use existing services as well as create custom services with custom characteristics and descriptors.

2 Launch the Bluetooth® Configurator

There are several ways to launch the Bluetooth® Configurator, and those ways depend on how you use the various tools in the ModusToolbox™ ecosystem.

2.1 make command

As described in the ModusToolbox ™ tools package user guide "ModusToolbox™ build system'' chapter, you can run numerous make commands in the application directory, such as launching the Bluetooth® Configurator. After you have created a ModusToolbox™ application, navigate to the application directory and type the following command in the appropriate bash terminal window:

```
make bt-configurator
```

If the application has an existing Bluetooth® configuration (*.*cybt*) file, this command opens the Bluetooth® Configurator GUI for the specific application in which you are working.

If there is no *.*cybt* file, this command launches an empty Bluetooth® Configurator. You must Open an existing configuration file or Create a new configuration file for the application in which you want to configure Bluetooth®.

2.2 VS Code and Eclipse

VS Code and Eclipse have tools to launch the Bluetooth® Configurator from within an open application. Refer to the applicable user guide for more details:

- VS Code for ModusToolbox™ user guide
- Eclipse IDE for ModusToolbox™ user guide

2.3 Executable (GUI)

To create an application or just to see what the Bluetooth® Configurator looks like, launch the Bluetooth® Configurator GUI by running its executable as appropriate for your operating system – double-click it or select it using the Windows **Start** menu. By default, it is installed here:

<install_dir>/ModusToolbox/tools_<version>/bt-configurator<version>

When opened this way, the Bluetooth® Configurator GUI opens blank without any information.



You must open an existing *.*cybt* file or create a new one for the application in which you want to configure Bluetooth®.

2.3.1 Open an existing configuration file

To open an existing configuration file, select **File > Open** ([**Ctrl**]+[**O**]). In the "Open Configuration File" dialog, navigate to the appropriate directory, and select the desired configuration file (*.*cybt*).

2.3.2 Create a new configuration file

To create a new configuration file, select **File > New** ([**Ctrl**]+[**N**]). The Device dialog drop-down menu displays the following options:

- AIROC™ BTSDK [BTSTACK 1.0] (CYW20706, CYW20719, CYW20721, CYW20735, CYW20819, CYW20820, CYW20835, CYW43012)
- AIROC™ BTSTACK with Bluetooth® LE only (CYW20829, PSoC™ 63, PSoC™ 6 with CYW43xxx Connectivity device)
- AIROC™ BTSDK [BTSTACK 3.x] (CYW43022, CYW5551x, CYW5591x, CYW5557x)
- Not for new design: PSoC™ Bluetooth® LE only Legacy Stack (PSoC™ 63)

When a new configuration is created as a ModusToolbox™-generated application based on a supported device(s) (Supported devices and libraries), Bluetooth® Configurator tries to detect the proper Middleware type (**Option1..4**) for that device(s). If the application is created successfully, the Device dialog does not display and the relevant Option is selected automatically. If auto-detection fails, you can still select the desired Middleware library in the displayed Device dialog manually.

***Note**: **Option4** is not recommended for creating new configuration.*

Various parameter configuration options depend on the selected device to generate code for.

2.3.3 Command line options for GUI executable

The GUI executable accepts the following command line options:

Option name	Short name	Description	Valid values
`--help`	`-?, -h`	Displays help on command line options	
`--help-all`		Displays help including Qt specific options	
`--version`	`-v`	Displays version information	
`--config <file>`	`-c <file>`	The path to the config file	
`--app_dir <dir>`		Opens all files in the context of the provided application/ project	
`--output-dir <dir>`	`-o <dir>`	The path to the generated source directory. It is either an absolute path or a path relative to the configuration file parent directory.	
`--stack <stack>`	`-s <stack>`	Middleware library (Section 1.1)	Option1 Option2 Option3 Option4

2.4 Executable (CLI)

The Bluetooth® Configurator executable can be run from the command line, and it also has a "cli" version of the executable. Running the configurator executable from the command line can be useful as part of batch files or shell scripts to re-generate the source code based on the latest configuration settings. The exit code for the

executable is zero if the operation is successful, or non-zero if the operation encounters an error. For more information about the command-line options, run the executable using the `-h` option.

2.4.1 Command line options for CLI executable

The CLI executable accepts the following command line options:

Option Name	Short Name	Description
`--help`	`-?, -h`	Displays help on command line options
`--help-all`		Displays help including Qt specific options
`--version`	`-v`	Displays version information
`--config <file>`	`-c <file>`	The path to the config file
`--app_dir <dir>`		Opens all files in the context of the provided application/project
`--output-dir <dir>`	`-o <dir>`	The path to the generated source directory. It is either an absolute path or a path relative to the configuration file parent directory.
`--clean`		Removes the generated files from the output folder

3 Quick start

1. Launch the Bluetooth® Configurator.
2. Use the Bluetooth® Configurator to configure the application (**GATT**, **GAP**, **L2CAP**, **LL**, **SDP**, etc.). Refer to the Parameters configuration section for more details.
3. Save the configuration file and generate source code.

4 Code generation

The Bluetooth® Configurator generates code into a *GeneratedSource* directory in your Eclipse IDE application or into the specified location for non-Eclipse IDE applications. That directory contains the necessary source (.c) and header (.h) files for the generated firmware, which uses the relevant driver APIs to configure hardware. The user configuration is saved to the *.*cybt* file.

4.1 Option1

- *design.cybt* – Contains an XML representation of the configuration; you may specify any file name
- *cycfg_gatt_db.h* and *cycfg_gatt_db.c* – Contain the GATT database code., include *cycfg_gatt_db.h* in your application
- *cycfg_sdp_db.h* and *cycfg_sdp_db.c* – Contain the SDP database code, include *cycfg_sdp_db.h* in your application.

4.2 Option2

- *design.cybt* – Contains an XML representation of the configuration. You may specify any file name.
- *cycfg_gatt_db.h* and *cycfg_gatt_db.c* – Contain the GATT database code. Include *cycfg_gatt_db.h* in your application.
- *cycfg_gap.h* and *cycfg_gap.c* – Contain the GAP configuration code.
- *cycfg_bt_settings.h* and *cycfg_bt_settings.c* – Contain the `wiced_bt_cfg_settings_t initialization` structure. Two sets of structures are generated (using the `#if` directive): compatible with BTSTACK 2.X and BTSTACK 3.X. Include *cycfg_bt_settings.h* in your application.

4.3 Option3

- *design.cybt* – Contains an XML representation of the configuration. You may specify any file name.
- *cycfg_gatt_db.h and cycfg_gatt_db.c* – Contain the GATT database code. Include *cycfg_gatt_db.h* in your application.
- *cycfg_gap.h* and *cycfg_gap.c* – Contain the GAP configuration code.
- *cycfg_bt_settings.h* and *cycfg_bt_settings.c* – Contain the structures necessary to initialize wiced_bt_cfg_settings_t structure. The structure `wiced_bt_cfg_settings_t` itself is not generated. Include *cycfg_bt_settings.h* in your application.
- *cycfg_bt_settings.h* and *cycfg_bt_settings.c* – Contain the structures necessary to initialize the `wiced_bt_cfg_settings_t` structure, which consists of names starting with `cy_ prefix`. Use either a complete generated `cy_bt_cfg_settings` structure or any of the generated inner structures to create customized configuration. Include *cycfg_bt_settings.h* in your application.
- *cycfg_sdp_db.h* and *cycfg_sdp_db.c* – Contain the SDP database code, include *cycfg_sdp_db.h* in your application.

4.4 Option4

- *design.cybt* – Contains an XML representation of the configuration. You may specify any file name.
- *cycfg_ble.h* and *cycfg_ble.c* – Contain the generated C code. Include the header file in your application. Use the generated structures as input parameters for the `Cy_BLE_Init()` function. Refer to the "Configuration Considerations" section in the Bluetooth® Low Energy Middleware API reference guide for more details about how to initialize and enable the Bluetooth® Middleware.

5 GUI description

The Bluetooth® Configurator GUI contains menus and tabs to configure Bluetooth® settings, as well as Notice List and Status Bar to provide indications. The tabs are described under Parameters configuration.

Menus

File

- **New** – Creates a new configuration file.
- **Open…** * – Opens a specified <file_name>.cybt configuration file. The current file, if any, will be closed.
- **Close** – Closes the configuration file. If there are unsaved changes, a dialog opens asking to save or not.
- **Save** – Saves changes to the file. If the file does not exist, the Save file dialog opens. When saving, it also generates code files.
- **Save As…**– Saves changes to a new file location.
- **Open in System Explorer** – Opens your computer's file explorer tool to the folder that contains the *.*cybt* file.
- **Import…** – Imports a specified configuration file.
- **Export…** – Exports the current configuration file into a specified file.
- **Recent files** ** – Shows recent files that you can open directly.
- **Exit** – Closes the configurator.

***Note**: * – The menu item is locked when the application is launched from the Eclipse IDE for ModusToolbox™.*

***Note**: ** – The menu item is not available when the application is launched from the Eclipse IDE for ModusToolbox™.*

Edit

- **Undo** – Undoes the last action or sequence of actions.
- **Redo** – Redoes the last undone action or sequence of undone actions.

View

- **Notice List** – Hides or shows the Notice List pane. The pane is shown by default.
- **Toolbar** – Hides or shows the Toolbar.
- **Reset View** – Resets the view to the default.

Help

- **View Help** – Opens this document.
- **About Bluetooth® Configurator** – Opens the About box for version information, with links to open https://www.infineon.com and the current session log file.

5.1 Main toolbar

The toolbar contains common commands from the **File** and **Edit** menus, such as **New**, Open, Save, etc. Use the check box under the View menu to show or hide the toolbar.



5.2 Tabs and tab components

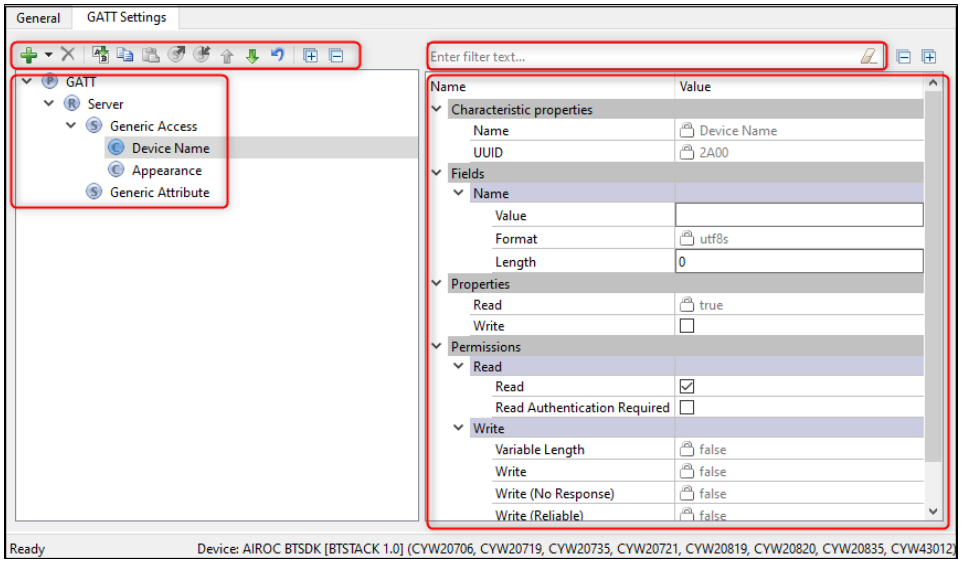
Tabs

- General tab (Option1, Option3)
- General tab (Option4)
- General LE tab (Option2, Option3)

- GATT Settings tab (Option1..4)
- GAP Settings tab (Option2, Option3, Option4)
- L2CAP Settings tab (Option2, Option3, Option4)
- Link Layer Settings tab (Option4)
- Service Discovery Settings tab (Option1, Option3) (Beta)

Tab components

For the different devices, the GUI contains one or more tabs to update various settings. The following shows a typical tab and identifies the key components.



The left side of the pane displays:

- the tab toolbar used for commands for a given tab
- the configuration tree to add nodes and define parameters on a given node

The right side of the pane displays:

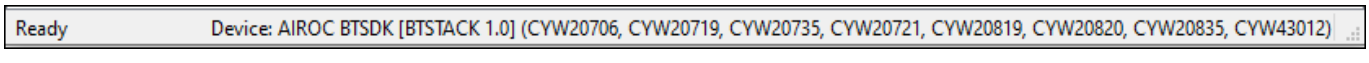
- the toolbar to filter, expand, and collapse the parameters
- the node configuration settings

5.3 Notice List

The **Notice List** pane combines notices (errors, warnings, tasks, and notes) from many places in the configuration into a centralized list. If a notice shows a location, you can double-click that entry to display the parameter causing the error or warning.

5.4 Status bar

The **Status bar** displays various information, including a file operation status, hints, input field ranges, etc. On the right side, it displays the selected device.



6 Parameters configuration

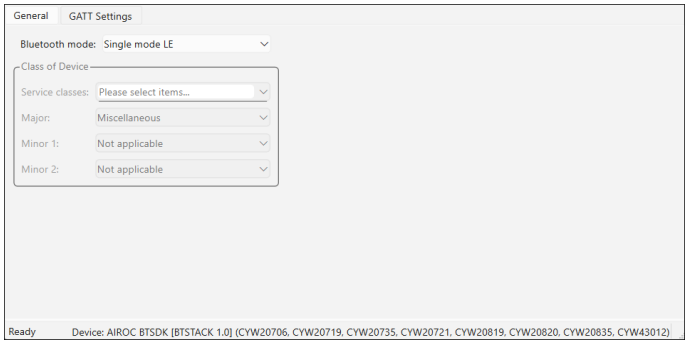
The Bluetooth® Configurator contains several tabs in which to configure parameters. The set of active tabs depends on the selected device and other parameter values.

- General tab (Option1, Option3)
- General tab (Option4)
- General LE tab (Option2, Option3)
- GATT Settings tab (Option1..4)
- GAP Settings tab (Option2, Option3, Option4)
- L2CAP Settings tab (Option2, Option3, Option4)
- Link Layer Settings tab (Option4)
- Service Discovery Settings tab (Option1, Option3) (Beta)

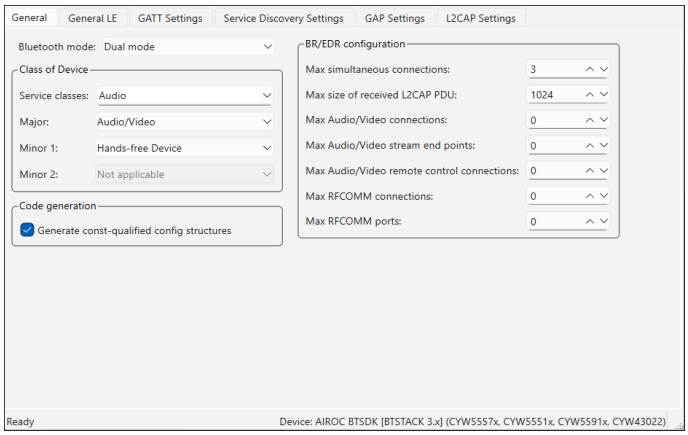
6.1 General tab (Option1, Option3)

The **General** tab allows general configuration of the Bluetooth® resource and displays depending on the mode selected:

for Option1



for Option3



The following chapters describe the **General** tab settings for **Option1** and **Option3**.

6.1.1 Bluetooth® mode

The **Bluetooth® mode** parameter supports the following functionality in different modes:

- **Single mode LE** – Only the Low Energy.
- **Single mode BR/EDR** – Only the Basic Rate (BR)/Enhanced Data Rate (EDR) or "classic" Bluetooth®.
- **Dual mode** – Both the Low Energy and BR/EDR.

6.1.2 Class of device

The **Class of Device** group of parameters indicates the device type and is available in **Single mode BR/EDR** or **Dual mode**:

- **Service classes** – Defines the general categories of devices your application will be associated with.
- **Major** – Defines the main device's function.
- **Minor 1** – Extends the Major device class value.
- **Minor 2** – Extends the Major device class value.

