

ECEN 5823-001 / -001B

Internet of Things Embedded Firmware

Lecture #20

2 November 2017

Agenda

- Class announcements
- BLE Health Temp Service Rubric
- Course Project
- Bluetooth DFU OTA
- Bluetooth Mesh

Class Announcements

- No quiz this week
- Course Project Proposal is due this Saturday the 4th at 11:59pm
- BLE assignment Rubric
- #bluetoothmesh slack channel has been established
 - This channel is dedicated for discussions related to Bluetooth Mesh. The professor may take input from this channel for lecture topics.

BLE Health Temp Sensor Rubric

- Functional code delivered per exercise. Max score is 10.0 pts.
 - Verify that Advertising Min/Max Interval is set to 800 1.0 pt
 - If using “magic” numbers, instead of 1.0 pt, 0.5 pts
 - Verify that the Connection Interval Min/Max is set to 60 1.0 pt
 - If using “magic” numbers, instead of 1.0 pt, 0.5 pts
 - Verify that the Slave Latency is set to 4 1.0 pt
 - If using “magic” numbers, instead of 1.0 pt, 0.5 pts
 - `gecko_cmd_le_connection_get_rssi(evt->data.evt_gatt_server_characteristic_status.connection);` in the BLE event:
`gecko_evt_gatt_server_characteristic_status_id:` 1.0 pt
 - if only in another event, then 0.5 pts
 - if setting TX_Power in `gecko_evt_le_connection_rssi_id:` 1.0 pt
 - if setting in another continuous routine 0.5 pts

BLE Health Temp Sensor Rubric

- Functional code delivered per exercise. Max score is 10.0 pts.
 - If TX power is set to -26dbm by using -260 in `gecko_cmd_system_set_tx_power` for RSSI > -35db
0.5 pts
 - If TX power is set to 8dm by using 80 in `gecko_cmd_system_set_tx_power` for RSSI < -85db
0.5 pts
 - While running the code, does a valid temp appear on the phone app 1.0pt
 - With the code providing temperatures to the phone app, and only 1 ft away, is the measured BLE connection event between 9.5 – 11.0 mA
0.5 pts
 - With the code providing temperatures to the phone app, and the phone is placed 15-20 ft away, is the measured BLE connection event between 13.5-16.0 mA
0.5 pts

BLE Health Temp Sensor Rubric

- Functional code delivered per exercise. Max score is 10.0 pts.
 - Does the connection interval measure to 75mS 0.5 pts
 - Does the system slave latency measure to 375mS 0.5 pts
 - Upon reset, using the BLE browser, is the TX power reported as 0 0.5 pts
 - Upon connecting the phone to the dev kit with the phone 1 ft away, disconnect the app. Using the BLE browser, is the TX power reported as 0 0.5 pts



REGISTER NOW

WEBINAR

what makes Bluetooth mesh so disruptive?

The behind-the-scenes story of the making of Bluetooth mesh

Join us for this webinar - 8 November 2017, 9 a.m. - 10 a.m. PT

<http://pages.bluetooth.com/mesh->

[webinar.html?utm_campaign=mesh&utm_source=homepage-cta&utm_medium=homepage-cta&utm_term=homepage-7](http://pages.bluetooth.com/mesh-webinar.html?utm_campaign=mesh&utm_source=homepage-cta&utm_medium=homepage-cta&utm_term=homepage-7)

[hero&utm_content=homepage-hero&_ga=2.230830367.1233636139.1509419350-](http://pages.bluetooth.com/mesh-webinar.html?utm_campaign=mesh&utm_source=homepage-cta&utm_medium=homepage-cta&utm_term=homepage-7hero&utm_content=homepage-hero&_ga=2.230830367.1233636139.1509419350-)

Another Bluetooth Mesh Product idea



Course Project idea





Course Project Clarifications

- You are not required to use the Joy Stick, the BMA280, or the Si7021
 - But you may if you would like
- Projects do not need to be unique, multiple projects can be implementing the same idea
 - The teams can help each other
 - But, the work needs to be individual
 - CU Honor Code
- If you are doing Bluetooth Mesh, you do not necessarily need to do an additional sensor unless you are a team of 3 or 4
- If you are a team of 1 or 2, I have spare kits if you would like to do mesh

Place your sensor request on the following Google Sheet !!!

