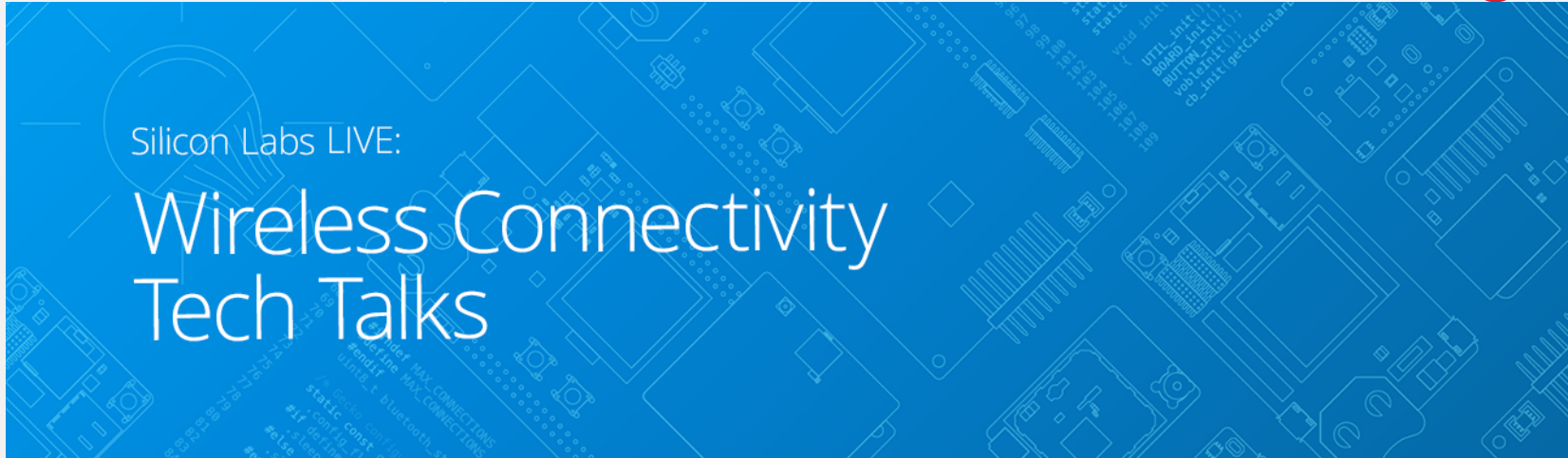


Tech Talks LIVE Schedule – Presentation will begin shortly



Topic	Date
Building a Proper Mesh Test Environment: How This Was Solved in Boston	Thursday, July 2
Secure Your Bluetooth Design with BG21/BG22	Thursday, July 23
New Bluetooth Mesh Light & Sensor Models	Thursday, July 30
Simplicity Studio v5 Introduction	Thursday, August 6
Long-Range Connectivity Using Proprietary RF Solution	Thursday, August 13
Wake Bluetooth from Deep Sleep Using an RF Signal	Thursday, August 20
Implementing a Bluetooth Network Co-Processor	Thursday, August 27

Fill out the survey for a chance to win a BG22 Thunderboard!



Find Past Recorded Sessions at:
<https://www.silabs.com/support/training>



Implementing a Bluetooth® Network Co-Processor

AUG 2020



Agenda

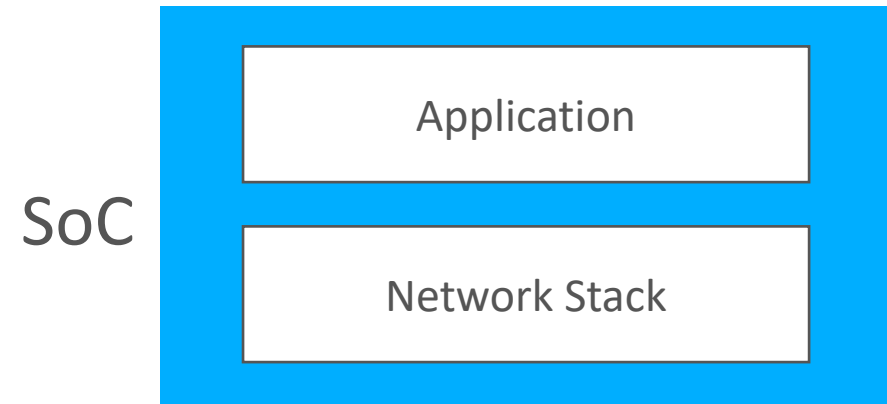
- What is a Bluetooth Network Co-Processor?
- How does it work?
- How can you implement one?
- What all can you use it for?



Network Co-Processor (NCP): Definition

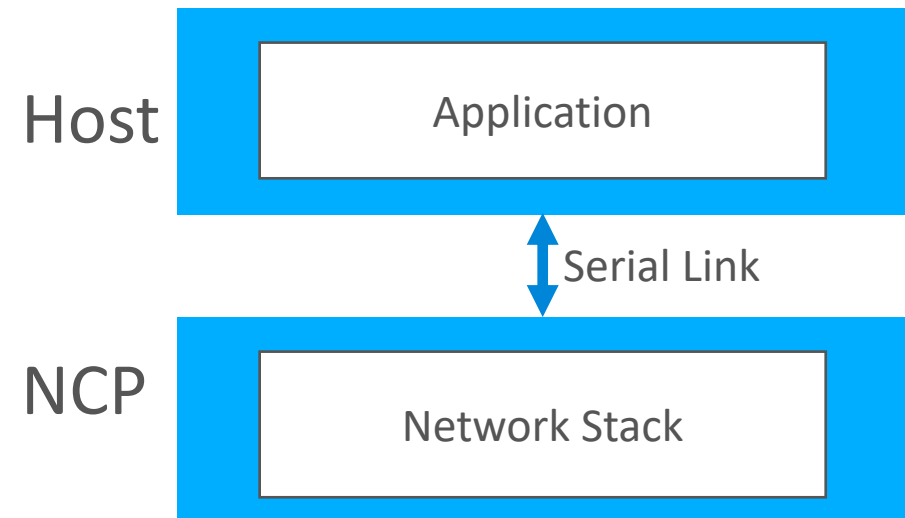
System on Chip (SoC)