6.1.3 Isochronous Connection configuration

The **Isochronous Connection** configuration group is located in the **General LE** tab and is available in **Single mode LE**, and **Dual mode**. Isochronous Channels support is one of the main new features added in Bluetooth® specification 5.2. In conjunction with new GATT Audio profiles, it supports Bluetooth® LE Audio transport. LE Audio exists in the two main forms:

- Multi-Stream Audio – Supports the connection between an audio source device and one or more audio sink devices. It is bidirectional, with acknowledgments.
- Auracast™ Broadcast audio. – Supports the broadcasting of one or more audio streams from the source to an unlimited number of sink devices. It is unidirectional, without acknowledgments.

Isochronous Connection configuration parameters:

- **Max SDU size** – The maximum size of Service Data Unit (octets).
- **Max audio channels per packet** – The maximum number of audio channels (left, right …) contained in one packet.
- **Max number of CIS connections** – The maximum number of Connected Isochronous Stream connections. CIS is point-to-point, bidirectional connection between the central and peripheral device.
- **Max number of CIG connections** – The maximum number of Connected Isochronous Groups. CIG is the group of one or more Connected Isochronous Streams created by the central device.
- **Max number of buffers per CIS** – The maximum number of buffers per one Connected Isochronous Stream.
- **Max number of BIG connections** – The maximum number of Broadcast Isochronous Groups. BIG is the group of one or more Broadcast Isochronous Streams created by the Isochronous Broadcaster device.

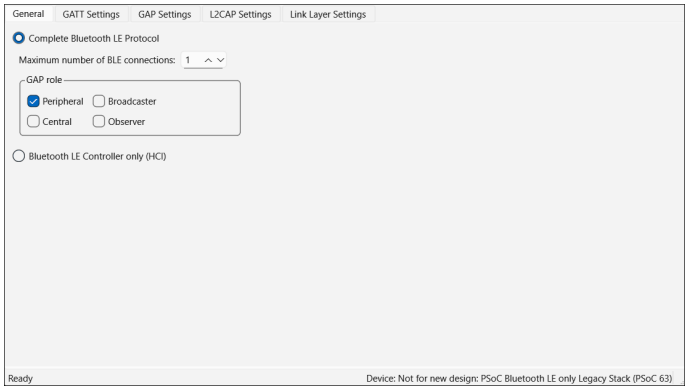
6.1.4 BR/EDR configuration

The **BR/EDR configuration** group of parameters is available only for **Single mode BR/EDR** and **Dual mode**:

- Max simultaneous connections
- Max size of received L2CAP PDU
- Max Audio/Video connections
- Max Audio/Video stream end points
- Max Audio/Video remote control connections
- Max RFCOMM connections
- Max RFCOMM ports

6.2 General tab (Option4)

The **General** tab for **Option4** allows general configuration of the Bluetooth® resource. The tab displays as follows, depending on the mode selected.



The following chapters describe the **General** tab settings for **Option4**.

6.2.1 Complete Bluetooth LE Protocol

The **Complete Bluetooth LE Protocol** mode enables both Bluetooth® Host and Controller. All GAP roles are exposed for configuration.

6.2.1.1 Maximum number of Bluetooth® LE connections

This parameter displays how many Bluetooth® LE connections (both Central and Peripheral) are allowed. The valid range is 1 - 4.

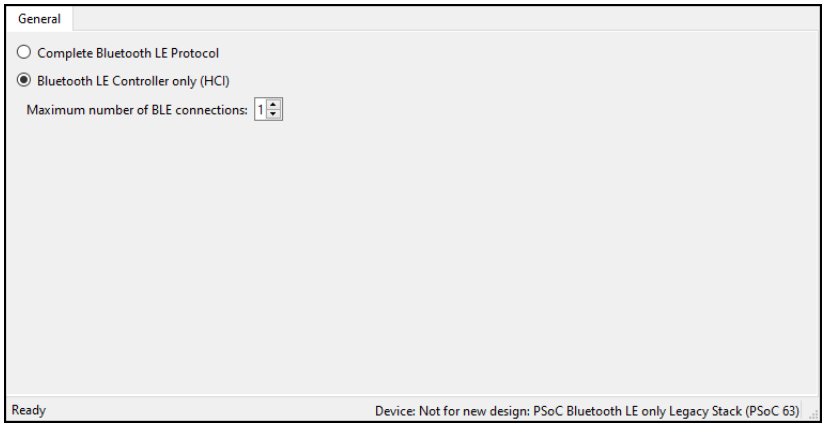
6.2.1.2 GAP role

The **GAP role** parameter can take the following values:

- **Peripheral** – Defines the device to advertise using connectable advertising packets in order to establish connection. Peripheral devices need a Central device, because the Central device initiates connection. A Peripheral device can broadcast general information about the device through the advertisement data.
- **Central** – Defines the device to initiate connection. Peripheral devices need a Central device, because the Central device initiates connection.
- **Broadcaster** – Defines the device to send advertising data – similar to the Peripheral role. However, the Broadcaster does not support connection, it sends data but does not receive it.
- **Observer** – Defines the device to scan for a Broadcaster and reports the received information to an application. The Observer role does not allow transmission.

6.2.2 Bluetooth LE Controller only (HCI)

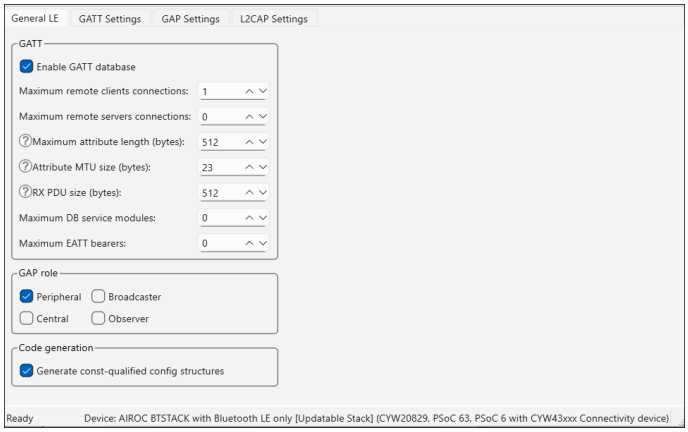
HCI mode enables the use of the device as **Bluetooth LE Controller**. Selecting this mode makes all other tabs unavailable.



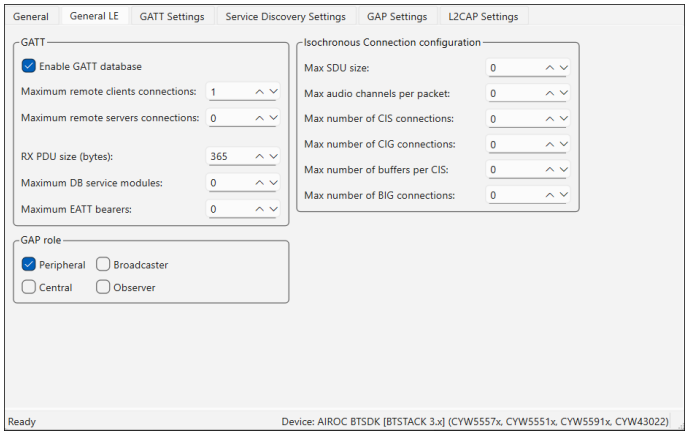
6.3 General LE tab (Option2, Option3)

The **General LE (Option2, Option3)** tab allows general configuration of the Bluetooth® resource:

for Option2



for Option3



The following chapters describe the **General LE tab** tab settings for **Option2** and **Option3**.

6.3.1 Enable GATT database

This parameter enables the GATT database. When not enabled, the **GATT Settings** tab hides and the files *cycfg_gatt_db.h* and *cycfg_gatt_db.c* are not generated.

6.3.2 Maximum remote clients/servers connections

These parameters display the maximum number of clients/servers that a local server can connect to.

6.3.3 Maximum attribute length

This value is the maximum length of an attribute in the design. The valid range is 23 - 512 bytes.

***Note**: This parameter is used in BTSTACK 3.0 or earlier.*

6.3.4 Attribute MTU size

This value is the Maximum Transmission Unit size (bytes) of an attribute in the design. The valid range is 23 - 517 bytes. This value is used to respond to an Exchange MTU request from the GATT Client.

***Note**: This parameter is used in BTSTACK 3.0 or earlier.*

6.3.5 RX PDU size

This value is the maximum size allowed for any received L2CAP PDU. The valid range is 65 - 65535 bytes.

***Note**: This parameter is used in BTSTACK 3.0 or earlier.*

6.3.6 GAP role

See GAP role.

6.3.7 Maximum DB service modules

The maximum number of GATT database segments to be added to the GATT DB. Set to 0 for most applications that do not update the GATT DB after invocation. Otherwise, a typical value is 1. The valid range is 0 - 255.

6.3.8 Maximum EATT bearers

The maximum number of allowed EATT bearers. Set to 0 for applications that support only legacy bearers. Otherwise, a typical value is 5. The valid range is 0 - 255.

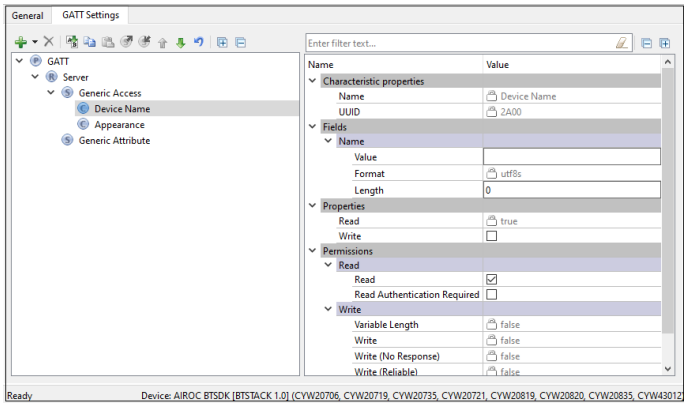
6.4 GATT Settings tab (Option1..4)

The **GATT Settings (Option1..4)** tab is used to configure profile-specific parameters. The tab has three areas:

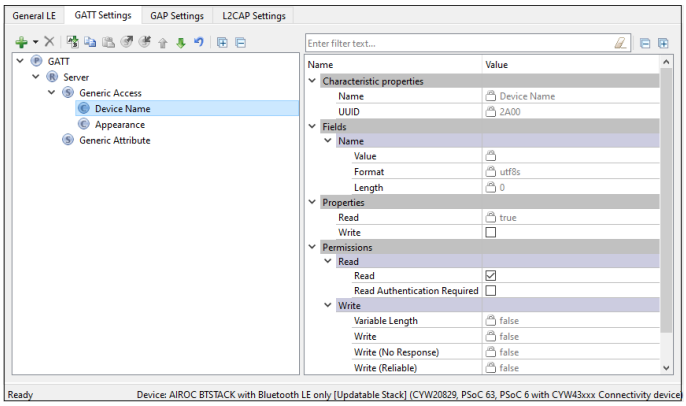
- toolbars
- profies tree
- parameters configuration pane

Depending on the selected device, some parameters may be unavailable. The tab displays for the different options as follows:

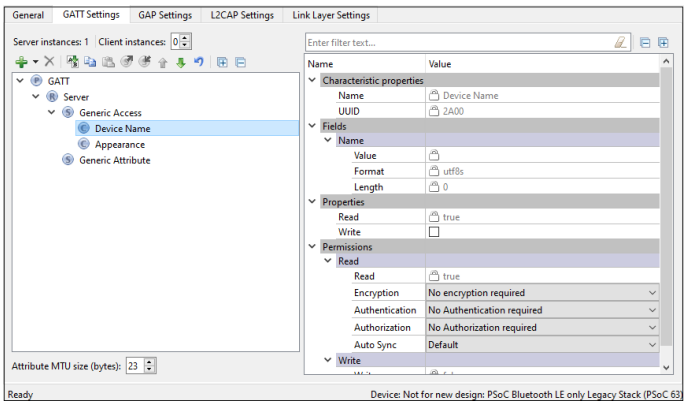
for Option1



for Option2, Option3



for Option4



6.4.1 Toolbars (GATT Settings)

The toolbars contain navigation options and various commands to add or delete Services, Characteristics, and Descriptors.

- **Server instances** (**Option4**) – The number of GATT Server instances. The Bluetooth® resource supports a single instance of a GATT Server (single GATT database). You can add additional Services or complete Profiles to the existing Server Profiles tree to build the GATT database. This single GATT database will be re-used across all Bluetooth® Low Energy connections.

***Note**: The CCCD values for each of active connections will be unique.*

- **Client instances** (**Option4**) – The number of GATT Client instances. One GATT Client instance exists per connection. You can configure up to four GATT Client instances. All GATT Client instances have one common Client Profiles tree configuration.
- **Add Profile** – Available when the GATT node is highlighted in the Profile tree. It allows adding a whole Profile to the Profiles tree. This option does not remove existing Services from the tree. Several Profiles can exist in the tree simultaneously.
- **Add Service** – Available when the Profile Role is highlighted in the Profile tree. Allows loading of Services in the selected Profile Role. In the GATT server configuration, adds the selected service data to the server GATT database and enables service-specific APIs.

  For **Option4**, in the GATT client configuration, the data structures for auto discovery of this service are created. If services not populated in the GUI are discovered during auto discovery, this option ignores those services and the application is responsible for discovering the details of such services. Refer to the Profiles section for the available Services.
- **Add Characteristic** – Available when a service is highlighted in the Profile tree. The Characteristic options are unique to each Service and are loaded automatically when a service is added to the design. The Add Characteristic button can be used to manually add a new characteristic to the service. All characteristics for the above-mentioned services plus Custom Characteristic are available for selection.
- **Add Descriptor** – Available when a characteristic is highlighted in the Profile tree. Similar to the Characteristic options, the Descriptor options are unique to a characteristic and are all automatically loaded when a characteristic is added to the design.
- **Delete** – Deletes the selected Service, Characteristic, or Descriptor.
- **Rename** – Renames the selected item in the Profiles tree.
- **Copy/Paste** – Copies/pastes items in the Profiles tree.
- **Export to file** – Exports the settings of the selected GATT Service and saves to an XML file .
- **Import from file** – Imports the saved file back to the GATT tree.
- **Move Up/Down** – Moves the selected item up or down in the Profiles tree.
- **Reset branch to default** – Resets the selected item with child items in the Profiles tree to the default.
- **Expand All** – Expands all items in the Profiles tree.
- **Collapse all Services** – Collapses all Services in the Profiles tree.

6.4.2 Profiles tree

The **Profiles** tree is used to view GATT Services, Characteristics, and Descriptors of the GATT Server and Client roles (GATT Client role is supported for **Option4**. By navigating through the tree, you can quickly add, delete, or modify Services, Characteristics, and Descriptors using the toolbar buttons or the context menu. You can configure the parameters by clicking an item on the tree. These parameters will show in the Parameters configuration section.

For GATT Services defined by Bluetooth® SIG (except for Generic Attribute and Generic Access), the version number displays on the tree next to the Service name. This corresponds to the Service specification version currently implemented in the Bluetooth® Configurator.

The tree may contain the following nodes:

- Profile
- Profile Role
- Service
- Characteristic
- Descriptor

In addition, nodes may include colors and icons as follows:

- Shaded icon – The node is mandatory
- White icon – The node is optional
- A '+' sign – The Service includes other Services
- Error icon – The node parameters have errors

6.4.3 Parameters configuration

The Parameters configuration pane allows you to configure a Profile, Service, or Characteristic by selecting the type of a Service or Characteristic in the tree. The filter box above the list of parameters allows limiting the items shown in the pane.

Attribute MTU size

Applicable for **Option4**. The Maximum Transmission Unit size (bytes) of an attribute to be used in the design. The valid range is 23 – 512 bytes. This value is used to respond to an Exchange MTU request from the GATT Client.

6.4.4 Profiles

You can add a whole profile to the Profiles tree from a list of supported profiles.

***Note**: All Profiles must have a Generic Access Service and a Generic Attribute Service.*

***Note**: For **Option1**, **Option2**, and **Option3** devices, only the GATT Server role is configurable.*

***Note**: The Service Characteristics are configurable only if they belong to a GATT Server node.*

***Note**: The security settings located in the **GAP Settings** tab are applied globally. Also, you may manually configure the security of each Characteristic/Descriptor.*

The following profiles are available for selection:

6.4.4.1 Alert Notification

This profile enables a GATT Client device to receive different types of alerts and event information, as well as information on the count of new alerts and unread items, which exist in the GATT Server device.