- <https://docs.google.com/a/colorado.edu/spreadsheets/d/1fNUuaJ-69DEbhiX6ZufN6sRnKK9IEzBg-cTpXQGtF5o/edit?usp=sharing>
- **Orders must be in by this Saturday the 4th at 11:59pm**
- If you are planning on a Bluetooth Mesh project, please include the proper Blue Gecko dev kit radio module



| | |
|----------------------|---|
| Mouser Part #: | 634-SLWRB4104A |
| Manufacturer Part #: | SLWRB4104A |
| Manufacturer: | Silicon Labs |
| Description: | Development Boards & Kits - Wireless EFR32BG13 Radio Brd 2.4GHz 10dBm |
| Lifecycle: |  New Product: New from this manufacturer. |
| |  SLWRB4104A Datasheet |

Foscam Incident

- August 2013

Gilbert says he first heard a voice from down the hall. As he and his wife got closer, what it was saying got worse.

"He said, 'Wake up Allyson, you little (expletive),'"
Gilbert said.

And soon he knew it was coming from the camera.

"I see the camera move on us," Gilbert said.

Gilbert immediately pulled the plug and started doing research. He believes someone hacked his router as well as the camera. The person could see Allyson's name on the bedroom wall to call her by it.




Foscam Incident

- Researchers had discovered that an attacker can use the following IP address of the baby monitor to download the entire memory contents of the baby monitor
 - `http://[IP Address]/proc/kcore`
- Once access to the memory content, the hacker can use a hex editor to obtain:
 - Username
 - Password
- Question is, how did the hacker locate a specific baby monitor that is exposed to the internet?

← → ↻ <https://www.shodan.io> ☆ 🗨️ 🌐 ☰

📱 Apps ★ Bookmarks 📄 IRU 📁 Google Drive 🛡️ Identity Theft Protec 🌐 CU D2L


Shodan Developers Book View All...

 SHODAN 🔍 Explore Enterprise Access Contact Us New to Shodan? Login or Register

The search engine for the Internet of Things

Shodan is the world's first search engine for Internet-connected devices.

Create a Free Account Getting Started



Explore the Internet of Things

Use Shodan to discover which of your devices are connected to the Internet, where they are located and who is using them.



See the Big Picture

Websites are just one part of the Internet. There are power plants, Smart TVs, refrigerators and much more that can be found with Shodan!



Monitor Network Security

Keep track of all the computers on your network that are directly accessible from the Internet. Shodan lets you understand your digital footprint.



Get a Competitive Advantage

Who is using your product? Where are they located? Use Shodan to perform empirical market intelligence.