- Application and Network Stack are both resident in the same device



Network Co-Processor (NCP)

- Network Stack is on the NCP device
- Application is on a separate host processor
- A serial link is used to communicate between the host processor and the NCP



NCP Use Cases

The NCP architecture is typically used when it's easier or more efficient to manage the application on a separate processor than the processor on which the network stack resides.

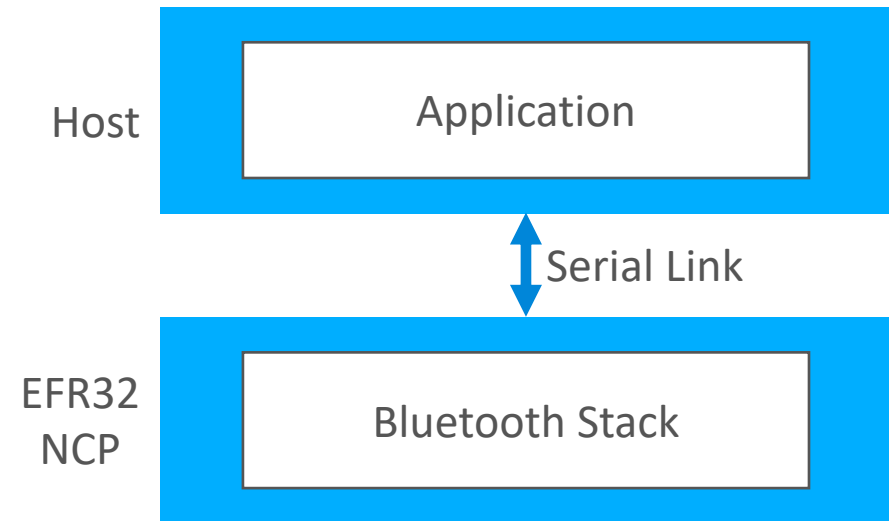
- Gateways
- Test Fixtures
- Adding wireless interfaces to specialized SoCs
- High performance multi-protocol (SoC + NCP)