- Alert Notification Server Profile role – Specified as a GATT Server. Requires the following Service: Alert Notification Service.
- Alert Notification Client Profile role – Specified as a GATT Client.

Refer to the Alert Notification Profile Specification for detailed information about the Alert Notification Profile.

6.4.4.2 Automation IO

This profile enables a device to connect and interact with an Automation IO Module (IOM) in order to access digital and analog signals.

- Automation IO Server Profile role – Specified as a GATT Server. Requires the following Service: Automation IO Service.
- Automation IO Client Profile role – Specified as a GATT Client.

Refer to the Automation IO Profile Specification for detailed information about the Automation IO Profile.

6.4.4.3 Blood Pressure

This profile enables a device to connect and interact with a Blood Pressure Sensor device for use in consumer and professional health care applications.

- Blood Pressure Sensor Profile role – Specified as a GATT Server. Requires the following Services: Blood Pressure Service, Device Information Service.
- Blood Pressure Collector Profile role – Specified as a GATT Client. Requires support of the following Services: Blood Pressure Service. Support of Device Information Service is optional.

Refer to Blood Pressure Profile Specification for detailed information about the Blood Pressure Profile.

6.4.4.4 Continuous Glucose Monitoring

This profile enables a device to connect and interact with a Continuous Glucose Monitoring Sensor device for use in consumer healthcare applications.

- Continuous Glucose Monitoring Sensor Profile role – Specified as a GATT Server. Requires the following Services: Continuous Glucose Monitoring Service, Device Information Service. Optionally may include Bond Management Service.
- Collector Profile role – Specified as a GATT Client. Requires support of the following Services: Continuous Glucose Monitoring Service. Support of Bond Management Service and Device Information Service is optional.

Refer to Continuous Glucose Monitoring Profile Specification for detailed information about the Continuous Glucose Monitoring Profile.

6.4.4.5 Cycling Power

This profile enables a Collector device to connect and interact with a Cycling Power Sensor for use in sports and fitness applications.

- Cycling Power Sensor Profile role – Specified as a GATT Server. Requires the following Service: Cycling Power Service. Optionally may include Device Information Service and Battery Service.
- Cycling Power Sensor and Broadcaster Profile role. Requires the following Service: Cycling Power Service.
- Collector Profile role – Specified as a GATT Client. Requires support of the following Service: Cycling Power Service. Support of Device Information Service and Battery Service is optional.

Refer to Cycling Power Profile Specification for detailed information about the Cycling Power Profile.

6.4.4.6 Cycling Speed and Cadence

This profile enables a Collector device to connect and interact with a Cycling Speed and Cadence Sensor for use in sports and fitness applications.

- Cycling Speed and Cadence Sensor Profile role – Specified as a GATT Server. Requires the following Service: Cycling Speed and Cadence Service. Optionally may include Device Information Service.
- Collector Profile role – Specified as a GATT Client. Requires support of the following Service: Cycling Speed and Cadence Service. Support of Device Information Service is optional.

Refer to Cycling Speed and Cadence Profile Specification for detailed information about the Cycling Speed and Cadence Profile.

6.4.4.7 Electronic Shelf Label

This profile specifies how a Generic Attribute Profile (GATT) Client can control and update electronic shelf labels (ESLs) using Bluetooth® wireless technology.

- Electronic Shelf Label (ESL) role – Specified as a GATT Server. A device that can display a graphic (for example, containing a price) for an item offered for sale on a shelf. Such a graphic typically includes integrated sensors and light-emitting diodes (LEDs). Requires the support of the Electronic Shelf Label service. The support of Object Transfer and Device Information services is optional.
- Access Point (AP) role – Specified as a GATT Client. A central device that can exchange data with many ESLs in a system by using Bluetooth® wireless technology. Requires the support of the Electronic Shelf Label and Object Transfer services.

For more details, refer to the Electronic Shelf Label Profile specification.

6.4.4.8 Environmental Sensing

This profile enables a Collector device to connect and interact with an Environmental Sensor for use in outdoor activity applications.

- Environmental Sensor Profile role – Specified as a GATT Server. Requires the following Service: Environmental Sensing Service. Optionally may include Device Information Service and Battery Service.
- Collector Profile role – Specified as a GATT Client. Requires support of the following Service: Environmental Sensing Service. Support of Device Information Service and Battery Service is optional.

Refer to Environmental Sensing Profile Specification for detailed information about the Environmental Sensing Profile.

6.4.4.9 Find Me

This profile defines the behavior when a button is pressed on one device to cause an alerting signal on a peer device.

- Find Me Target Profile role – Specified as a GATT Server. Requires the following Service: Immediate Alert Service.
- Find Me Locator Profile role – Specified as a GATT Client. Requires support of the following Service: Immediate Alert Service.

Refer to Find Me Profile Specification for detailed information about the Find Me Profile.

6.4.4.10 Glucose

This profile enables a device to connect and interact with a Glucose Sensor for use in consumer healthcare applications.

- Glucose Sensor Profile role – Specified as a GATT Server. Requires the following Services: Glucose Service, Device Information Service.
- Collector Profile role – Specified as a GATT Client. Requires support of the following Service: Glucose Service. Support of Device Information Service is optional.

Refer to Glucose Profile Specification for detailed information about the Glucose Profile.

6.4.4.11 Health Thermometer

This profile enables a Collector device to connect and interact with a Thermometer sensor in healthcare applications.

- Thermometer Profile role – Specified as a GATT Server. Requires the following Services: Health Thermometer Service, Device Information Service.
- Collector Profile role – Specified as a GATT Client. Requires support of the following Service: Health Thermometer Service. Support of Device Information Service is optional.

Refer to Health Thermometer Profile Specification for detailed information about the Health Thermometer Profile.

6.4.4.12 HTTP Proxy

This service allows a Client device, typically a sensor, to communicate with a Web Server through a gateway device. HTTP Proxy Service is not available in the Add Profile drop-down list. It can be added as a separate Service.

Refer to HTTP Proxy Service Specification for detailed information about the HTTP Proxy Service.

6.4.4.13 Heart Rate

This profile enables a Collector device to connect and interact with a Heart Rate Sensor for use in fitness applications.

- Heart Rate Sensor Profile role – Specified as a GATT Server. Requires the following Services: Heart Rate Service, Device Information Service.
- Collector Profile role – Specified as a GATT Client. Requires support of the Heart Rate Service. Support of Device Information Service is optional.

Refer to Heart Rate Profile Specification for detailed information about the Heart Rate Profile.

6.4.4.14 HID over GATT

This profile defines how a device with Bluetooth® LE wireless communications can support HID Services over the Bluetooth® LE protocol stack using the Generic Attribute Profile.

- HID Device Profile role – Specified as a GATT Server. Requires the following Services: HID Service, Battery Service, and Device Information Service. Optionally may include Scan Parameters Service as part of the Scan Server role of the Scan Parameters Profile. HID Device supports multiple instances of HID Service and Battery Service and may include any other optional Services.
- Boot Host Profile role – Specified as a GATT Client. Requires support of the following Service: HID Service. Support of Battery Service and Device Information Service is optional.
- Report Host Profile role – Specified as a GATT Client. Requires support of the following Services: HID Service, Battery Service, Device Information Service. Support of Scan Client role of the Scan Parameters is optional.
- Report and Boot Host Profile role – Specified as a GATT Client. Requires support of the following Services: HID Service, Battery Service, Device Information Service. Support of Scan Client role of the Scan Parameters is optional.

Refer to HID over GATT Profile Specification for detailed information about the HID over GATT Profile.

6.4.4.15 Indoor Positioning

This service exposes location information to support mobile devices to position themselves in an environment where GNSS signals are not available. For example, on indoor premises. The location information is mainly exposed via advertising and the GATT-based service is primarily intended for configuration.

The Indoor Positioning Service is not available in the Profile drop-down list. It can be added as a separate Service.

Refer to Indoor Positioning Service Specification for detailed information about the Indoor Positioning Service.

6.4.4.16 Internet Protocol Support

This profile provides the support of exchanging Ipv6 packets between devices over the Bluetooth® Low Energy transport. The IPSP defines two roles – Node role and Router role. A device may support both Node role and Router role. A device supporting the Node role can be a sensor or actuator. A device supporting the Router role can be an Access Point (such as home router, mobile phone, or similar).

- Node Profile role – Specified as a GATT Server. Requires the following Service: Internet Protocol Support Service.
- Router Profile role – Specified as a GATT Client. Requires support of the following Services: Internet Protocol Support Service.

Refer to Internet Protocol Support Profile Specification for detailed information about IPSP.

6.4.4.17 Location and Navigation

This profile enables devices to communicate with a Location and Navigation Sensor for use in outdoor activity applications.

- Location and Navigation Sensor Profile role – Specified as a GATT Server. Requires the following Service: Location and Navigation Service. Optionally may include Device Information Service and Battery Service.
- Collector Profile role – Specified as a GATT Client. Requires support of the following Services: Location and Navigation Service. Support of Device Information Service and Battery Service is optional.

Refer to Location and Navigation Profile Specification for detailed information about the Location and Navigation Profile.

6.4.4.18 Object Transfer

This profile defines the fundamental requirements to enable an Object Client in order to create and delete objects and to execute a variety of actions using the currently-selected object such as reading the object data from or writing the object data to an Object Server that exposes the Object Transfer Service. This profile is designed to be referenced by a higher-layer specification to enable a variety of object transfer use cases.

- Object Server role – Specified as a GATT Server. The Object Server device interacts with an Object Client to read or write the object data and expose the object metadata. Requires the Object Transfer Service.
- Object Client role – Specified as a GATT Client. The Object Client device initiates the reading, writing or another object action from the Object Server. Requires the Object Transfer Service.

Refer to Object Transfer Profile Specification for detailed information about the Object Transfer Profile.

6.4.4.19 Phone Alert Status

This profile enables a device to alert its user about the alert status of a phone connected to the device.

- Phone Alert Server Profile role – Specified as a GATT Server. Requires the following Services: Phone Alert Status Service.
- Phone Alert Client Profile role – Specified as a GATT Client. Requires support of the following Service: Phone Alert Service.

Refer to Phone Alert Status Profile Specification for detailed information about the Phone Alert Status Profile.

6.4.4.20 Proximity

This profile enables the proximity monitoring between two devices.

- Proximity Reporter Profile role – Specified as a GATT Server. Requires the following Service: Link Loss Service. Optionally may include Immediate Alert Service and TX Power Service if both are used. Using only one of the optional Services is not allowed.
- Proximity Monitor Profile role – Specified as a GATT Client. Requires support of the following Services: Link Loss Service. Support of Immediate Alert Service and TX Power Service is optional. Same restrictions apply as to Proximity Reporter.

Refer to Proximity Profile Specification for detailed information about the Proximity Profile.

6.4.4.21 Pulse Oximeter

This profile enables a device to connect and interact with a Pulse Oximeter device for use in consumer and professional health care applications.

- Pulse Oximeter Sensor Profile role – Specified as a GATT Server. Requires the following Services: Pulse Oximeter Service, Device Information Service. Optionally may include Bond Management Service, Current Time Service and Battery Service.
- Collector Profile role – Specified as a GATT Client. Requires support of the following Services: Pulse Oximeter and Device Information Service. Support of Bond Management Service, Current Time Service and Battery Service are optional.

Refer to Pulse Oximeter Profile Specification for detailed information about the Pulse Oximeter Profile.

6.4.4.22 Ranging

The Ranging Profile (RAP) specification can be used by a distance-measurement application on a device to read remote ranging results of an ongoing Channel Sounding (CS) procedure and configure the desired content and delivery method from a remote device.

Ranging Data is the Channel Sounding Procedure results generated by a Core Controller and sent to the Host via the Host Controller Interface (HCI) Low Energy (LE) CS Subevent Result and Subevent Result Continue events. Ranging Data can be used by the client to perform a distance estimation.

Examples of distance-measurement applications include car and door locks that implement seamless unlocking and locking while a person using a hand-held mobile device approaches or departs the lock. The other uses include asset tracking in a warehouse and smart functionality based on proximity.

6.4.4.23 Running Speed and Cadence

This profile enables a Collector device to connect and interact with a Running Speed and Cadence Sensor for use in sports and fitness applications.

- Running Speed and Cadence Sensor Profile role – Specified as a GATT Server. Requires the following Service: Running Speed and Cadence Service. Optionally may include Device Information Service.
- Collector Profile role – Specified as a GATT Client. Requires support of the following Services: Running Speed and Cadence Service. Support of Device Information Service is optional.

Refer to Running Speed and Cadence Profile Specification for detailed information about the Running Speed and Cadence Profile.

6.4.4.24 Scan parameters

This profile defines how a Scan Client device with Bluetooth® LE wireless communications can write its scanning behavior to a Scan Server, and how a Scan Server can request updates of the Scan Client scanning behavior.

- Scan Server Profile role – Specified as a GATT Server. Requires the following Service: Scan Parameters Service.
- Scan Client Profile role – Specified as a GATT Client. Required support of the following Service: Scan Parameters Service.

Refer to Scan Parameters Profile Specification for detailed information about the Scan Parameters Profile.

6.4.4.25 Time

This profile enables the device to get the date, time, time zone, and DST information and control the functions related to time.

- Time Server Profile role – Specified as a GATT Server. Requires the following Service: Current Time Service. Optionally may include Next DST Change Service and Reference Time Update Service.
- Time Client Profile role – Specified as a GATT Client. Requires support of the following Service: Current Time Service. Support of Next DST Change Service and Reference Time Update Service is optional.

Refer to Time Profile Specification for detailed information about the Time Profile.

6.4.4.26 Weight Scale

This profile is used to enable a data collection device to obtain data from a Weight Scale that exposes the Weight Scale Service.

- Weight Scale Profile role – Specified as a GATT Server.
- Requires the following Services: Weight Scale Service and Device Information Service.
- Optionally may include: User Data Service, Body Composition Service, Battery Service and Current Time Service.
- Collector Profile role – Specified as a GATT Client.
- Required support of the following Service: Weight Scale Service and Device Information Service.
- Support of User Data Service, Body Composition Service, Battery Service and Current Time Service is optional.

Refer to Weight Scale Profile Specification for detailed information about the Weight Scale Profile.

6.4.4.27 Wireless Power Transfer (A4WP)

Enables the communication between Power Receiver Unit and Power Transmitter Unit in Wireless Power Transfer systems.

- Power Receiver Unit Profile role – Specified as a GATT Server. Requires the following Service: Wireless Power Transfer.
- Power Transmitter Unit Profile role – Specified as a GATT Client. Requires support of the following Service: Wireless Power Transfer.

Wireless Power Transfer Profile is a custom service defined by the Alliance for Wireless Power (A4WP). Refer to the AirFuel Alliance web site for detailed information about the Wireless Power Transfer Profile.

6.4.4.28 Bootloader

(**Option4**) The Bluetooth® Configurator supports the Bootloader Profile and Bootloader Service, which allow a Bootloader to update the existing firmware on the Bluetooth® Low Energy device. The Bootloader Service uses the Bluetooth® Low Energy interface as a communication interface. It can be added to any of the profiles if the design requires updating the firmware Over-the-Air (OTA).

Refer to the Bootloader service configuration section for detailed information about the Bootloader Service.

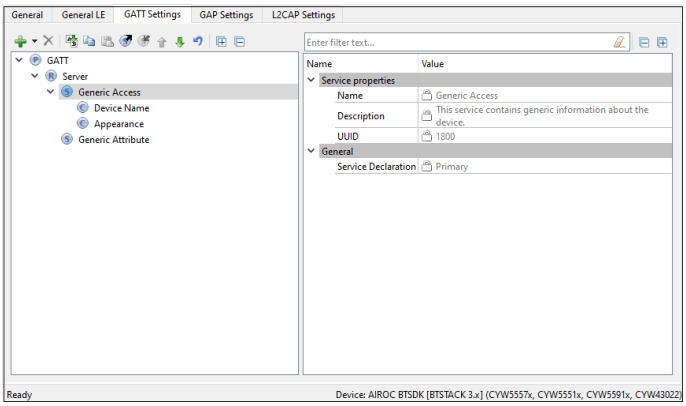
6.4.4.29 OTA Upgrade and OTA Secure Upgrade services

(**Option1** devices only) The Bluetooth® resource provides the firmware upgrade feature. It supports secure and non-secure versions of the upgrade. The OTA Upgrade Service supports the non-secure upgrade, and the OTA Secure Upgrade Service supports the secure upgrade. They can be added to any of the profiles if the design requires updating the firmware OTA.