Foscam Incident

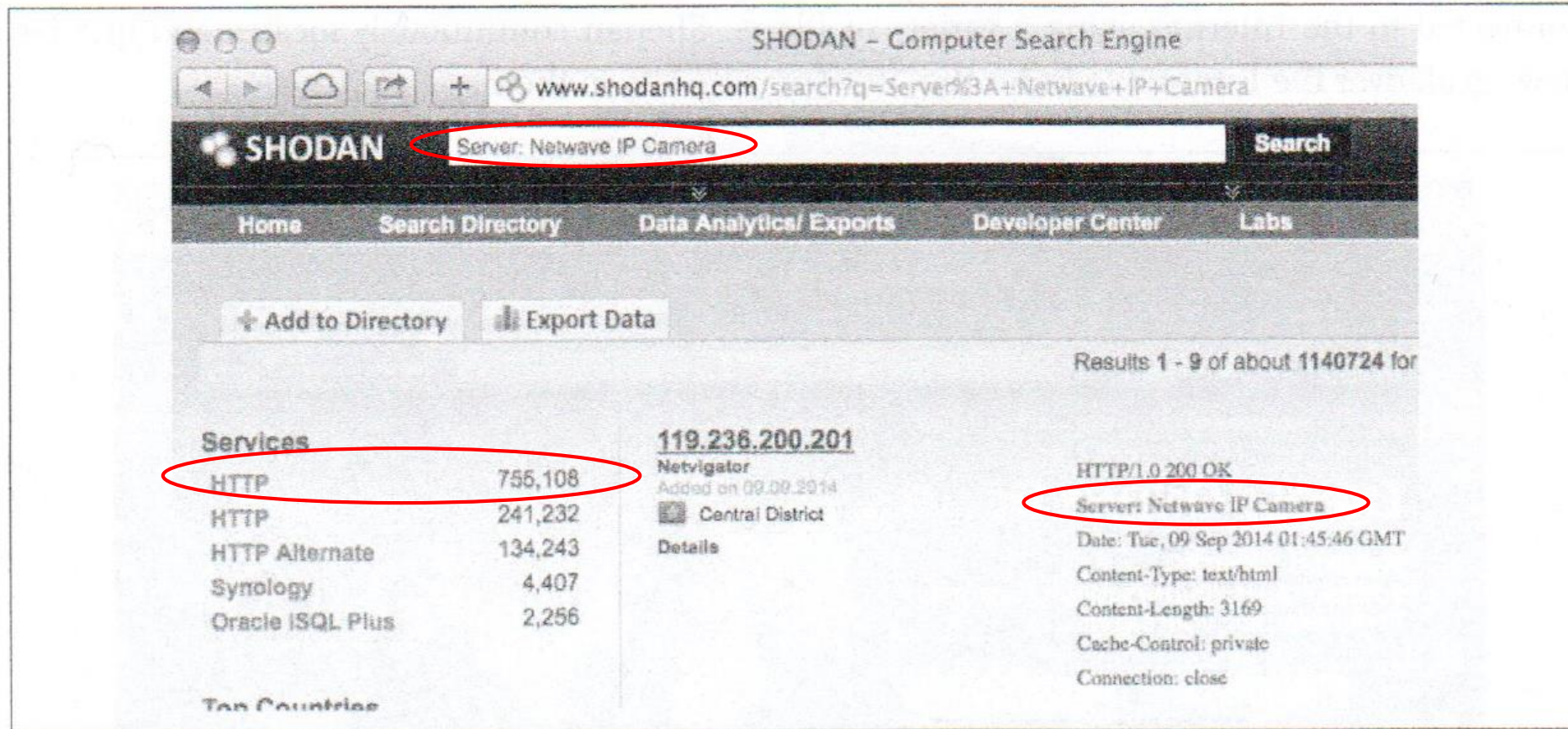


FIGURE 3-3. Shodan query to locate Foscam devices on the Internet

Foscam – Exploiting Default Credentials

- Foscam devices were originally known to have default username of “admin” and a blank password
- Most users are likely to use the default username and password unless the setup requires the user to select a stronger username or password
- In August 2013, Foscam released an upgrade to the firmware and prompted the users to change the user name and blank passwords, but the upgrade was not automatic
- Users had to locate the software update manually and then apply it using a web interface
- Researchers have concluded that 0 Foscam cameras in the “wild” run the latest firmware

Foscam saga continues

- In April 2014, another incident similar to the Gilbert's in August 2013 occurred

CINCINNATI, OH (FOX19) -Heather Schreck was asleep around midnight in her Hebron home when a voice startled her.

"All of a sudden, I heard what sounded like a man's voice but I was asleep so I wasn't sure," Heather said.

Disoriented and confused, Heather picked up her cell phone to check the camera in her 10-month-old daughter Emma's room. The camera was moving, but she wasn't moving it.

"About the time I saw it moving, I also heard a voice again start screaming at my daughter. He was screaming, 'Wake up baby. Wake up baby.' Then just screaming at her trying to wake her up."

That's when Heather's husband, Adam, ran into Emma's room. Adam said the camera then turned from his petrified daughter to point directly at him.