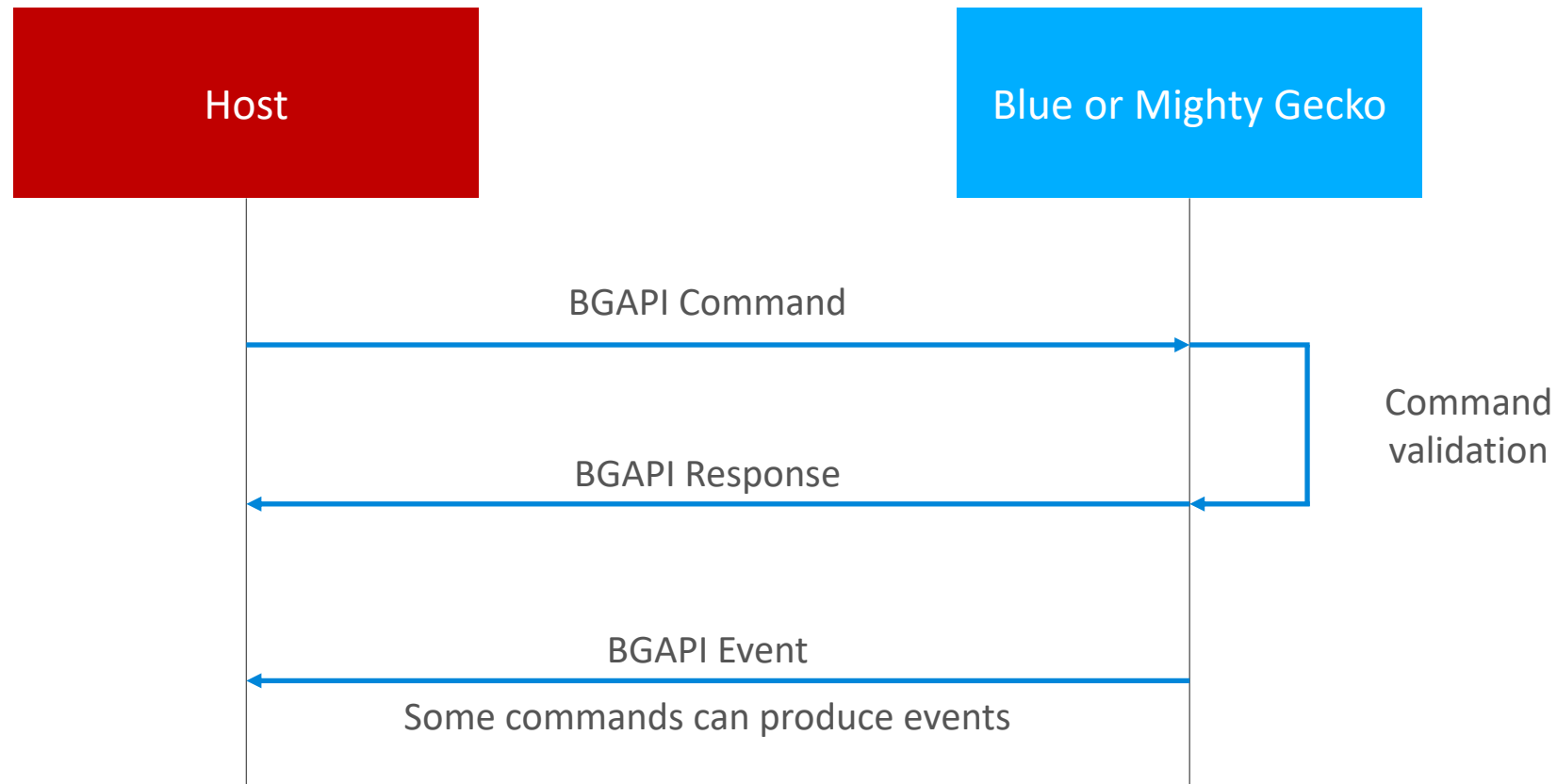
Blue Gecko NCP

Well Defined Serial API (BGAPI)

- Command
 - Response
 - Event
-
- Portable C BGAPI library code included in the SDK
 - XML API definition included for generating libraries for other programming languages
-
- Runs on any EFR32MG or EFR32BG device
 - BGAPI NCP host code is identical to EFR32 SoC code!



BGAPI Serial Protocol Message Exchange



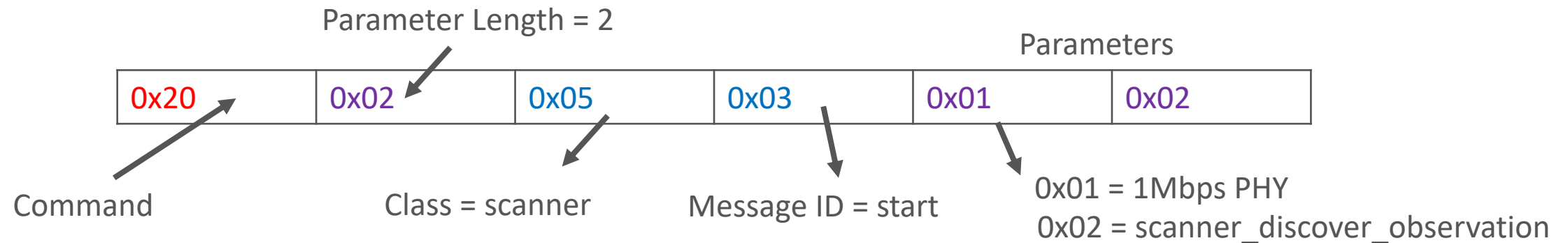
BGAPI Serial Protocol Packet Structure

Byte	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4-255
Explanation	Message type	Payload Length	Message Class	Message ID	Payload
Values	0x20: command 0x20: response 0xA0: event	0x00 – 0xFF	0x00 – 0xFF	0x00 – 0xFF	Specific to command, response, or event

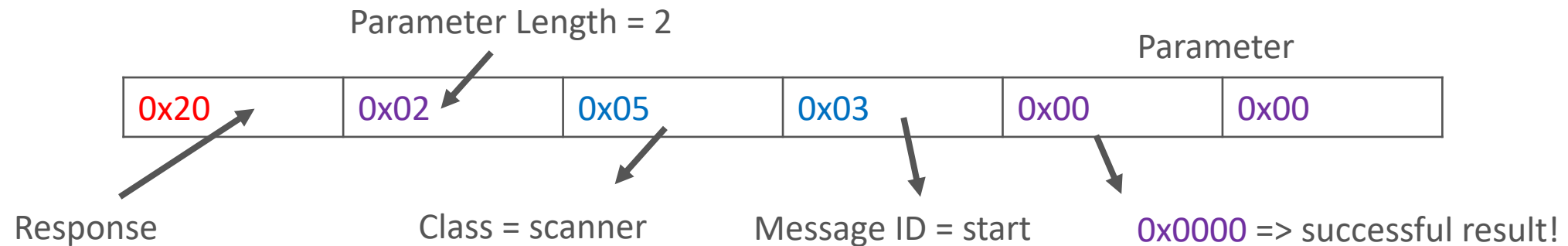
Blue Gecko NCP: Example

Command: sl_bt_scanner_start

- Begin scanning for advertising with previously configured parameters along with command parameters