Refer to the AIROC™ Firmware Upgrade Library documentation for detailed information about the OTA Upgrade and OTA Secure Upgrade Services.

6.4.5 Generic Access service

The **Generic Access Service** is used to define the basic Bluetooth® connection and discovery parameters. Click on the Characteristic under the Generic Access Service to view particular Characteristic settings.



When the **GAP Settings** tab is available, you perform the actual Characteristics configuration in the General options located in it.

- **Device Name** – This is the name of your device. It has a Read (without authentication/authorization) property associated with it by default. This parameter can be up to 248 bytes. For **Option2**, **Option3**, and **Option4**, the value comes from the **Device Name** field on the **GAP Settings** tab, under **General**.
- **Appearance** – The device's logo or appearance, which is a SIG defined 2-byte value. It has a Read (without authentication/authorization) property associated with it by default. For **Option2**, **Option3**, and **Option4**, the value comes from the **Appearance** field on the **GAP Settings** tab, under **General**.
- **Peripheral Preferred Connection** – A device in the peripheral role can convey its preferred connection parameter to the peer device. This parameter is 8 bytes in total and is composed of the following sub-parameters.

***Note**: This parameter is available only when the device supports a Peripheral role. Refer to the GAP Settings tab Connection parameters section for more information.*

***Note**: **Minimum Connection Interval** – A 2-byte parameter that denotes the minimum permissible connection time.*

Note: ***Maximum Connection Interval*** *– A 2-byte parameter that denotes the maximum permissible connection time.*

Note: ***Peripheral Latency*** *– A 2-byte value that defines the latency between consecutive connection events.*

Note: ***Connection Supervision Timeout*** *– A 2-byte value that denotes the LE link supervision timeout interval. Defines the timeout duration for which an LE link needs to be sustained if there is no response from the peer device over the LE link.*

Note: *For proper operation, the* ***Connection Supervision Timeout*** *must be larger than (1 + Peripheral latency) * Connection Interval * 2 (ms). Refer to Bluetooth® Core Specification Volume 6, Part B, Chapter 4.5.2 for more information on Connection Supervision Timeout.*

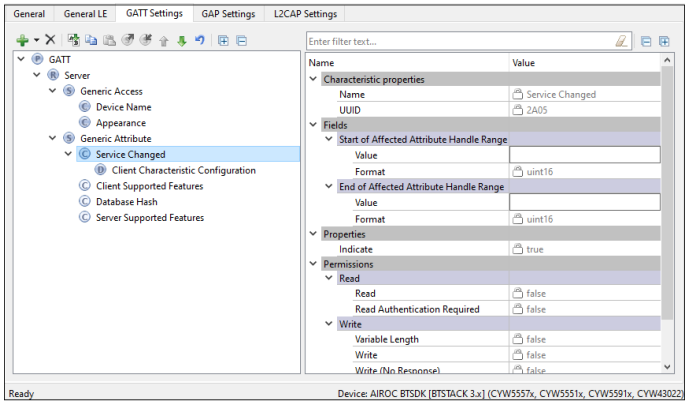
Note: *The above parameters are used for the connection parameters update procedure over L2CAP if a GAP central device does not use the peripheral preferred connection parameters. For example, iOS7 ignores peripheral preferred connection parameter Characteristics and establishes a connection with a default 30 ms connection interval. The peripheral device requests a connection parameter update by sending an L2CAP connection parameter update request at an appropriate time.*

Note: *A typical peripheral implementation initiates the L2CAP connection parameter update procedure once any Characteristic is configured for periodic notification or indication.*

- **Central Address Resolution** – A device in the central role can convey whether it supports privacy with address resolution. The Peripheral checks if the peer device supports address resolution by reading the **Central Address Resolution** characteristic before using directed advertisement where the initiator address is set to a Resolvable Private Address (RPA).
- **Resolvable Private Address Only** – Defines whether the device will use only Resolvable Private Addresses (RPAs) as local addresses.

6.4.6 Generic Attribute service

By default, the **Generic Attribute Service** does not have any Characteristics.

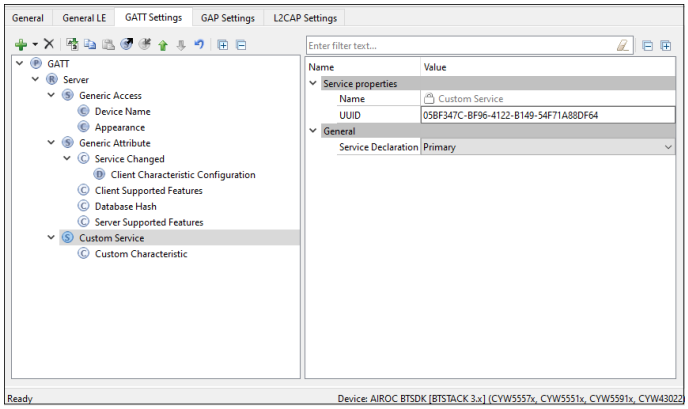


Add the optional **Service Changed** characteristic under the **Generic Attribute Service** if needed.

- **Service Changed** – This Characteristic is used to indicate to the connected devices that a Service has changed (added, removed, modified). It is used to indicate to GATT Clients that have a trusted relationship (bond) with the GATT Server when GATT-based Services have changed when they re-connect to the GATT

Server. It is mandatory for the device in the GATT Client role. For the device in the GATT Server role, the Characteristic is mandatory if the GATT Server changes the supported Services in the device.

6.4.7 Custom Service configuration



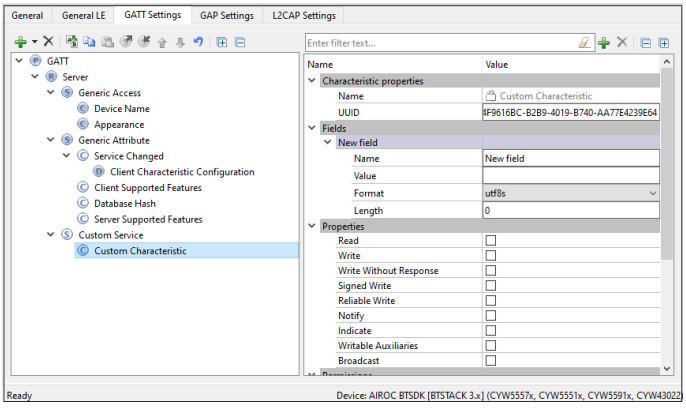
- **UUID**

A universally unique identifier of the service. This field is editable for Custom Services. By default, it is initialized with a random 128-bit UUID.

- **Service type**
  - **Primary** – Represents the primary functionality of the device.
  - **Secondary** – Represents additional functionality of the device. The secondary service must be included in another service.
- **Included sevirces**

This is the list of the services that can be included in the selected service. Each service may have one or more included services. The included services provide additional functionality for the service.

6.4.8 Custom Characteristic configuration



- **UUID**

  A universally unique identifier of the Characteristic. This field is editable for **Custom Characteristic**. By default, it is initialized with a random 128-bit UUID.

- **Fields**

  The fields represent a Characteristic value. The default value for each field can be set in the Value property. The fields are customizable for the Custom Characteristic. You can add or delete fields using the tool buttons located above the Properties editor.

- **Properties**

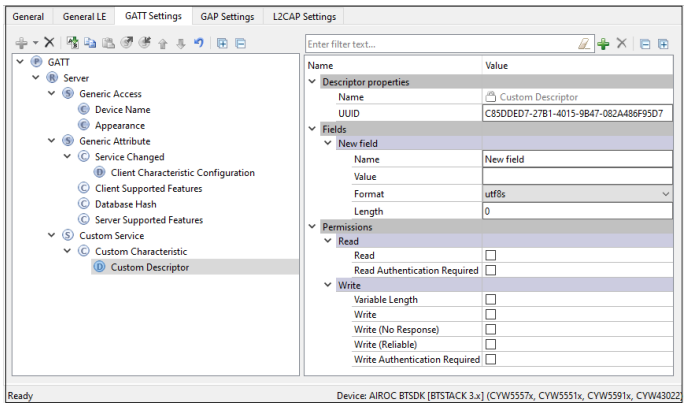
  The Characteristic properties define how the Characteristic value can be used. Some properties (Broadcast, Notify, Indicate, Reliable Write, Writable Auxiliaries) require the presence of a corresponding Characteristic Descriptor. For details, see Bluetooth ® Core Specification Vol.3, part G (GATT), Section 3.3.1.1 "Characteristic Properties".
- **Permissions**

  Characteristic permissions define how the **Characteristic Value** attribute can be accessed and the security level required for this access. Access permissions are set based on the Characteristic properties.

  Permissions parameters differ depending on the selected device.

  For **Option4**, the **Auto sync** property is available which determines if the **Security permissions** are automatically updated when the **Security Mode** or **Security Level** parameters are changed in **Security Configuration 0** on the **GAP Settings** tab. Additional Security configurations do not affect attribute permissions.

6.4.9 Custom Descriptor configuration



- **UUID**

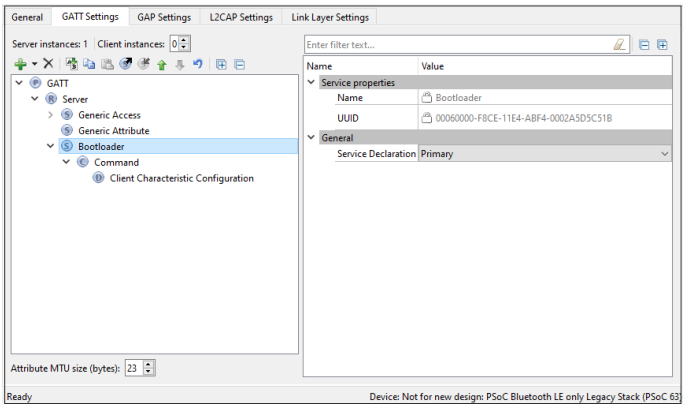
  A universally unique identifier of the **Descriptor**. This field is editable for **Custom Descriptors**. By default, it is initialized with a random 128-bit UUID.
- **Fields**

  Fields represent a descriptor value. The default value for each field can be set in the Value property. In case of the **Custom Descriptor**, the fields are customizable. You can add or delete fields using the tool buttons located above the **Properties** editor.
- **Permissions**

  Descriptor permissions define how the **Descriptor** attribute can be accessed and the security level required for this access. **Permissions** parameters differ depending on the selected device.

6.4.10 Bootloader service configuration

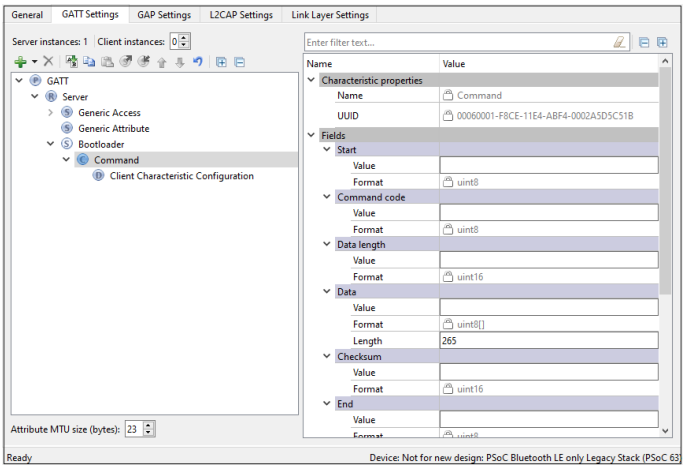
This service is available only for **Option4**.



- **UUID**

  A universally unique identifier of the service. The **UUID** is set to 00060000-F8CE-11E4-ABF4-0002A5D5C51B.

6.4.11 Command characteristic configuration



- **UUID**

  The universally unique identifier of the Characteristic. The **UUID** is set to 00060001-F8CE-11E4-ABF4-0002A5D5C51B.

- **Fields**
  - **Start of packet** – The start of a bootloader packet.
  - **Command** – A bootloader command.
  - **Status Code** – Command status code.
  - **Data Length** – The length of a bootloader command/response to set to the maximum command data length possible in the design.

  Per Bluetooth® Low Energy protocol, if the command requires a response larger than 20 bytes, the attribute MTU size should be increased. To support responses with a data length set to 56 (response for Get Metadata command), set the attribute MTU size to 66 according to the equation:

  MTU size = Data Length + Bootloader command overhead + notification parameters overhead where:

  - Data Length = the response data length
  - Bootloader command overhead = 7
  - Notification parameters overhead = 3

If the requirements are not followed, the Bluetooth® resource will fail to send a response to the requested command.

- **Data** – Bootloader command data. The length of this field is specified by the Data Length field.
- **Checksum** – A checksum computed for the entire packet except for the Checksum and End of Packet fields.
- **End of Packet** – The end of a bootloader packet.

6.4.12 GATT service import/export

When creating new or updating existing configuration, you can export the selected GATT Service settings to an XML file on disk. Later, when working with another configuration, the saved file (service) can be imported back to the GATT tree. The imported configuration is analyzed for any possible errors (e.g. duplicate service name, etc.)

Partial configuration, containing a single GATT service is saved into a separate XML file on disk. This file has the special extension **.btsvc*. To export a service, click its name on the GATT tree and press the **Export** button. To import a previously exported service, click “Server” on the GATT tree, press the **Import** button, and select the previously created **.btsvc* file.

6.5 GAP Settings tab (Option2, Option3, Option4)

The **GAP Settings** tab parameters define the general connection settings required when connecting Bluetooth® devices. Select an item in the tree to see the various sections of parameters.

In the **General** tab, select the **GAP role** to display the **GAP Settings** tab settings. Use the **Restore Defaults** button to restore the default settings of the active tree item.

***Note**: Values of many parameter settings can be customized. Also, many parameters have check boxes to enable or disable them.*

The following sections describe the different categories of parameters based on the item selected in the tree.

6.5.1 Toolbar (GAP Settings)

The toolbar contains options to add or delete **GAP role** configuration and **Security configuration**.

- **Add** (**Option4**)) – Allows adding: Peripheral, Central, Broadcaster, Observer or Security configurations.

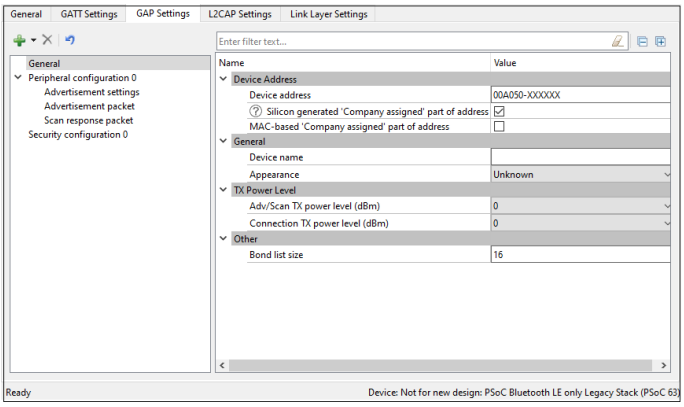
  ***Note**: The available options depend on the GAP role selected in the **General** tab. You can add several configurations for one GAP role and switch between them using the firmware.*

- **Delete** (**Option4**)) – Deletes the selected Configuration.