Foscam saga continues

- The April 2014 Foscam incident exemplifies how security vulnerabilities in IoT devices can persist if device manufacturers do not implement a seamless method to push security patches to existing devices
- Research proves a manual procedure required to update a device pretty much guarantees most people are unlikely to do so
 - Few people are likely to make the effort to find and apply security patches

Course Project – 2nd Bonus Opportunity

- Implementing Bluetooth OTA firmware update
- Opportunity is worth **1% towards your final grade!**
- Requirements to get the extra credit:
 - Must demo to one of the three instructing team members by Saturday, November 11th at 11:59pm
 - By being a bonus, you must demo during convenient instructing team availability and not expect availability at 11:58pm on November 11th
 - For distant students, the demo should be arranged over video chat, skype, etc.
 - Instructing team will ask some change to your code that will verify a successful OTA update

How to implement DFU via Bluetooth OTA?

- Why firmware updates?
 - To improve SECURITY!
 - Provide feature upgrades
 - Fix product bugs / issues
 - Update to the latest Bluetooth Stack
- What is updated during an DFU OTA?
 -  • Bootloader
 -  • Bluetooth Stack
 -  • Application

How long does an OTA take?

- It depends, are you updating just the app or both the app and stack?
- (Theoretical) example:
 - The total size of the firmware image is 130 kB (stack and application)
 - The stack size is 115 kB; the application size is 15 kB
 - The average transfer speed when uploading the firmware image is 25 kbit/s
 - Transferring the full firmware image takes $(130 \times 8) / 25 = 42$ seconds
 - The minimal OTA image can be transferred in just $(15 \times 8) / 25 = 5$ seconds
 - The shorter duration also reduces energy consumption

Bluetooth High Level Requirements

Table 1.2. OTA High Level Requirements

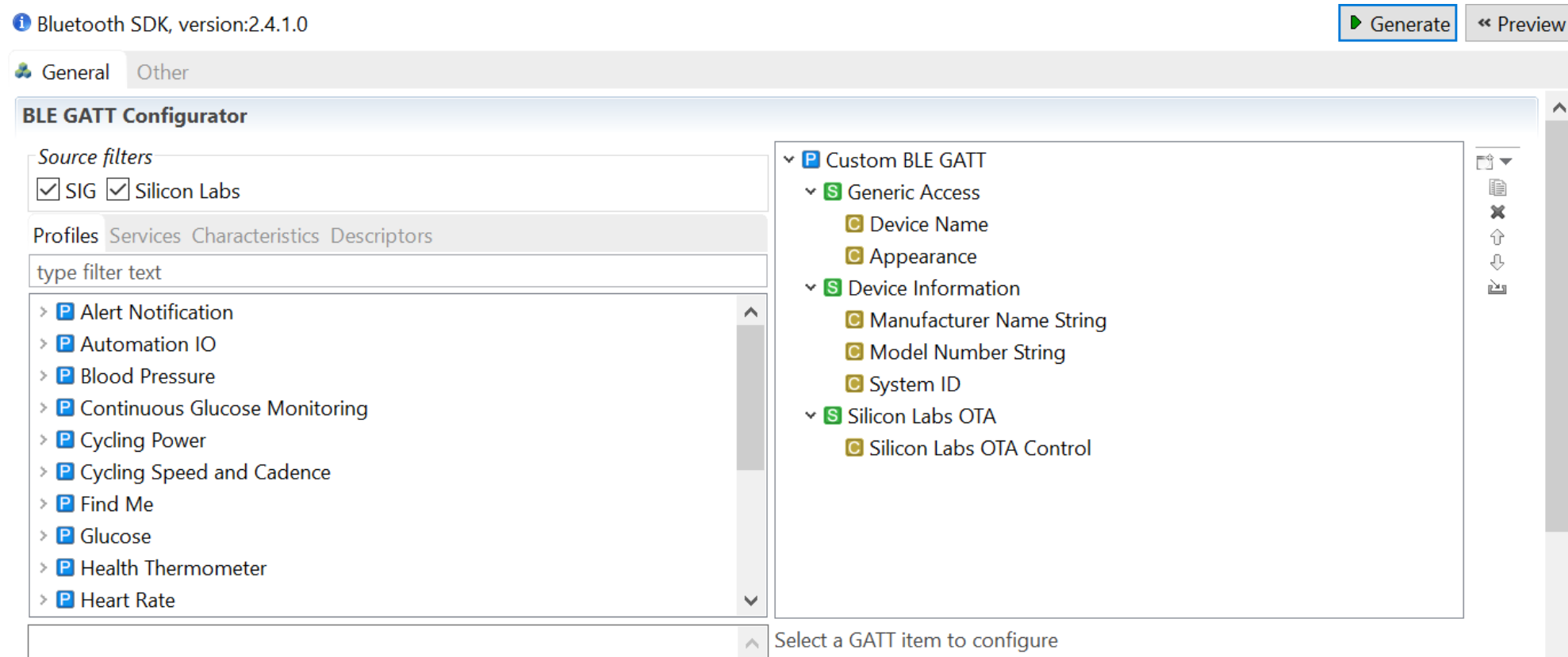
| Category | Requirements |
|-----------------------------|--|
| OTA client | <ul style="list-style-type: none">• Can perform OTA update using normal BLE-enabled smartphone/tablet/PC (no special hardware needed) |
| Robustness | <ul style="list-style-type: none">• Target device can verify that uploaded firmware is valid• Procedure can recover from interrupted OTA due, for example, to sudden power loss or a dropped connection |
| Security | <ul style="list-style-type: none">• Use encrypted update images• Prevent OTA update from unauthorized clients |
| Non-functional requirements | <ul style="list-style-type: none">• Ease of use (from end user viewpoint)• Small overhead in terms of hardware cost / software development effort |