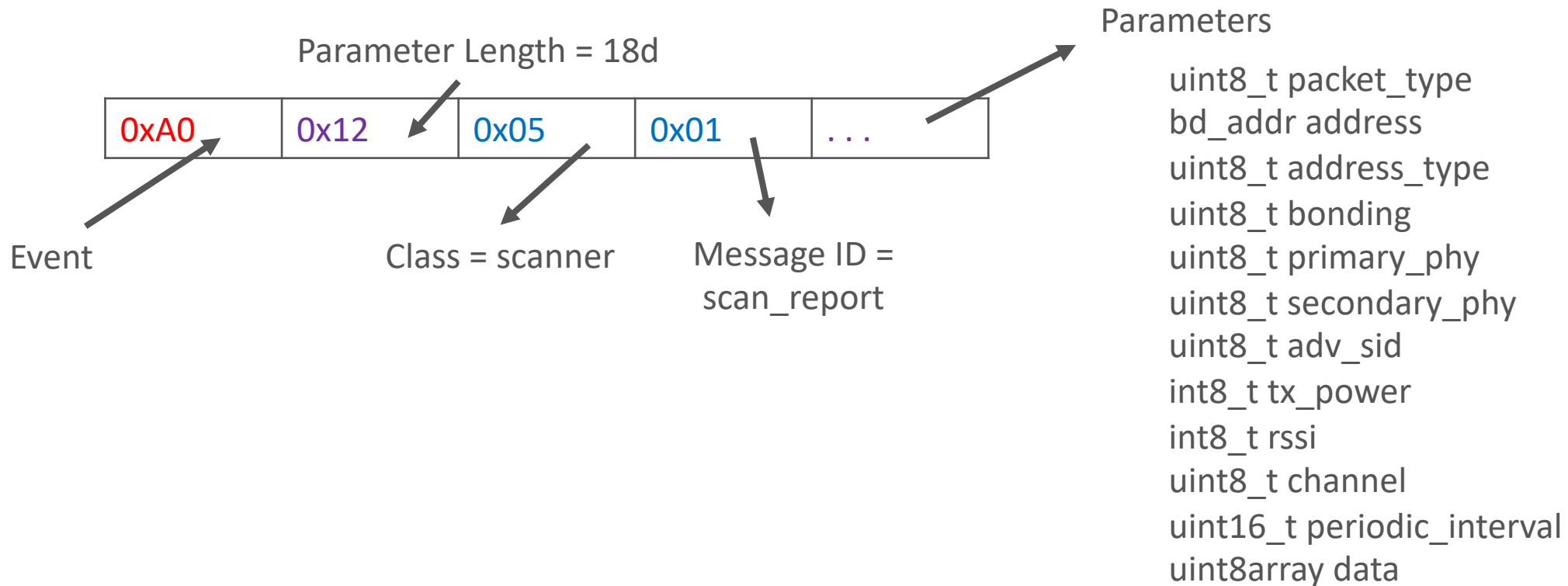
Response: sl_bt_scanner_start




Blue Gecko NCP: Example

Event: `sl_bt_evt_scanner_scan_report`

- Event triggered each time an advertising packet is received in scanning mode. Contains all the data for each received advertising (address, payload, etc.)



Blue Gecko NCP: Documentation



docs.silabs.com

search

Bluetooth

- Scanner
 - sl_bt_evt_scanner_scan_report
 - scanner_discover_mode_t
 - sl_bt_scanner_set_timing
 - sl_bt_scanner_set_mode
 - sl_bt_scanner_start
 - sl_bt_scanner_stop
- Synchronization
- Connection
- GATT Client
- GATT Server
- NVM
- Testing Commands
- Security Manager
- OTA
- Coexistence
- CTE Transmitter

◆ sl_bt_scanner_start()

```
sl_status_t sl_bt_scanner_start( uint8_t scanning_phy,
                                uint8_t discover_mode
                                )
```

Start the GAP discovery procedure to scan for advertising devices on the specified scanning PHY or to perform a device discovery. To cancel an ongoing discovery process use the [sl_bt_scanner_stop](#) command.

The invalid parameter error will be returned if the scanning PHY value is invalid or the device does not support the PHY.

Parameters

[in]	scanning_phy	Enum gap_phy_type_t . The scanning PHY. Values: <ul style="list-style-type: none">• gap_1m_phy (0x1): 1M PHY• gap_coded_phy (0x4): Coded PHY
[in]	discover_mode	Enum scanner_discover_mode_t . Bluetooth discovery Mode. Values: <ul style="list-style-type: none">• scanner_discover_limited (0x0): Discover only limited discoverable devices.• scanner_discover_generic (0x1): Discover limited and generic discoverable devices.• scanner_discover_observation (0x2): Discover all devices.


Returns

SL_STATUS_OK if successful. Error code otherwise.

Events

- [sl_bt_evt_scanner_scan_report](#) - This event is triggered each time an advertising packet is received. Packets are not filtered in any way, so multiple events will be received for every advertising device in range.

Blue Gecko NCP: Documentation



SILICON LABS
docs.silabs.com

search

Bluetooth

- Common types
- BGAPI message ID summary**
- Error Codes
- BGAPI Headers

Lab Manuals

- Getting Started with Silicon Labs Bluetooth
- Bluetooth 5 Features Lab Manual >

Code Examples

- Stack Features >
- Applications >
- Peripheral >
- Tools >

Revision History

- Document Revisions
- Release Notes

You are viewing documentation for version: **3.0** (latest) | [2.13](#) | [Version History](#)

BGAPI message ID summary

Summary of BGAPI command, response, and event IDs.

The following table summarizes the command, response, and event IDs used in the BGAPI protocol. The table shows the minimum payload length for each message. Messages that have an array parameter can have longer payload length depending on the length of the array.