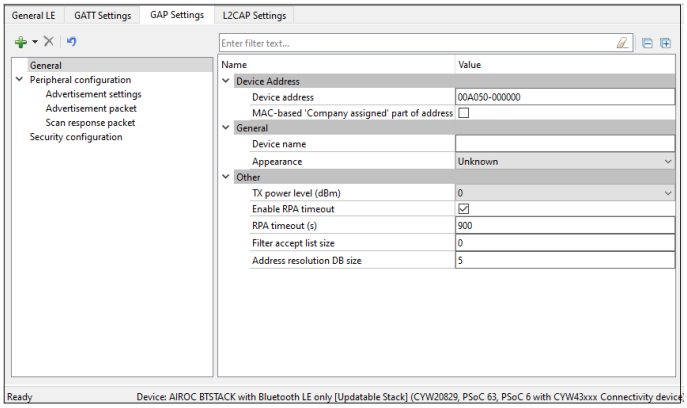
6.5.2 GAP Settings tab General

This General section contains general GAP parameters. **Option2**, **Option3**, and **Option4** have different sets of parameters.

Option4



Option2, Option3



6.5.2.1 Device Address

This is a unique 48-bit Bluetooth® public address used to identify the device. It is divided into two parts:

- **Company ID** – The 24 most significant bits: a 24-bit Organization Unique Identifier (OUI) address assigned by IEEE.
- **Company assigned** – The 24 least-significant bits.

The address configured here is static and designed for development purposes only.

***Note**: For **Option4**, during production, the device address is programmed into the user area of the supervisory flash (Sflash) location for the device address (Row 0) via the SWD interface. Normally, this address must be programmed only once during mass production, and then never changed in-field. However, application flash can be reprogrammed in-field many times.*

During prototyping (FW design), the device address can be programmed into the user area of the Sflash location using the CYPRESS™ Programmer OpenOCD CLI. Refer to https://www.infineon.com/dgdlac/?fileId=8ac78c8c7ddc01d7017ddcfc7a355604 for more details.

The following command is an example of using the CLI to update the device address structure (type of `cy_stc_ble_gap_bd_addr_t`) with the data: 11 00 00 50 A0 00 00 (first 6 bytes are address with LSB first and last byte is the address type) in the Row 0 (0x1600 0800) of user area of the Sflash.

```
openocd.exe -s ../scripts -f interface/kitprog3.cfg -f target/psoc6.cfg -c " init; reset init;
psoc6 allow_unsafe_sflash on; flash rmw 0x16000800 11000050A00000; reset; exit"
```

```
Warn : Writes to 'unsafe' Supervisory Flash rows is now ALLOWED
[100%] [##################################] [ Programming ]
[100%] [##################################] [ Programming ]
modified 7 byte(s) in 512 byte region at 0x16000800 in 0.088930s (5.622 KiB/s)
Warn : Only resetting the Cortex-M core, use a reset-init event handler to reset any peripherals or configure hard
ware srst support.
```

Row 1, Row 2, and Row 3 of the user area of the Sflash are not used by Bluetooth® LE and available for the user information storage.

6.5.2.2 Silicon-generated 'Company assigned' part of address (Option4)

When checked, the "'Company assigned' part of address" is generated using the factory programmed die X/Y location, wafer ID and lot ID of the silicon.

***Note**: The **Silicon-generated 'Company assigned' part of address** option does not guarantee a unique device address. For mass production, Infineon Technologies strongly suggests that the device address be programmed into the user area (Row 0) of the Sflash location via the SWD interface.*

6.5.2.3 MAC-based 'Company assigned' part of address

When checked, the **'Company assigned' part of address**is generated based on the MAC address of the PC.

6.5.2.4 Device name

The device name displays on the peer side. It can be up to 248 bytes.

This parameter:

- has the default-associated Read (without authentication/authorization) property
- configures the GAP Service Device Name Characteristic located in the Profile Tree
- can be modified only when the device is a GATT Server.

6.5.2.5 Appearance

The device logo or appearance is a SIG-defined 2-byte value.

This parameter:

- has the default-associated Read (without authentication/authorization) property
- configures the GAP Service Appearance Characteristic located in the Profile Tree
- can ber modified only when the device is a GATT Server.

6.5.2.6 Adv/Scan TX power level (Option4)

The initial transmitter power level (dBm) of the advertisement or scan channels upon startup. Default: 0 dBm. Possible values: -20 dBm, -16 dBm, -12 dBm, -6 dBm, 0 dBm, 4 dBm.

6.5.2.7 Connection TX power level (Option4)

The initial transmitter power level (dBm) of the connection channels upon startup. Default: 0 dBm. Possible values: -20 dBm, -16 dBm, -12 dBm, -6 dBm, 0 dBm, 4 dBm.

6.5.2.8 TX power level (Option2, Option3)

The initial transmitter power level (dBm) upon startup. Default: 0 dBm. Possible values: 20dBm, -16 dBm, -12 dBm, -8 dBm, -6dBm, -4 dBm, 0 dBm, 4 dBm, 8 dBm, 12 dBm.

***Note**: The **TX power level (dBm)** parameter on the **GAP Settings tab General** screen is not applicable to **CYW20829** devices and has no effect. Leave the default value (0dBm) in this field when using a **CYW20829** device.*

***Note**: Check your project device datasheet to know which TX power levels the device supports.*

6.5.2.9 Bond list size (Option4)

The maximum number of bonded devices supported by this device. The valid range is 1 - 128. Default: 16.

***Note**: The maximum number of bonded devices is also limited by the size of the available application flash (emulated EEPROM area) size to be consumed for data storage. The consumed application flash size is calculated as the multiple of the number of supported services and the multiple of the number of supported bonded devices.*

6.5.2.10 RPA timeout (Option2, Option3)

The interval of random address refreshing (seconds). The valid range is 1 - 41400. Default: 900.

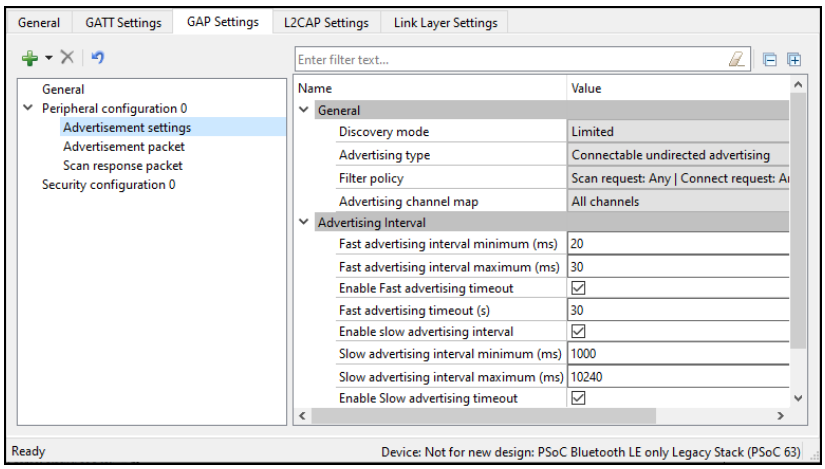
6.5.2.11 Filter-accept list size (Option2, Option3)

The maximum number of devices that can be added to the Filter accept list. The valid range is 0 - 128. Default: 0.

6.5.3 GAP Settings tab Advertisement settings

6.5.3.1 Option4

These parameters are available when the device is configured to contain a Peripheral or Broadcaster GAP role.



General

Discovery mode

- Non-discoverable – The device cannot be discovered by a Central device.
- Limited – Enables a device to be discoverable only for a limited period of time, during temporary conditions, or for a specific event. The device advertising in Limited Discoverable mode can connect to the Central device that is performing the Limited Discovery procedure. The timeout duration is defined by the applicable advertising timeout parameter.
- General – The device is used by devices to be discoverable continuously or for no specific condition. The device advertising in General Discoverable mode can connect to the Central device that is performing the General Discovery procedure.

Advertising type

This parameter defines the advertising type to be used by the LL for appropriate **Discovery mode**.

- Connectable undirected advertising – Advertises advertising and scan response data. It allows any other device to connect to this device.
- Scannable undirected advertising – Broadcasts advertising data and scan response data to active scanners.
- Non-connectable undirected advertising – Broadcasts advertising data.

Filter policy

This parameter defines the advertising type to be used by the LL for appropriate**Discovery mode**.

- Scan request – Any | Connect request: Any – Processes scan and connect requests from all devices.
- Scan request – Filter accept list | Connect request: Any – Processes scan requests only from the devices in the Filter accept list and connect requests from all devices.
- can request – Any | Connect request: Filter accept list – Processes scan requests from all devices and connect requests only from the devices in the Filter accept list.
- Scan request – Filter accept list | Connect request: Filter accept list – Processes scan and connect requests only from the devices in the Filter accept list.

Advertising channel map

This parameter is used to enable a specific advertisement channel.

- Channel 37 – Enables advertisement channel #37.
- Channel 38 – Enables advertisement channel #38.

- Channel 39 – Enables advertisement channel #39.
- Channels 37 and 38 – Enables advertisement channels #37 and #38.
- Channel 37 and 39 – Enables advertisement channels #37 and #39.
- Channels 38 and 39 – Enables advertisement channels #38 and #39.
- All channels – Enables all three advertisement channels.

Advertising interval

This parameter defines the interval between two advertising events. Set the permissible minimum and maximum values of two **Advertising interval** types: Fast advertising interval and Slow advertising interval. Typically, after the device initialization, a peripheral device uses the Fast advertising interval. After the Fast advertising interval timeout value expires, and if a connection with a Central device is not established, then the Profile switches to Slow advertising interval to save the battery life. After the Slow advertising interval timeout value expires, CY_BLE_EVT_GAPP_ADVERTISEMENT_START_STOP event is generated.

***Note**: The Advertising interval requires aligning with the selected Profile specification.*

Fast advertising interval

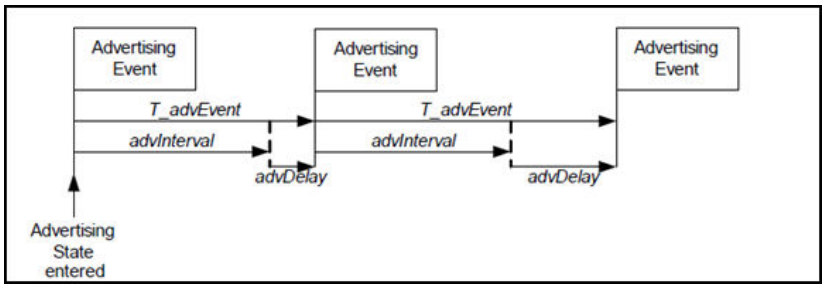
This advertisement interval results in faster LE Connection. The Bluetooth® resource uses this interval value when the connection time is between the specified minimum and maximum values of the interval.

- Minimum – The minimum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms. The valid range is from 20 ms to 10240 ms.
- Maximum – The maximum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms. The valid range is from 20 ms to 10240 ms.
- Timeout – The timeout value of advertising with fast advertising interval parameters. When equals 0, the device is advertising continuously and slow advertising settings become unavailable. A timeout cannot occur before the advertising interval expires. So, if a timeout value is smaller than the fast advertising interval minimum value, a warning displays.

Slow advertising interval

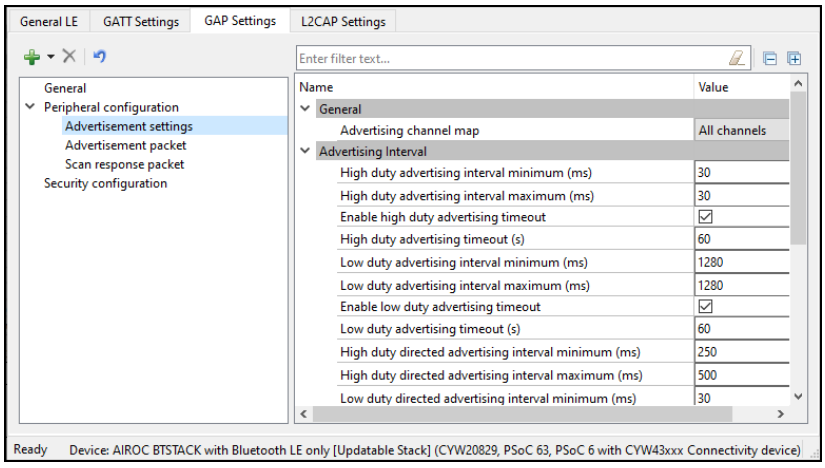
Defines the advertising interval for slow advertising. This is an optional parameter to implement advertising with a lower duty cycle to save battery life. The Slow advertising interval parameters are applied to the device after an internal fast advertising interval timeout occurs. The minimum and maximum values defined using this parameter allow the Bluetooth® LE Stack to expect advertising to happen within these intervals. This parameter is not applicable in General discovery mode.

- Minimum – The minimum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms. The valid range is from 20 ms to 10240 ms.
- Maximum – The maximum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms. The valid range is from 20 ms to 10240 ms.
- Timeout – The timeout value of advertising with fast advertising interval parameters. When equals 0, the device is advertising continuously and slow advertising settings become unavailable. A timeout cannot occur before the advertising interval expires. So, if a timeout value is smaller than the fast advertising interval minimum value, a warning displays.



- **AdvDelay** – A pseudo-random delay of 0-10 ms.
- **Complete Advertising Event** – Consists of one advertising PDU sent into each of the used advertising channels.

6.5.3.2 Option2, Option3



General

Advertising channel map

This parameter is used to enable a specific advertisement channel:

- Channel 37 – Enables advertisement channel #37.
- Channel 38 – Enables advertisement channel #38.
- Channel 39 – Enables advertisement channel #39.
- Channels 37 and 38 – Enables advertisement channel #37 and #38.
- Channels 38 and 39 – Enables advertisement channels #38 and #39.
- All channels – Enables all three advertisement channels.

Advertising interval

This parameter defines the interval between two advertising events.

High duty advertising interval

- Minimum – The minimum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms
- Maximum – The maximum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms.
- Duration – The duration of advertising with high-duty advertising interval parameters. When not enabled, the device is advertising continuously.

Low duty advertising interval

- Minimum – The minimum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms.
- Maximum – The maximum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms.
- Duration – The duration of advertising with high-duty advertising interval parameters. When not enabled, the device is advertising continuously.

High duty directed advertising interval

- Minimum – The minimum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms.
- Maximum – The maximum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms.

Low duty directed advertising interval

- Minimum – The minimum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms.
- Maximum – The maximum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms
- Duration – The duration of advertising with high-duty advertising interval parameters. When not enabled, the device is advertising continuously.

High duty non connectable advertising interval

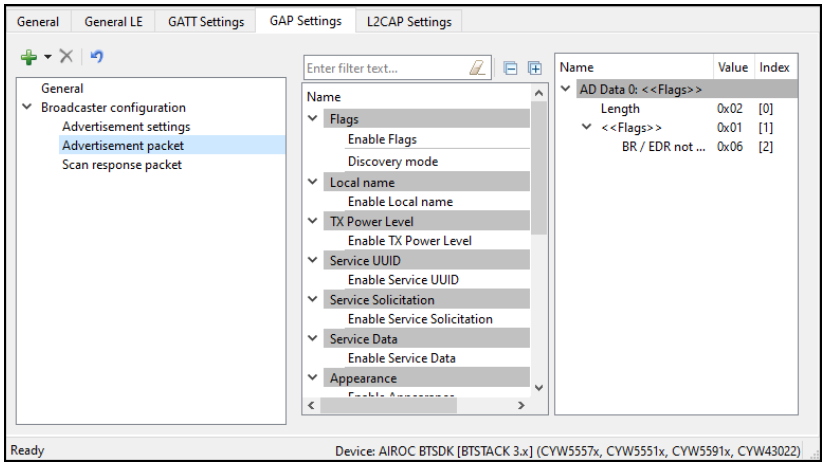
- Minimum – The minimum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms.
- Maximum – The maximum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms.
- Duration – The duration of advertising with high-duty advertising interval parameters. When not enabled, the device is advertising continuously.

Low duty non connectable advertising interval

- Minimum – The minimum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms.
- Maximum – The maximum interval for advertising data and establishing the LE Connection. The parameter is configured to increment in multiples of 0.625 ms.
- Duration – The duration of advertising with high-duty advertising interval parameters. When not enabled, the device is advertising continuously.