What implements Bluetooth OTA?

- Bluetooth stack
- User Application
- The majority of the OTA is done by the Bluetooth stack, but the user application has a minor roll to enable OTA and to provide access to the Bluetooth stack OTA operation

Application OTA requirements

- The OTA service must be added as a service



Application OTA requirements

- The OTA service must be added as a service

Bluetooth SDK, version:2.4.1.0 Generate

General Other

Name: Silicon Labs OTA Control
 Type: com.silabs.characteristic.ota_control
 UUID: F7BF3564-FB6D-4E53-88A4-5E37E0326063
 Abstract: Silicon Labs OTA Control.
 Value:
 Length: 1 byte
 Variable length: false
 Type: USER

Property requirements:
 Write - Optional - true

Name: Silicon Labs OTA Control
☐ User description

Characteristic settings
☒ ID: ota_control UUID: F7BF3564-FB6D-4E53-88A4-5E37E0326063
 SIG type: com.silabs.characteristic.ota_control

Value settings
 Value: Value type: user
 Length: 1 byte ☐ Variable length

Properties

| Name | Requirement | State | |
|-----------------------------|-------------|-------|--|
| Write | Optional | True | |
| Set properties' information | | | |
| | | | |
| | | | |

Application OTA requirements

```
case gecko_evt_gatt_server_user_write_request_id:
    if (evt->data.evt_gatt_server_user_write_request.characteristic ==
gattdb_ota_control){
    /* set to boot for OTA upon receiving closure of BLE connection */
    boot_to_dfu = true;

    /* acknowledge response back to client's write request */
    gecko_cmd_gatt_server_send_user_write_response(evt-
>data.evt_gatt_server_user_write_request.connection, gattdb_ota_control, bg_err_success);

    /* close connection to reboot into supervisor / dfu mode */
    gecko_cmd_endpoint_close(evt->data.evt_gatt_server_user_write_request.connection);
    }
    break;
```

Application OTA requirements

```
case gecko_evt_le_connection_closed_id:
```

```
    /* Check if need to boot to dfu mode */
```

```
    if (boot_to_dfu) {
```

```
        /* Enter to DFU OTA mode */
```

```
        gecko_cmd_system_reset(2);
```

```
    } else {
```

```
        /* Restart advertising after client has disconnected */
```

```
        gecko_cmd_le_gap_set_mode(le_gap_general_discoverable,  
le_gap_undirected_connectable);
```

```
    }
```

```
    break;
```