BGAPI message ID summary

Name	Type	Minimum Payload Length	Class	Message ID
Scanner				
cmd_scanner_set_timing	0x20	0x05	0x05	0x01
rsp_scanner_set_timing	0x20	0x05	0x05	0x01
cmd_scanner_set_mode	0x20	0x02	0x05	0x02
rsp_scanner_set_mode	0x20	0x02	0x05	0x02
cmd_scanner_start	0x20	0x02	0x05	0x03
rsp_scanner_start	0x20	0x02	0x05	0x03
cmd_scanner_stop	0x20	0x00	0x05	0x05
rsp_scanner_stop	0x20	0x00	0x05	0x05
evt_scanner_scan_report	0xa0	0x12	0x05	0x01
Synchronization				

BGAPI 3.x vs. 2.x

- BGAPI has been revised with the Bluetooth SDK 3.0
- The new API commands and events use a new naming convention to comply with Silicon Labs standards
- New BGAPI classes and commands have been introduced and some removed to make the API more transparent and consistent
- API changes are described in section 5 of [AN1255: Transitioning from the v2.x to the v3.x Bluetooth® SDK](#)
- **BGAPI 2.x is not compatible with BGAPI 3.x!**



AN1255: Transitioning from the v2.x to the v3.x *Bluetooth*® SDK

Bluetooth Software Development Kit (SDK) v3.0 contains a number of changes compared to Bluetooth SDK v2.x. Many of these changes are due to an underlying framework redesign that results in an improved developer experience within the new Simplicity Studio 5. Projects are now built on a component architecture. Simplicity Studio 5 includes project configuration tools that provide an enhanced level of software component discoverability, configurability, and dependency management. These include a Component Editor and a redesigned GATT configurator.

KEY POINTS

- Reviews differences in:
- Software architecture
 - The API

Producing Blue Gecko NCP Target Firmware: Demo

Pre-built example NCP firmware images are available within Simplicity Studio for most Silicon Labs boards

1. Plug board into USB port
2. Go to the Simplicity Studio "Launcher" perspective
3. Select target board under "Debug Adapters"
4. Select "Demo".
5. Click "Run" next to "Bluetooth – NCP Empty"



Thunderboard Sense 2



BG22 Thunderboard



Wireless Starter Kit (WSTK) with Radio Board

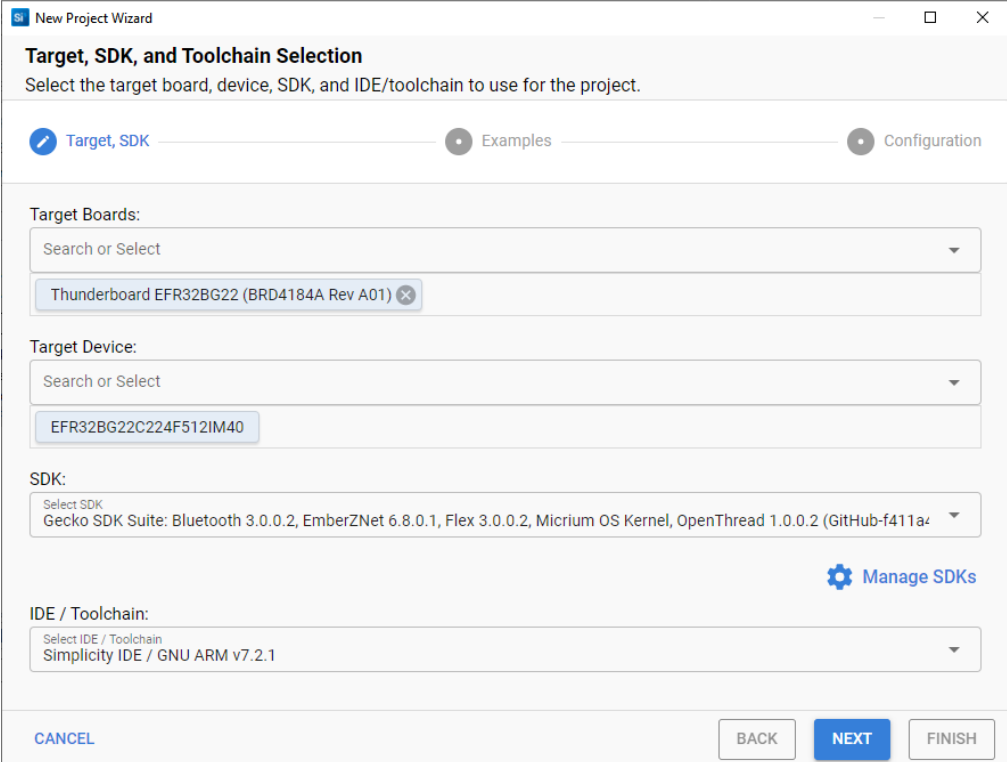
Producing Blue Gecko NCP Target Firmware: Demo

The screenshot displays the Simplicity IDE interface. On the left, the 'Debug Adapters: 5' window shows a tree view of hardware components. The 'Thunderboard EFR32BG22 (ID: 440172627)' is selected, indicated by a red arrow labeled '3'. The main workspace shows the 'Thunderboard EFR32BG22 (ID: 000440172627)' target selected, with a red arrow labeled '2' pointing to the title. Below the target name are tabs for 'OVERVIEW', 'EXAMPLE PROJECTS', 'DOCUMENTATION', 'DEMOS', and 'COMPATIBLE TOOLS'. The 'DEMOS' tab is active, indicated by a red arrow labeled '4'. The main content area displays 'Run a pre-compiled demo to test on your device' and '4 resources found'. A search box labeled 'Filter on keywords' is present. One resource is visible: 'Bluetooth - NCP Empty', described as a 'Bluetooth NCP (Network Co-Processor) target application with a minimal GATT database, that makes it possible to access the Bluetooth stack from a host controller via'. A red arrow labeled '5' points to the 'RUN' button associated with this resource.