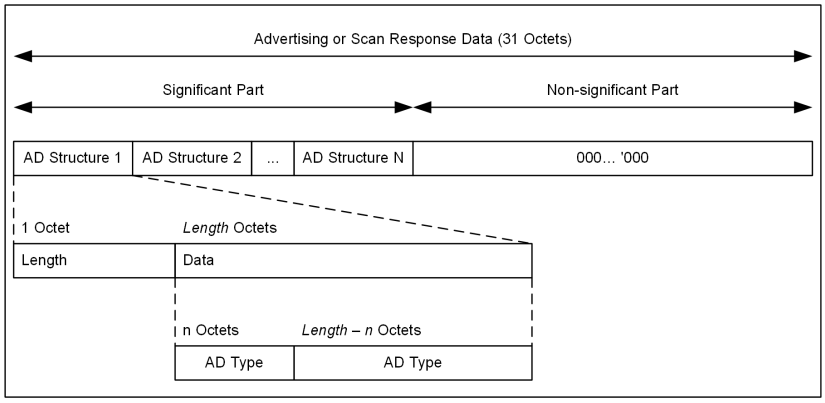
6.5.4 GAP Settings tab Advertisement packet

This Advertisement packet section displays when the device is configured to contain a Peripheral or Broadcaster GAP role. It is used to configure Advertisement data.



Advertisement/Scan response packet data settings

Advertisement (AD) or Scan response packet data is a 31-byte payload used to declare the device's Bluetooth® Low Energy capability and its connection parameters. This data structure is shown in the below graph per Bluetooth® specification.



A data packet can contain a number of AD structures. Each of these structures is composed of the following parameters:

- **AD Length** – The size of the AD Type and AD Data in bytes.
- **AD Type** – The type of an advertisement within the AD structure.
- **AD Data** – Data associated with the AD Type.

The total length of a complete Advertising packet cannot exceed 31 bytes.

An example structure for Advertisement data or Scan response data is AD Structure Element Definition:

- AD Length – Size of AD Type and associated AD Data = 5 bytes
- AD Type (1 byte) – 0x03 (Service UUID)
- AD Data (4 bytes) – 0x180D, 0x180A (Heart Rate Service, Device Information Service)

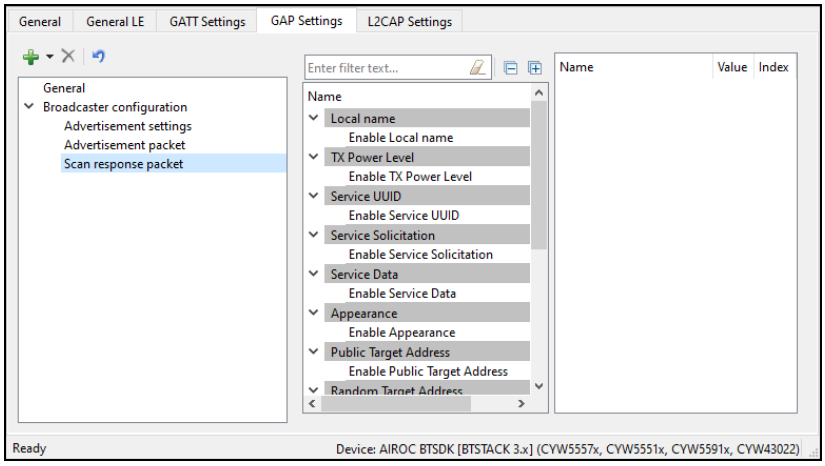
The following table shows the AD Types:

AD Type	Description
Flags	Flags to broadcast the underlying Bluetooth® LE transport capability such as Discoverable mode, LE only, etc.
Local Name	Device Name (complete of shortened). The device name value comes from the Device name field on the **GAP Settings** tab, under **General**.

AD Type	Description
TX Power Level	Transmit Power Level. Taken from the Adv/Scan TX power level (**Option4**) or TX power level (**Option2**, **Option3**) field on the **GAP Settings** tab, under **General**.
Peripheral Connection Interval Range	The preferred connection interval range for the device. Not available in the Broadcaster GAP role.
Service UUID	The list of device-implemented Service UUIDs to be broadcasted. There are different AD Type values to advertise 16-bit, 32-bit, and 128-bit Service UUIDs. 16-bit and 32-bit Service UUIDs are used if they are assigned by the Bluetooth® SIG.
Service Solicitation	The list of Service UUIDs from the central device to be used by the peripheral device. There are different AD Type values to advertise 16-bit, 32-bit, and 128-bit Service UUIDs.
Service Data	2/4/16-byte Service UUID, followed by additional Service data.
Security Manager TK value	A temporal key to use at the time of pairing. Not available in the Broadcaster GAP role.
Appearance	The device external appearance. The value comes from the Appearance field on the **GAP Settings** tab, under **General**.
Public Target Address	The public device address of intended recipients.
Random Target Address	The random device address of intended recipients.
Advertising Interval	For **Option4**, calculated as an average of Fast advertising interval minimum and maximum values configured on the GAP Settings tab, under Advertisement settings.
LE Bluetooth® Device Address	The device address of the local device. The value comes from the Public device address field on the **GAP Settings** tab, under **General**.
LE Role	Supported LE roles. Not available in the Broadcaster GAP role.
URI	URI, as defined in the IETF STD 66.
Manufacturer Specific Data	The 2-byte company identifier followed by manufacturer-specific data.
Indoor Positioning	Data specified in the Indoor Positioning Service Specification. Available when the Indoor Positioning Service is in the Profile.

6.5.5 GAP Settings tab Scan response packet

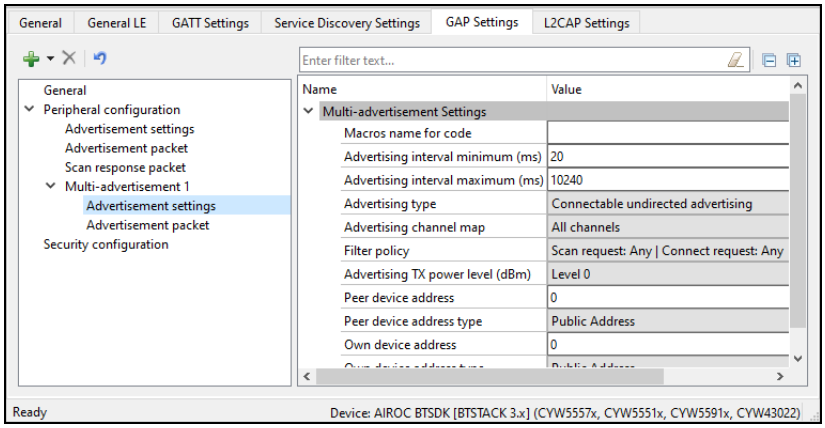
This **Scan response packet**t section displays when the device is configured to contain a Peripheral or Broadcaster GAP role. It is used to configure a **Scan response data** packet to be used in response to device scanning performed by a GATT Client device.



The **Scan response packet** structure is the same as of the **Advertisement packet**. See Advertisement / Scan response packet data settings under GAP Settings tab Advertisement packet for information on configuring the **Scan response packet**.

6.5.6 GAP Settings tab – Multi-advertisement

The **Multi-advertisement Settings** option is available when the device is configured to contain a Peripheral or Broadcaster GAP role. This feature allows you to advertise different sets of data at different time intervals. You can add up-to-ten multi-advertisement instances to the configuration tree and set different advertisement parameters and data.



The parameters are as follows:

- **Macros name for code** – Provides a meaningful name to the instance to create a macro-definition in the generated code.
- **Advertising interval minimum (ms)/Advertising interval maximum (ms)** – Configured to increment in multiples of 0.625 ms.
- **Advertising type** – Has several selectable options of advertising: Connectable undirected, Connectable directed, Scannable undirected, Non-connectable undirected, Connectable low duty cycle directed.
- **Advertising channel map** – Selection of primary advertising channels – 37, 38, 39
- **Filter policy** – Setting the Scan request and Connect request filtering options
- **Advertising TX power level (dBm)** – Selection of a transmission power level: from -12dBm (level 0) to 4dBm (level 4
- **Peer device address** – The six-byte address of a peer Bluetooth® LE device
- **Peer device address type** – The public, random, Address or ID address of a peer Bluetooth® LE device

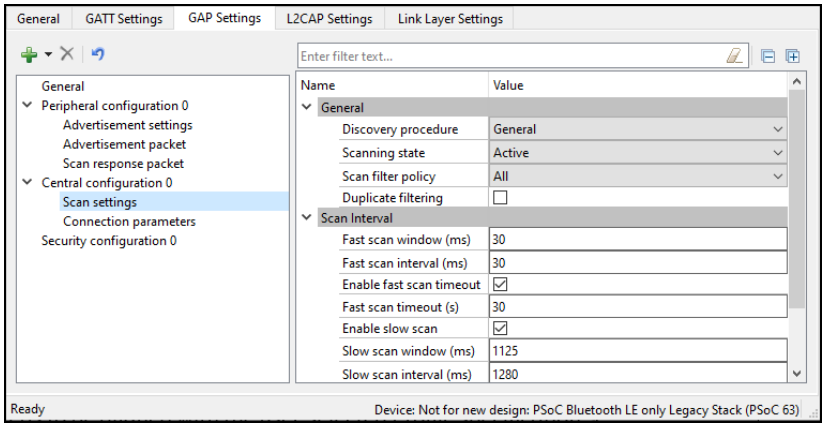
- **Own device address** – Device own six-byte address
- **Own device address type** – The own device public, random, Address or ID address type

For more details on the parameters, refer to GAP Settings tab General and GAP Settings tab Advertisement settings.

6.5.7 GAP Settings tab Scan settings

6.5.7.1 Option4

These parameters are available when the device is configured to contain the Central or Observer GAP role. Typically, during a device discovery, the GATT Client device initiates the scan procedure. It uses the Fast scan parameters for a period of time, approximately 30 to 60 seconds, and then it reduces the scan frequency using the Slow scan parameters.



***Note**: The scan interval requires aligning with the user-selected Profile specification.*

General

Discovery procedure

- Limited – The performing device discovers a device for limited Discovery mode advertising only.
- General – The performing device discovers a device for both general and limited Discovery advertising.

Scanning state

- Passive – The device can only listen to advertisement packets.
- Active – The device may ask the advertiser for additional information.

Scan filter policy

This parameter defines how the advertisement packets are filtered.

- **All** – All advertisement packets are processed
- **Filter accept list** only – Only advertisement packets from the devices in the Filter accept list are processed.

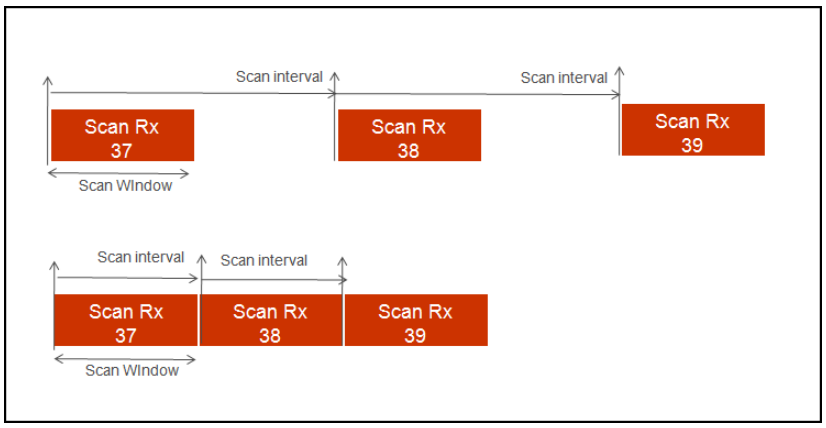
Duplicate filtering

When enabled, this activates filtering of duplicated advertisement data. If disabled, the Bluetooth® LE stack will not perform filtering of advertisement data.

***Note**: The controller firmware has eight address locations reserved to cache the previously seen advertiser devices and filter duplicate packets from them. If there are more than eight advertising devices in the scanner proximity during the scan period, the address storing buffer is exhausted. The firmware algorithm for overwriting the address cache buffer is implemented in the FIFO fashion. When the scanner sees more than eight advertisers, the ninth advertiser replaces the first one, the tenth advertiser replaces the second one, and so on, in the address cache. After flushing the first advertiser from the address cache, if the scanner meets the first advertiser's ADV packet again, it treats it as a new device (because the first advertiser is no longer in the address cache) and sends an ADV packet to the host.*

Scan parameters

These parameters define the scanning time and interval between scanning events. Two different sets of Scan parameters are used: Fast scan parameters and Slow scan parameters. Typically, after the device initialization, a central device uses the Fast scan parameters. After the Fast scan timeout value expires, and if a connection with a Peripheral device is not set, then the Profile switches to the Slow scan parameters to save the battery life. After the Slow scan timeout value expires, the CY_BLE_EVT_GAPC_SCAN_START_STOP event is generated. See the API documentation.



Fast scan parameters – This connection type results in a faster connection between the GATT Client and Server devices than it is possible using normal connection.

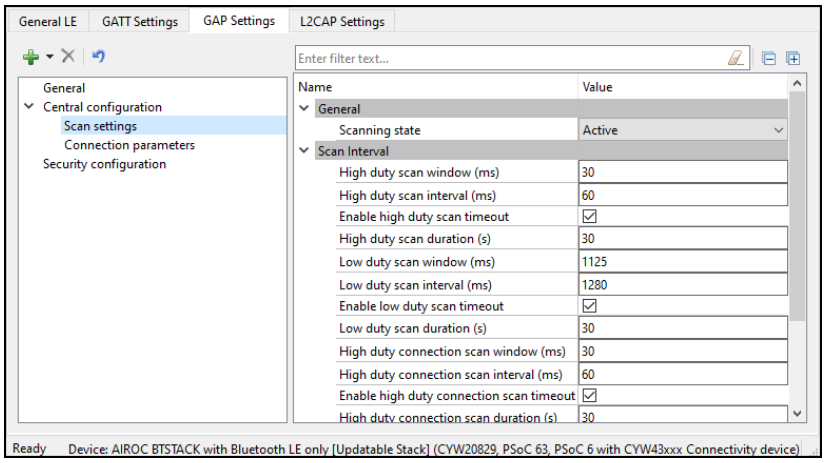
- Scan Window – This parameter defines the scan window when operating in Fast connection. The parameter is configured to increment in multiples of 0.625 ms. The valid range is from 2.5 ms to 10240 ms. Scan Window must be less than the Scan Interval. Default: 30 ms.
- Scan Interval – This parameter defines the scan interval when operating in Fast connection. The parameter is configured to increment in multiples of 0.625 ms. The valid range is from 2.5 ms to 10240 ms. Default: 30 ms.
- Scan Timeout – The timeout value of scanning with fast scan parameters. Default: 30 s. When not enabled, the device is scanning continuously. A timeout cannot occur before the scanning interval is expired, that is why if a timeout value is smaller than the slow scanning interval minimum value, a warning displays.

Slow scan parameters – This connection results in a slower than possible connection between the GATT Client and GATT Server devices that use a normal connection. However, this method consumes less power.

- Scan Window – This parameter defines the scan window when operating in Slow Connection. The parameter is configured to increment in multiples of 0.625ms. The valid range is from 2.5 ms to 10240 ms. Scan Window must be less than the Scan Interval. Default: 1125 ms.

- Scan Interval – This parameter defines the scan interval when operating in Slow Connection. The parameter is configured to increment in multiples of 0.625 ms. The valid range is from 2.5 ms to 10240 ms. Default: 1280 ms..
- Scan Timeout – The timeout value of scanning with slow scan parameters. Default: 150 s. When not enabled, the device is scanning continuously. A timeout cannot occur before the scanning interval expires, so if a timeout value is smaller than the slow scanning interval minimum value, a warning displays.

6.5.7.2 Option2



General

Scanning state

- Passive – A device can only listen to advertisement packets
- Active – A device may ask the advertiser for additional information

Scan interval

High duty scan parameters

- Scan Window – The parameter is configured to increment in multiples of 0.625 ms. Scan Window must be less than Scan Interval.
- Scan Interval – The parameter is configured to increment in multiples of 0.625 ms.
- Scan Duration – The timeout value of scanning with high-duty scan parameters. Default: 30 s. When not enabled, the device is scanning continuously

Low duty scan parameters

- Scan Window – The parameter is configured to increment in multiples of 0.625 ms. The valid range is from 2.5 ms to 10240 ms. Scan Window must be less than Scan Interval.
- Scan Interval – The parameter is configured to increment in multiples of 0.625 ms.
- Scan Duration –– The timeout value of scanning with low-duty scan parameters. Default: 30 s. When not enabled, the device is scanning continuously.

High duty connection scan parameters

- Scan Window – The parameter is configured to increment in multiples of 0.625 ms. Scan Window must be less than Scan Interval.
- Scan Interval – The parameter is configured to increment in multiples of 0.625 ms.
- Scan Duration – The timeout value of scanning with high duty connection scan parameters. Default: 30 s. When not enabled, the device is scanning continuously.

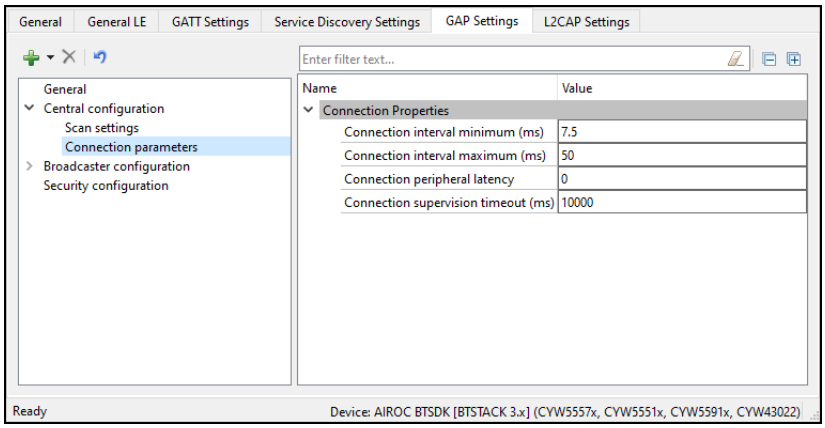
Low duty connection scan parameters

- Scan Window – The parameter is configured to increment in multiples of 0.625 ms. Scan Window must be less than Scan Interval

- Scan Interval – The parameter is configured to increment in multiples of 0.625 ms.
- Scan Duration – The timeout value of scanning with low-duty connection scan parameters. Default: 30 s. When not enabled, the device is scanning continuously.

6.5.8 GAP Settings tab Connection parameters

These parameters define the preferred Bluetooth® LE interface connection settings of the Central.



***Note**: The scaled values of these parameters are used internally by the Bluetooth® LE stack. These are the actual values sent over the air.*

- **Connection interval** – The Central device connecting to a Peripheral device needs to define the time interval for a connection to happen.
  - Minimum (ms) – This parameter is the minimum permissible connection time value to be used during a connection event. It is configured in steps of 1.25 ms. The valid range is from 7.5 ms to 4000 ms.
  - Maximum (ms) – This parameter is the maximum permissible connection time value to be used during a connection event. It is configured in steps of 1.25 ms. The valid range is from 7.5 ms to 4000 ms.

  ***Note**: For the multi-connection use case, recommended the minimum connection interval per connection greater than N * Max Time taken by individual connections to complete a Bluetooth® Connection Event (CE).*

  Min_CI = N * Average Time Per CE

  The average time for each CE is the amount of time taken to complete one Bluetooth® LE TX and RX transaction. This time varies depending on the Link Layer Data Length Extension (DLE) and Bluetooth® LE data rate (1 Mbps or 2 Mbps) configuration. The application can use the following timing lookup table for the CE value:

  - If DLE is enabled and data rate is 1Mbps, Average time = 6ms
  - If DLE is enabled and data rate is 2Mbps, Average time = 3.5ms
  - If DLE is disabled and data rate is 1Mbps, Average time = 2ms
  - If DLE is disabled and data rate is 2Mbps, Average time = 1.6ms

  For example, if an application supports 4 Bluetooth® LE connections with DLE and 1-Mbps data rate, the recommended minimum connection interval for each of the connections is:

  4 * 6 = 24 ms

  ***Note**: Connection intervals shorter than this value will still work, but under certain conditions, real-time control procedures (connection update, channel map update etc.) with a shorter update instance might result in a link disconnection.*

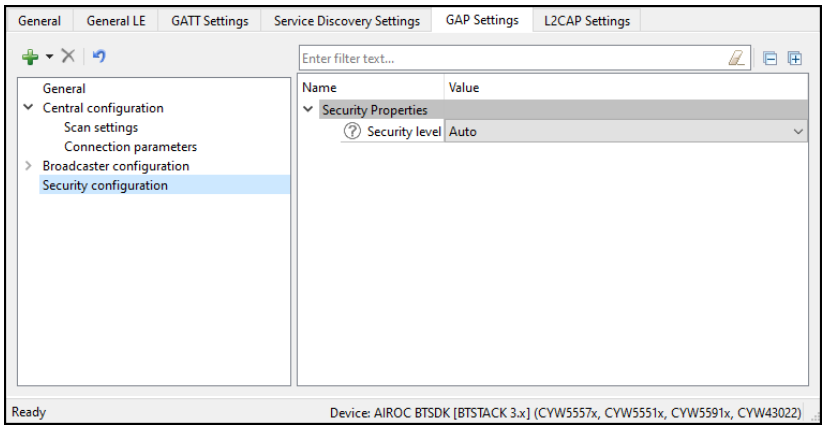
- **Peripheral Latency** – Defines the latency of the peripheral by responding to a connection event in consecutive connection events. This is expressed in terms of multiples of connection intervals, where only one connection event is allowed per interval. The valid range is 0 - 499 events.
- **Connection Supervision Timeout** – This parameter defines the LE link supervision timeout interval. It defines the timeout duration for which an LE link needs to be sustained if there is no response from the peer device over the LE link. The time interval is configured in multiples of 10 ms. The valid range is from 100 ms to 32000 ms.

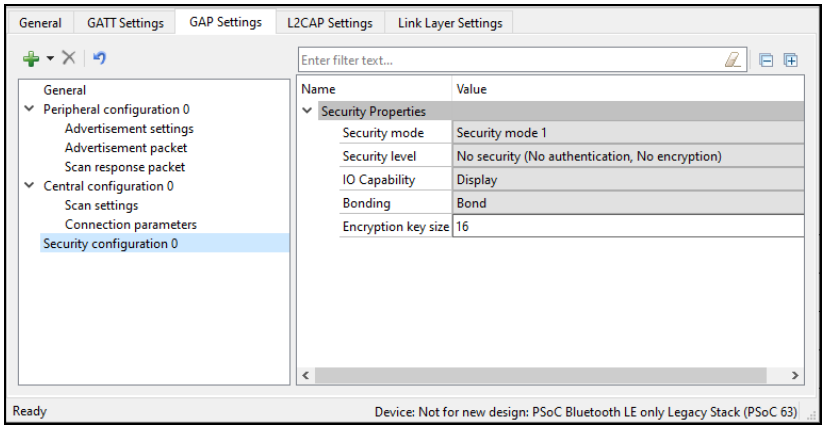
***Note**: For proper operation, the Connection Supervision Timeout must be larger than (1 + Peripheral latency) * Connection Interval * 2 (ms). Refer to Bluetooth® Core Specification Volume 6, Part B, Chapter 4.5.2 for more information on Connection Supervision Timeout.*

6.5.9 GAP Settings tab Security configuration

This Security configuration section contains several parameters to configure the global security options. **Option2**, **Option3**, and **Option4** devices have different sets of parameters.

These parameters are configurable only if a connectable GAP role, Peripheral or Central, is selected. You can optionally set each parameter using its own unique **Security Properties** in the **Profile Tree**.





Security Properties

Security mode

Defines the GAP security modes. Both available modes may support authentication:

- Mode 1 – For designs where data encryption is required.
- Mode 2 – For designs where data signing is required.

Security level

- **Option4** library: Enables different levels of security depending on the selected Security mode.
  - Mode 1
    - No security – The device does not use encryption or authentication.
    - Unauthenticated pairing with encryption – The device sends encrypted data after establishing a connection with the remote device.
    - Authenticated pairing with encryption – The device sends encrypted data after establishing a connection with the remote device. To establish a connection, the devices perform the authenticated paring procedure.
    - Authenticated LE Secure Connections pairing with encryption – The device uses an algorithm called Elliptic curve Diffie–Hellman (ECDH) for key generation, and a new pairing procedure for the key exchange. It also provides a new protection method from Man-In-The-Middle (MITM) attacks - Numeric Comparison.
  - Mode 2
    - Unauthenticated pairing with data signing – The device performs data signing prior to sending it to the remote device after a connection is set up.
    - Authenticated pairing with data signing – The device performs data signing prior to sending it to the remote device after a connection is set up. To establish a connection, the device performs the authenticated paring procedure.
- **Option2, Option3:**

  Provides the following levels of security:
  - Auto – The recommended choice for most applications to connect to the widest range of devices. Allows the stack to choose the highest level of security possible between the two devices.
  - Secure Connections – Can be set by applications that need to enforce secure connections.
  - Authentication – Can be set by applications that need to enforce Authentication.
  - Secure Connections with MITM protection – Can be set by applications that need to enforce secure connections with man-in-the-middle (MITM) protection.

  ***Note**: This parameter is used in BTSTACK 3.0 or above.*

- **Keypress notifications**

  Provides an option for a keyboard device during the LE secure pairing process to send key press notifications when the user enters or deletes a key. This option is available when the Security level is set to Authenticated LE Secure Connections pairing with encryption and I/O capabilities option is set to either Keyboard or Keyboard and Display.
- **I/O capabilities**

  Refers to the device input and output capability, which can enable or restrict a particular pairing method or security level.
  - Display – In devices with the display capability and may display authentication data. GAP authentication is required.
  - Display Yes/No – In devices with a display and at least two input keys for the Yes/No action. GAP authentication is required.
  - Keyboard – In devices with a numeric keypad. GAP authentication is required.
  - No Input/No Output – In devices without any capability to enter or display the authentication key data to the user. Used in mouse-like devices. No GAP authentication is required.
  - Keyboard and Display – In devices like PCs and tablets. GAP authentication is required.
- **Bonding Requirement**

Used to configure the bonding requirements. The purpose of bonding is to create a relation between two Bluetooth® devices based on a common link key (a bond). The link key is created and exchanged (pairing) during the bonding procedure and is expected to be stored by both Bluetooth® devices, to be used for future authentication. The maximum number of remote devices that can be bonded is 128.

- Bonding – The device stores the link key of a connection after paring with the remote device in the application flash memory. If a connection is lost and re-established, the device uses the previously stored key for the connection.

  ***Note**:* *Bonding information is stored in RAM and is written to the application flash to be retained during a shutdown.*

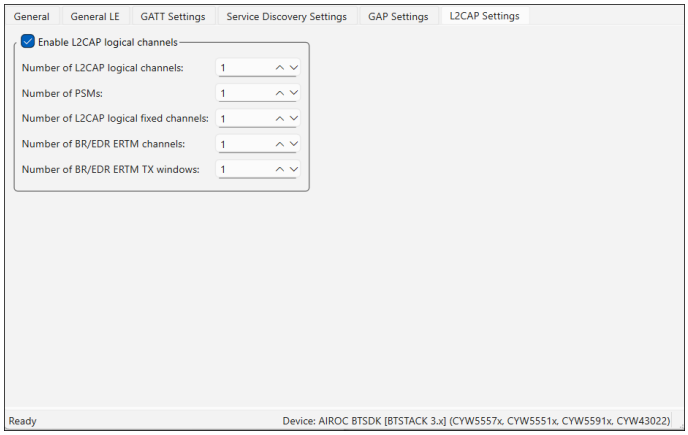
- No Bonding – The pairing process is performed on each connection establishment.

- **Encryption Key Size**

  Defines the encryption key size based on the Profile requirement. The valid values of an encryption key size are 7 - 16 bytes.

6.6 L2CAP Settings tab (Option2, Option3, Option4)

The **L2CAP Settings** define the parameters for L2CAP connection-oriented channel configuration.



6.6.1 Enable L2CAP logical channels

This parameter enables configuration of the L2CAP logical channels.

The default value for **Option4** is true, for **Option2** and **Option3** – false.

6.6.2 Number of L2CAP logical channels

This parameter defines the number of Low Energy **L2CAP** connection oriented logical channels required by the application. The valid range is 1 - 255. Default: 1.

6.6.3 Number of PSMs

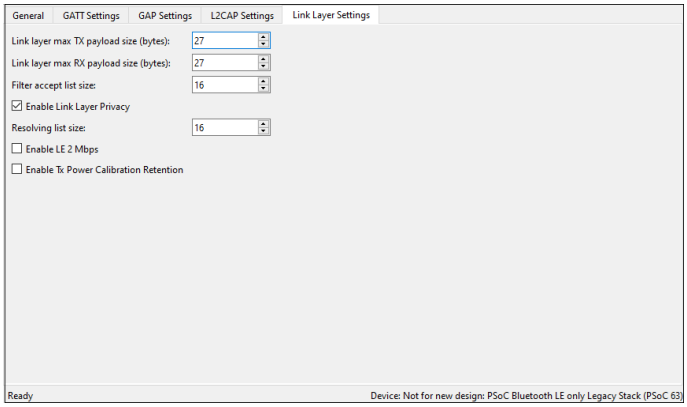
This parameter defines the number of PSMs required by the application. The valid range is 1 - 255. Default: 1.

6.6.4 L2CAP MTU size

This parameter defines the maximum SDU size of an L2CAP packet. The valid range is 23 - 65488 bytes. Default: 1280 bytes when Internet Protocol Support Service is supported and 23 bytes otherwise.

6.7 Link Layer Settings tab (Option4)

The **Link Layer Settings** parameters for the Link Layer.



6.7.1 Link Layer Max TX payload size

The maximum link layer transmits a payload size to be used in the design. The actual size of the link-layer transmit packet is decided based on the peer device's link-layer receive packet size during Data Length Update Procedure and will be informed through the `CY_BLE_EVT_GAP_DATA_LENGTH_CHANGE` event. The valid range is 27 - 251 bytes.

6.7.2 Link Layer Max RX Payload Size

The maximum link layer receives a payload size to be used in the design. The actual size of the link-layer receive packet is decided based on the peer device's link-layer transmit packet size during Data Length Update Procedure and will be informed through `CY_BLE_EVT_GAP_DATA_LENGTH_CHANGE` event. The valid range is 27 - 251 bytes.

Setting **Link Layer Max TX Payload Size** or **Link Layer Max RX Payload Size** to the value greater than 27 enables the Low Energy Data Length Extension feature.

6.7.3 Filter accept list size

The maximum number of devices that can be added to the **Filter accept list**. The valid range is 1 - 16. Default: 16.

6.7.4 Enable Link Layer Privacy

Enables LL Privacy 1.2 feature of Bluetooth® 4.2 and enables generation of `CY_BLE_EVT_GAP_ENHANCE_CONN_COMPLETE` and `CY_BLE_EVT_GAPC_DIRECT_ADV_REPORT` events.

***Note**: The `CY_BLE_EVT_GAP_DEVICE_CONNECTED` event is not generated when this feature is enabled.*

6.7.5 Resolving list size

The maximum number of peer devices whose addresses should be resolved by this device. This parameter is applicable when the Enable Link Layer Privacy feature is enabled. The valid range is 1 - 16. Default:16.

6.7.6 Enable LE 2 Mbps

Enables the LE 2 Mbps feature of Bluetooth® 5.0.

The LE 2 Mbps feature enables a new Physical (PHY) modulation scheme that allows increasing data throughput between the two devices which support this feature. Refer to Bluetooth® Core Specification v5.0 for more details about this feature.

Use the `Cy_BLE_SetDefaultPhy()` API after the `CY_BLE_EVT_STACK_ON` event to set the preferred default PHY for all connections, or the `Cy_BLE_SetPhy()` API to set PHY for the current connection.

The `CY_BLE_EVT_PHY_UPDATE_COMPLETE` event will indicate when the controller changes the transmitter PHY or receiver PHY in use.