Producing Blue Gecko NCP Target Firmware

For boards on which the pre-built demo firmware is not available (including your own custom hardware), you can create and build your own NCP firmware project

1. Select File->Silicon Labs Project Wizard
2. Select target board or target device (if custom board).
3. Select "Bluetooth – NCP Empty"



The screenshot shows the 'New Project Wizard' dialog box with the title 'Target, SDK, and Toolchain Selection'. The instruction reads: 'Select the target board, device, SDK, and IDE/toolchain to use for the project.' The progress bar shows three steps: 'Target, SDK' (active), 'Examples', and 'Configuration'. The 'Target Boards' section has a search box with 'Thunderboard EFR32BG22 (BRD4184A Rev A01)' selected. The 'Target Device' section has a search box with 'EFR32BG22C224F512IM40' selected. The 'SDK' section has a search box with 'Gecko SDK Suite: Bluetooth 3.0.0.2, EmberZNet 6.8.0.1, Flex 3.0.0.2, Micrium OS Kernel, OpenThread 1.0.0.2 (GitHub-f411a4)' selected, and a 'Manage SDKs' link. The 'IDE / Toolchain' section has a search box with 'Simplicity IDE / GNU ARM v7.2.1' selected. At the bottom, there are 'CANCEL', 'BACK', 'NEXT', and 'FINISH' buttons.

Other Useful NCP Features

Custom BGAPI

- Implement custom BGAPI NCP commands for custom functionality in the NCP firmware
- Examples:
 - System temperature sensing
 - UI handling (LEDs, buttons, etc.)

Local Event Handling

- Implementing local handlers in the NCP firmware reduces the amount of messages that the host has to service
- Example: advertisement filtering – only send advertisement events to the host that meet specific criteria

Other Useful NCP Features

Low Power Support

- Implemented with wake-lock pins
- Wake-lock input: when asserted by external host, wakes and then sends event `sl_bt_evt_system_awake` to indicate to the host that it has woken up.
- Wake-lock output: NCP can assert prior to sending an event (allows host to sleep too)

Angle of Arrival / Angle of Departure NCP Support

- EFR32BG22 supports CTE transmit and receive commands over NCP for Angle of Arrival / Angle of Departure implementations
- **Wi-Fi Coexistence**

Blue Gecko NCP Host Software: BGTool

Simplicity Studio includes an NCP host GUI utility called BGTool

Provides the ability to command the NCP to:

- Transmit advertising
- Receive advertising
- Establish a connection with a connectable device and browse the remote GATT database
- Perform DTM transmit and receive tests
- Execute arbitrary API commands

The screenshot displays the BGTool interface for Bluetooth Smart. The top navigation bar includes 'Bluetooth Smart', 'Security Manager', and 'Persistent storage'. The main content area is titled 'Bluetooth Smart' and contains a 'Generic Access Profile' section. This section has two tabs: 'Basic settings' (selected) and 'Advanced settings'. Under 'Basic settings', there are two main sections: 'Advertise (Slave)' and 'Discover (Master)'. The 'Advertise (Slave)' section includes 'Discoverable mode' (with options: Non-discoverable, Limited discoverable, General discoverable (selected), Broadcast, User data) and 'Connectable Mode' (with options: Non-connectable, Connectable-scannable (selected), Scannable-not-connectable, Connectable-not-scannable). There are 'Start' and 'Stop' buttons. The 'Discover (Master)' section includes 'Scan PHY' (with options: 1M PHY (selected), Coded PHY) and 'Scan type' (with options: Limited, Generic (selected), Observation). Below these settings is a table with columns: Address, RSSI (dBm), Bonding handle, Advertising data, Scan Response, and a 'Connect' button. The table contains one entry with Address '6f:37:ad:c5:03:b9', RSSI '-49', Bonding handle '255 (=not bonded)', and Advertising data 'Raw data: 0x0201060aff4c0010050b189d7753'. At the bottom, there is a 'Log' section with a 'Settings' dropdown and a list of log entries. The log entries show 'gecko_evt_le_gap_scan_response' messages with details like rssi, packet_type, address, address_type, bonding, and data. Below the log is a 'BGAPI commands' input field with 'Send', 'Save', and 'Clear' buttons. At the very bottom, there is a status bar with 'Interactive view', 'Device Details', and 'Connected (tty.SLAB_USBtoUART)'.

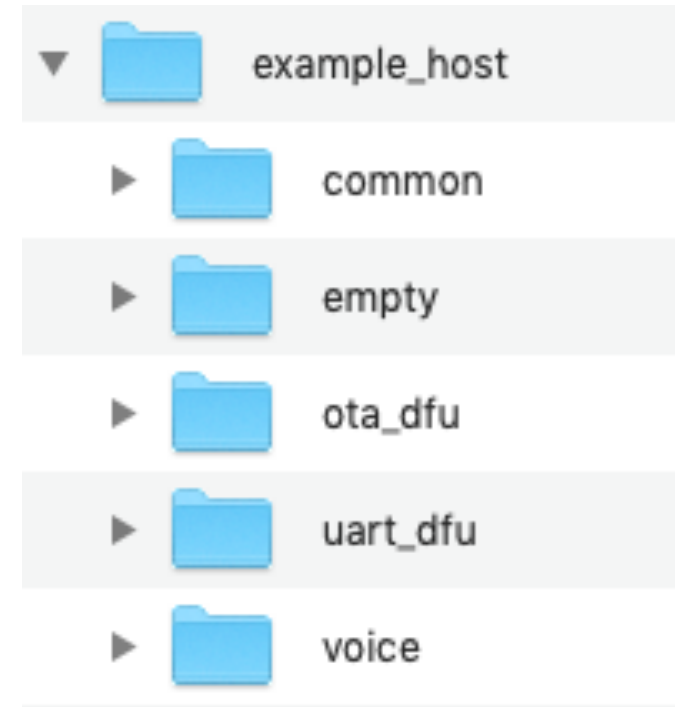
Blue Gecko NCP Host Software

The Silicon Labs Bluetooth SDK includes several example host applications as portable GCC C language projects with makefile:

Studio v5 (Gecko SDK 3.x): `app\bluetooth\example_host\`

Studio v4 (Gecko SDK 2.x): `app\bluetooth\examples_NCP_host\`

- **empty** – An empty “shell” project to use to implement your own application
- **ota_dfu** – Used to perform an OTA firmware update to a device running the Silicon Labs Apploader
- **uart_dfu** – Used to perform a serial firmware update to the NCP device
- **voice** – Example using BLE to collect ADPCM compressed voice data



Building Blue Gecko NCP Host Examples

Builds natively on Mac or Linux – just go to the folder using a terminal and type “make”

For Windows, there are two options (both also use gcc make):

1. MinGW

- Minimalist GNU development environment for Windows
- Builds native MS-Windows command-line applications
- <https://mingw.org/>

2. Cygwin

- POSIX Emulator for Windows
- Build executables that run within the Cygwin environment
- <http://cygwin.com/>

3rd Party Blue Gecko NCP Host Software

There are several Blue Gecko NCP host implementations that are not official Silicon Labs releases, but are quite useful:

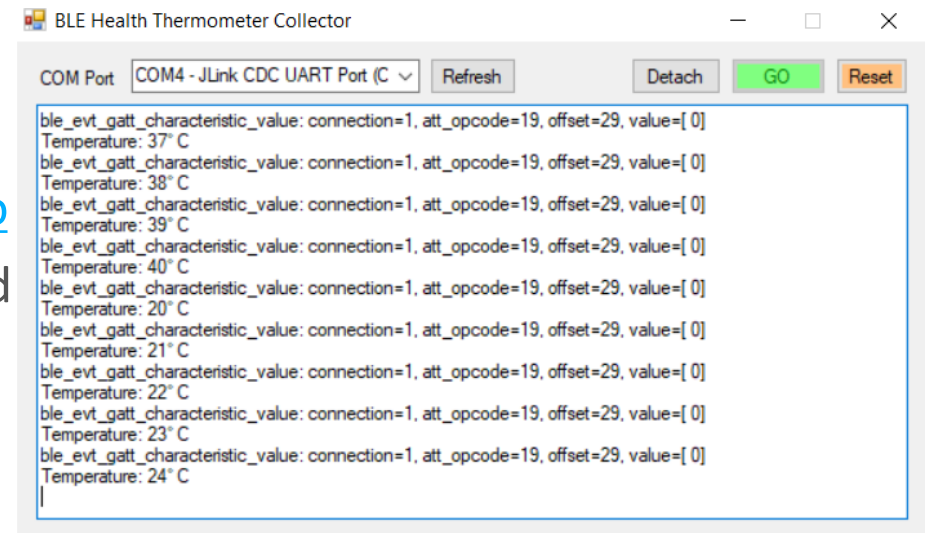
- <https://github.com/jrowberg/bglib/tree/master/BlueGecko>

This repo includes API 2.x Blue Gecko NCP host libraries and example projects for both MS Visual C# and Python 3

- <https://github.com/kryoung-silabs/>

This repo includes a few additional API 2.x BGLIB C library example projects

- **soc-thermometer-client**: Connects to multiple health thermometer server devices and displays the temperature readings
- **BLETest**: A general purpose NCP host RF test utility



```
$ ./exe/thermometer-client /dev/ttyACM0 115200 1
Starting up...
Resetting NCP target...

BLE Central started

ADDR  TEMP  RSSI | ADDR  TEMP  RSSI | ADDR  TEMP
2a9b 29.90C -28dBm|0000  0.00C  0dBm|0000  0.00C
```

Silicon Labs' Bluetooth Module Families



BGM13P



BGM13S



BGM210P



BGM210L



BGM220P (Q3'20)



BGM220S (Q3'20)

	BGM13P	BGM13S	BGM210P	BGM210L	BGM220P (Q3'20)	BGM220S (Q3'20)
Protocols	5.1 and mesh (1M, 2M, Coded PHY and AE)	5.1 and mesh (1M, 2M, Coded PHY and AE)	5.1 and mesh 1.0 (1M, 2M, Coded PHY and AE)	5.1 and mesh 1.0 (1M, 2M, Coded PHY and AE)	5.2 and mesh 1.0 LPN (1M, 2M, Coded PHY, AE and AoA/D)	5.2 and mesh 1.0 LPN (1M, 2M, Coded PHY, AE and AoA/D)
EFR32 SoC	BG13	BG13	BG21	BG21	BG22	BG22
Antenna	Built-in or U.FL	Built-in or RF pin	Built-in or RF pin	Built-in	Built-in	Built-in or RF pin
Max TX power	+8 / +19 dBm	+8 / +18 dBm	+10 / +20 dBm	+12.5 dBm	+8 dBm	+6 dBm
Sensitivity (1M)	-94.8 dBm	-94.1 dBm	-97 dBm	-97 dBm	-98 dBm	-98 dbm
Flash (kB)	512	512	1024	1024	512	512
RAM (kB)	64	64	96	96	32	32
GPIO	25	30	20	12	24,25	25
Operating Voltage	1.8V – 3.6V	1.8V – 3.6V	1.8 – 3.8V	1.8 – 3.8V	1.71V – 3.8V	1.71V – 3.8V
Operating Temp.	-40 to +85C	-40 to +85C	-40 to +125C	-40 to +125C	-40 to +105C	-40 to +105C
Dimensions W x L x H (mm)	13.0 x 15.0 x 2.2	6.5 x 6.5 x 1.4	13.0 x 15.0 x 2.2	13.0 x 15.0 x 2.2	13.0 x 15.0 x 2.2	6 x 6 x 1.3
Certifications	BT, CE, FCC, ISED, Japan, S-Korea and Taiwan	BT, CE, FCC, ISED, Japan & S-Korea	BT, CE, FCC, ISED, Japan & S-Korea	BT, CE, FCC, ISED, Japan & S-Korea	BT, CE, FCC, ISED, Japan & S-Korea	BT, CE, FCC, ISED, Japan & S-Korea