6.7.7 Enable TX Power Calibration Retention

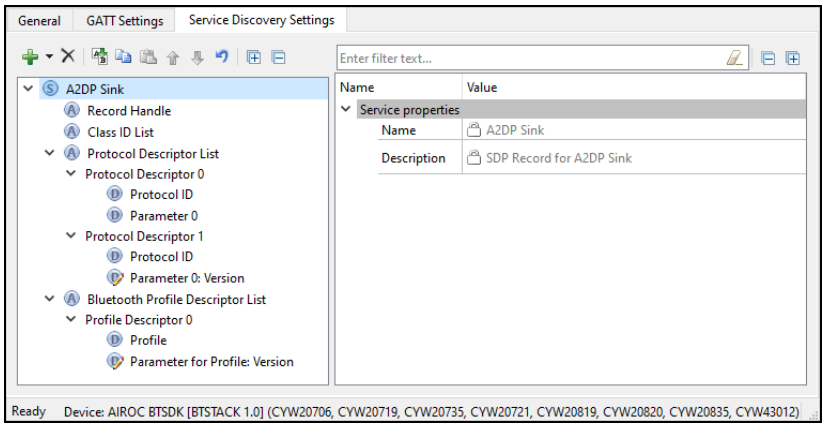
When enabled, the Bluetooth® Low Energy radio TX power calibration is performed only once after programming and the calibration values are retained in the Sflash location. This retained value is reloaded to radio power calibration registers during consecutive device reboots. This reduces the Bluetooth® LE stack boot-up time significantly.

***Note**: In Bluetooth® LE Dual-core mode, ensure to call the `Cy_SysDisableCM4()` function before enabling the Bluetooth® LE controller [that is, before calling `Cy_BLE_Enable()` on the controller core].*

***Note**: The calibration values are retained in user's row 0 (after `BLE_DEVICE_ADDRESS`) of the Sflash location.*

6.8 Service Discovery Settings tab (Option1, Option3) (Beta)

The **Service Discovery Settings** are used to configure SDP Services. This tab is available for Bluetooth® mode **Single mode BR/EDR** or **Dual mode**. The tab has three areas: toolbars, the Services tree, and the Parameter configuration pane.



6.8.1 Toolbars

The toolbars contain navigation options and means to add or delete Services and Attributes.

- **Add** – Allows adding a Service or an Attribute to the Services tree. To add an Attribute, highlight the parent Service in the Services tree. Refer to the Profiles section for the available Services.
- **Delete** – Deletes the selected Service or Attribute.
- **Rename** – Renames the selected item in the Services tree.
- **Copy/Paste** – Copies/pastes items in the Services tree.
- **Move Up** – Moves the selected item up in the Services tree
- **Move Down** – Moves the selected item down in the Services tree.
- **Reset branch to default** – Resets the selected item with child items in the Services tree to the default.

- **Expand All** – Expands all items in the Services tree.
- **Collapse All** – Collapses all items in the Services tree.

6.8.2 Services tree

The **Services** tree is used to view SDP Services, Attributes, Data element groups, and Data elements. By navigating through the tree, you can quickly add, delete, or modify Services and Attributes using the toolbar buttons or the context menu.

You can configure the parameters by clicking an item on the tree. These parameters will show in the Parameters configuration section.

The tree may contain the following nodes:

- Service
- Service Attribute
- Data element group: Data element. means that the Data element has editable fields.

In addition, nodes may have the following colors:

- Shaded icon – The node is mandatory
- White icon – The node is optional

6.8.3 Parameters configuration

The Parameters configuration pane allows you to view and configure a Service, Attribute, or Data element by selecting them in the tree. Attribute values are mainly hard-coded (and thus editing is disabled) to supply the required values according to the Bluetooth® Core Specification. Click through each Attribute and Data element under each Service to identify configurable values, and perform any additional configuration needed.

6.8.4 SDP services

The following SDP Services are available for selection:

- Advanced Audio Distribution Profile (A2DP) Sink
- Advanced Audio Distribution Profile (A2DP) Source
- A/V Remote Control (AVRC) Controller
- A/V Remote Control (AVRC) Target
- Device ID
- Hands-Free Audio Gateway
- Hands-Free Unit
- Human Interface Device
- Serial Port

7 Version changes

This section lists and describes the changes for each version of this tool.

Version	Change description
1.0	New tool.
1.1	Added 20xxx devices support.
	Fixed the issue with code generation for GAP Central role connection parameters. Their values were always generated as default.
2.0	Added the Bluetooth® BR/EDR support for 20xxx devices (Beta version). Added the General and Service Discovery Settings tabs for 20xxx devices.
	Changed the user configuration storage location. Previously the configuration was stored in the header file as a comment in XML format. Now it is stored in the *.cybt file in XML format.
	Added Custom OTA Firmware Upgrade Service for 20xxx devices.
	Minor changes in the generated code for 20xxx devices: 'u' suffix not added to unsigned integer values.
	Changed Bluetooth® LE Stack RAM memory allocation (PSoC 6 devices).
	Changed the names of the advertisement packet defines (PSoC 6 devices). Before: CY_BLE_ADV_PKT_<INDEX>_INDEX<NAME>; After: CY_BLE_ADV_PKT_<INDEX>_INDEX_<NAME>.
	Added New, Close, Import, Export, and Reset View commands. User interface details were improved.
	Removed the functionality to launch the Bluetooth® Configurator from the Device Configurator.
2.1	Added menu Edit with two items Undo/Redo.
	Fixed minor issues in the representation of the GATT-profiles properties.
	Modified the default selection of the GATT-characteristic permissions behavior for 20xxx devices.
2.20	Added the PSoC 6 MCU with 43xxx Connectivity support.
	Added Copy feature to the Notice List.
	Fixed issues in the HID report editor.
	Fixed issues in the Advertisement packet editor.
	Improved validation of fields.
	Removed the Profile role Weight Scale and Collector from the Add Profiles menu.
2.21	Added a General tab for 43xxx devices.
	Added a possibility to disable the GATT database for 43xxx devices.
	Added new parameters for 43xxx devices: Maximum remote servers and clients connections, and Maximum attribute length.
	Changed the maximum limit of the MTU Size parameter from 512 to 517 for 43xxx devices.
	Fixed the issue with the CY_BT_L2CAP_MTU_SIZE value for 43xxx devices. In case when L2CAP logical channels were disabled, the default value generated for this define was 23 instead of 512 if the configurator's output files were generated by the build script (not GUI itself).

7 Version changes

Version	Change description
	Fixed the issue that the Peripheral GAP role was restored after the configurator reopening even when previously removed (43xxx devices).
	Advertisement packet configuration: in case when the GATT Settings tab is disabled, the GATT services configured previously on the GATT Setting tab are not available for selection for the Advertisement packet Service UUID and Service Data AD types.
	The configurations with characteristics of length 0 produced empty arrays in the generated code for 20xxx and 43xxx devices. This could cause build warnings on some compilers. Added an extra '0' element to the empty arrays to avoid that.
	Removed duplicated CY_BT_HIGH_DUTY_CONN_SCAN_WINDOW define.
	GAP scan connection parameters: removed possibility to disable Connection interval minimum/maximum and Connection supervision timeout.
	GAP scan interval settings for 43xxx: corrected values of the following defines: CY_BT_HIGH_DUTY_SCAN_DURATION, CY_BT_LOW_DUTY_SCAN_DURATION, CY_BT_HIGH_DUTY_CONN_SCAN_DURATION, CY_BT_LOW_DUTY_CONN_SCAN_DURATION.
	When the GAP role is non-connectable (Broadcaster or Observer), the server_max_links field in the wiced_bt_cfg_settings struct is set to 0 instead of 1.
2.30	Added characteristic permissions validation for 43xxx devices.
	Removed the command-line generate options: -g and --generate.
	Configurations with HID service without the Report characteristics produced an empty array in the generated code for Bluetooth MCU. Changed the generated code to avoid that.
2.40	Added support of the BTSTACK 3.0 library.
	Added the Security level parameter for 43xxx devices.
	Added the possibility to generate a random Device address based on the PC's MAC address.
	For Bluetooth MCU, fixed minor inconsistencies in the generated code between bt-configurator and bt-configurator-cli executables.
	43xxx devices: wiced_bt_cfg_settings struct uses a device name declared in cycfg_gatt_db.h (app_gap_device_name) instead of cy_bt_device_name. cy_bt_device_name is obsolete. Added '\0' at the end of app_gap_device_name initialization.
	43xxx devices: modified the initialization value for the .max_simultaneous_links field of the wiced_bt_cfg_settings struct from 3 to (CY_BT_CLIENT_MAX_LINKS + CY_BT_SERVER_MAX_LINKS).
	Added cleanup of the generated source files before saving new files in order to remove not relevant files.
2.50	Added support for CYW20829 and CYW5557x devices. Renamed the existing device options.
	Added a new parameter RX PDU size on the General LE tab, which is used to initialize ble_max_rx_pdu_size field for BTSTACK 3.0 or above.

7 Version changes

Version	Change description
	Improved the HID Report editor: for Input, Output, and Feature items added a drop-down with the options. Removed the CCCD descriptor for those HID Report characteristics where it's not required. When the HID service is present, added possibility to add the External Report Reference descriptor to any characteristic in the profile.
	The GATT DB attribute handles in the generated code formatted as 16-bit values.
	Removed: the migration of configuration to the current XML format – configuration saved in the comments in generated HEADER files (the old method).
2.60	Added support for multi-advertisement.
	Renamed terms according to the "Appropriate Language Mapping Tables" by Bluetooth® SIG.
2.61	Added the Basic Audio, Call Control, Common Audio, Coordinated Set Identification, Hearing Access, Public Broadcast, Telephony and Media Audio profiles.
	Added missed version information.
2.70	The PSoC™ Bluetooth® LE only Legacy Stack (PSoC™ 63) library support starts being deprecated.
2.80	• Improved the logic for automatic Device/Middleware selection for ModusToolbox™ applications. • Added "Max number of BIG connections" field to "Isochronous Connection Configuration" structure on the General tab. • Fixed incorrect "Peripheral Connection Interval Range" millisecond values on the GAP tab
2.81	Added support for the CYW5551x and CYW5591x devices.
2.90	• Added new validation rules to check configuration settings consistency on GAP and GATT tabs. • Fixed bugs and typos in several GATT profiles and services definitions. • Fixed bug in generated code for characteristic descriptors. • Double-click on error or notice message now moves focus to specific field or tree node for General, General LE, GATT, L2CAP tabs. • "Isochronous Connection Configuration" settings group moved from General to General LE tab. • 'const' qualifier now can be optionally removed from some configuration structures in generated code. Corresponding check-box is available on General tab for CYW43xxx and CYW55xxx.
3.0	• Improved integration with the Device DB. It is used for automatic Stack Option (Middleware library) selection and device parameter discovery. • Changed supported TX power levels for CYW20829 device. • Added support for 'Google® Fast Pair' GATT service and its characteristics. • Added support for 'Detect Unwanted Location Trackers' GATT service and its characteristics. • Fixed wrong Battery service characteristics UUIDs • Various backend changes.
3.10	Minor back-end changes.

7 Version changes

Version	Change description
3.20	Isochronous connection configuration structure is conditionally excluded in generated sources for BTSTACK version 4.0 and later.
3.30	Minor back-end changes.
3.40	• Added Ranging Service and Profile support • Added support for GATT service import/export • The Host Address Resolution DB Size parameter is now configurable on the GUI (**GAP Settings Tab -> General section**)
3.50	• Added support for GATT service import/export • Added Electronic Shelf Label Service and Profile support • Parameters **Maximum DB service modules** and **Maximum EATT bearers** are now configurable from the GUI **General LE tab**
3.60	Minor back-end changes.
3.70	• Updated several GATT Services to the latest version • Fixed light and dark theme color issues • Minor back-end changes

Revision history

Revision	Date	Description
**	2018-11-12	New document.
*A	2019-02-26	Updated to version 1.1.
*B	2019-10-16	Updated to version 2.0.
*C	2020-03-27	Updated to version 2.1.
*D	2020-04-13	Updated a screen capture and corrected a sentence.
*E	2020-09-01	Updated to version 2.20.
*F	2020-12-14	Updated to version 2.21.
*G	2021-03-12	Updated to version 2.30.
*H	2021-04-26	Updated to version 2.40.
*I	2021-09-29	Updated to version 2.50.
*J	2022-09-29	Updated to version 2.60.
*K	2022-10-12	Added Multi-advertisement feature description.
*L	2023-03-02	Updated to version 2.70.
*M	2023-05-26	Updated to version 2.80.
*N	2023-09-25	Updated to version 2.81.
*O	2023-10-11	Minor changes.
*P	2024-01-19	Updated to version 2.90.
*Q	2024-10-01	• Updated to version 3.00. • Removed "Short name" terms. Middleware type **Options** are designated by **Option1...4** throughout the user guide.
*R	2024-12-05	Updated to version 3.10
*S	2025-02-07	Updated to version 3.20
*T	2025-03-21	Updated to version 3.30
*U	2025-09-01	Updated to version 3.40
*V	2025-12-11	Updated to version 3.50
*W	2026-03-21	Updated to version 3.60
*X	2026-08-14	Updated to version 3.70

Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2026-08-14

Published by

Infineon Technologies AG

81726 Munich, Germany

© 2026 Infineon Technologies AG

All Rights Reserved.

Do you have a question about any aspect of this document?

Email: erratum@infineon.com

Document reference
IFX-zdy1711641967626

Important notice

Products which may also include samples and may be comprised of hardware or software or both ("Product(s)") are sold or provided and delivered by Infineon Technologies AG and its affiliates ("Infineon") subject to the terms and conditions of the frame supply contract or other written agreement(s) executed by a customer and Infineon or, in the absence of the foregoing, the applicable Sales Conditions of Infineon. General terms and conditions of a customer or deviations from applicable Sales Conditions of Infineon shall only be binding for Infineon if and to the extent Infineon has given its express written consent.

For the avoidance of doubt, Infineon disclaims all warranties of non-infringement of third-party rights and implied warranties such as warranties of fitness for a specific use/purpose or merchantability. Infineon shall not be responsible for any information with respect to samples, the application or customer's specific use of any Product or for any examples or typical values given in this document.

If this document is marked as 'Preliminary', 'Target', 'Draft' or 'Proposal', Infineon reserves the right to change all information given in this document at any time without notice. Subject to the development and release of a Product for series supply by Infineon, the technical specifications of the Product are set forth in the relevant datasheet provided by Infineon without such marking.

The data contained in this document is exclusively intended for technically qualified and skilled customer representatives. It is the responsibility of the customer to evaluate the suitability of the Product for the intended application and the customer's specific use and to verify all relevant technical data contained in this document in the intended application and the customer's specific use. The customer is responsible for properly designing, programming, and testing the functionality and safety of the intended application, as well as complying with any legal requirements related to its use.

Unless otherwise explicitly approved by Infineon, Products may not be used in any application where a failure of the Products or any consequences of the use thereof can reasonably be expected to result in personal injury.

However, the foregoing shall not prevent the customer from using any Product in such fields of use that Infineon has explicitly designed and sold it for, provided that the overall responsibility for the application lies with the customer. Infineon expressly reserves the right to use its content for commercial text and data mining (TDM) according to applicable laws, e.g. Section 44b of the German Copyright Act (UrhG). If the Product includes security features:

Because no computing device can be absolutely secure, and despite security measures implemented in the Product, Infineon does not guarantee that the Product will be free from intrusion, data theft or loss, or other breaches ("Security Breaches"), and Infineon shall have no liability arising out of any Security Breaches.

If this document includes or references software:

The software is owned by Infineon under the intellectual property laws and treaties of the United States, Germany, and other countries worldwide. All rights reserved. Therefore, you may use the software only as provided in the software license agreement accompanying the software.

If no software license agreement applies, Infineon hereby grants you a personal, non-exclusive, non-transferable license (without the right to sublicense) under its intellectual property rights in the software (a) for software provided in source code form, to modify and reproduce the software solely for use with Infineon hardware products, only internally within your organization, and (b) to distribute the software in binary code form externally to end users, solely for use on Infineon hardware products. Any other use, reproduction, modification, translation, or compilation of the software is prohibited.

For further information on the Product, technology, delivery terms and conditions, and prices, please contact your nearest Infineon office or visit www.infineon.com.