3rd Party Hardware – Darwin Tech Bluetooth 5.0 Dongle



- Agency Certifications
 - US
 - Canada
 - Australia and New Zealand
 - Europe
- +7 dBm
- Bluetooth v5.0 including long range PHY support
- Based on EFR32BG13
- Available through Symmetry Electronics

Blue Gecko NCP Summary

- Highly flexible solution for implementing Bluetooth in a variety of different applications and architectures
- It's easy to get started on Silicon Labs development boards and available software examples
- It's easy to deploy with agency certified modules from both Silicon Labs and third parties



Useful References

- [AN1259: Using the v3.x Silicon Labs Bluetooth® Stack in Network Co-Processor Mode](#)

- [AN1042: Using the v2.x Silicon Labs Bluetooth® Stack in Network Co-Processor Mode](#)

These are general guides with details on getting started with using NCP mode. Also includes information on implementing custom BGAPI.

- [AN1255: Transitioning from the v2.x to the v3.x Bluetooth® SDK](#)

- [KBA_BT_1602: NCP Host Implementation and Example](#)

This example implements an NCP host on an EFM32 device. This is a good guide for implementing NCP hosts on a generic embedded processor (specialized SoC, etc.).

- [Local Event Handling on Bluetooth NCP](#)

docs.silabs.com link explaining how to implement event handling local to the NCP

- <https://github.com/jrowberg/bglib/tree/master/BlueGecko>

- <https://github.com/kryoung-silabs/>

- <https://www.darwintechologiesllc.com/products>

BGTool Demo

The screenshot displays the BGTool application window, which is titled "BGTool" and features the Silicon Labs logo. The interface is divided into several sections:

- Navigation Bar:** Contains tabs for "Bluetooth Smart" (selected), "Security Manager", and "Persistent storage".
- Bluetooth Smart Section:**
 - Generic Access Profile:** Includes a "Show:" dropdown with "Basic settings" selected and "Advanced settings" as an option.
 - Advertise (Slave):** A section with a dropdown arrow, containing "Discoverable mode" (with options: Non-discoverable, Limited discoverable, General discoverable (selected), Broadcast, User data) and "Connectable Mode" (with options: Non-connectable, Connectable-scannable (selected), Scannable-not-connectable, Connectable-not-scannable). "Start" and "Stop" buttons are present.
 - Discover (Master):** A section with a dropdown arrow, containing "Scan PHY" (with options: 1M PHY (selected), Coded PHY) and "Scan type" (with options: Limited, Generic (selected), Observation).
 - Table:** A table with columns: Address, RSSI (dBm), Bonding handle, Advertising data, Scan Response, and a "Connect" button. The table contains one entry:

Address	RSSI (dBm)	Bonding handle	Advertising data	Scan Response	
6f:37:ad:c5:03:b9	-49	255 (=not bonded)	Raw data: 0x0201060aff4c0010050b189d7753		Connect
- Log Section:** A "Log" window with a "Settings" dropdown. It displays a list of scan responses with timestamps and details:
 - 16:25:33,0377: gecko_evt_le_gap_scan_response rssi: -17 (0xffffef) packet_type: 0 (0x00) address:00:0d:6f:20:b2:d6 address_type: 0 (0x00) bonding: 255 (0xff) data:020106060948656c6f
 - 16:25:33,0462: gecko_evt_le_gap_scan_response rssi: -26 (0xffffe6) packet_type: 0 (0x00) address:00:0d:6f:20:b1:e4 address_type: 0 (0x00) bonding: 255 (0xff) data:020106060948656c6f
 - 16:25:33,0478: gecko_evt_le_gap_scan_response rssi: -20 (0xffffec) packet_type: 0 (0x00) address:00:0d:6f:20:b2:d6 address_type: 0 (0x00) bonding: 255 (0xff) data:020106030327181516271853696c6162734465762dd6b2206f0d000000
 - 16:25:33,0514: gecko_evt_le_gap_scan_response rssi: -49 (0xffffcf) packet_type: 0 (0x00) address:6f:
- BGAPI commands:** A section with a text input field and "Send", "Save", and "Clear" buttons.
- Footer:** Includes "Interactive view", "Device Details", and "Connected (tty.SLAB_USBtoUART)" with a status indicator.



works with

BY SILICON LABS

SEPTEMBER 9–10, 2020 | VIRTUAL

workswith.silabs.com

*TWO DAYS OF TECHNICAL TRAINING
FROM BEGINNER TO ADVANCED*