Application OTA requirements

```
/* Gecko configuration parameters (see gecko configuration.h) */
static const gecko_configuration_t config = {
    .config_flags = 0,
    .sleep.flags = SLEEP_FLAGS_DEEP_SLEEP_ENABLE,
    .bluetooth.max_connections = MAX_CONNECTIONS,
    .bluetooth.heap = bluetooth_stack_heap,
    .bluetooth.heap_size = sizeof(bluetooth_stack_heap),
    .bluetooth.sleep_clock_accuracy = 100, // ppm
    .gattddb = &bg_gattddb_data,
    .ota.flags = 0,
    .ota.device_name_len = 3,
    .ota.device_name_ptr = "OTA",
#ifdef FEATURE_PTI_SUPPORT
    .pti = &ptiInit,
#endif
};
```

Application OTA requirements

```
typedef struct
{
    uint32_t flags;
    uint8_t  device_name_len;
    char     *device_name_ptr;
}gecko_ota_config_t;
```

```
#define GECKO_OTA_FLAGS_AUTHENTICATED_WRITE    0x200
#define GECKO_OTA_FLAGS_ENCRYPTED_WRITE       0x100
#define GECKO_OTA_FLAGS_BONDED_WRITE         0x400
```

- OTA_FLAGS is set to zero (the default value) defining that there are no access restrictions
- If access control is required then one of the three above values can be assigned to the flags parameter

Bootloader OTA requirements / setup

- **Bluetooth SDK 2.4.01 Gecko SDK Suite 1.1.0 MCU 5.2.1.0**
- For the OTA DFU we need the Internal Storage Bootloader (single image) (EFR32BG12 parts) or the Bluetooth in-place OTA DFU Bootloader (EFR32BG1 parts) configuration.
- Open Simplicity Studio, and select your device in the Devices tab
- Check the Preferred SDK at the top of the main window
- Click New Project button
- Select Gecko Bootloader application type, click Next
- Select the latest installed SDK, click Next
- Select Internal Storage Bootloader (single image) or Bluetooth in-place OTA DFU Bootloader sample application, click Next
- Name your project, click Next
- Check your device and choose toolchain. Click Finish
- Open the Plugins tab
- Click on Bootloader Core, On the right side tick the checkboxes Require signed firmware upgrade files Require encrypted firmware upgrade files Enable secure boot

Bootloader OTA requirements / setup

Launcher - Gecko BootloaderC:\IoT_Fall_2017_Simplicity\bootloader-storage-internal-ble\bootloader-storage-internal-ble.isc - Simplicity Studio™

File Edit Navigate Search Project Run Window Help

Sign In Search

Launcher Simplicity IDE Energy Profiler Debug Network Analyzer Configurator

Device

- J-Link Silicon Labs (440075209)
 - Blue Gecko BGM121 Wireless Starter Kit (SLWSTK6102A)

J-Link Silicon Labs (440075209)

Preferred SDK: Gecko SDK Suite v1.1.1: Bluetooth 2.4.1.0, Flex 1.2.1.0, MCU 5.2.2.0, Micrium OS 5.0.1 Click [here](#) to change the preferred SDK

Debug Mode: MCU [Change](#)

Adapter Firmware Version: 1v2n0h853 Updated adapter firmware available [Install](#)

[New Project](#) [Recent Projects](#)

Getting Started **Documentation** **Compatible Tools** **Resources**

Demos

- Bluetooth SDK 2.4.1.0
 - [Bluetooth SDK Demos](#)

Software Examples

- Bluetooth SDK 2.4.1.0
 - [Bluetooth SDK Examples](#)
- Gecko Bootloader 1.1.1
 - [Gecko Bootloader Examples](#)

SDK Documentation

- 32 bit MCU SDK 5.2.2.0
 - [API References](#)
- Bluetooth SDK 2.4.1.0
 - [API References](#)
 - [Application Notes](#)
 - [Fundamentals](#)
 - [Quick Start Guides](#)
 - [Release Notes](#)
 - [User's Guides](#)

Solutions

[New Solution](#)

Enter product name

- Custom Solution
 - EFR32 Blue Gecko Bluetooth Starter Kit (SLWSTK6020A)
 - Wireless Starter Kit Mainboard (BRD4001A Rev A01)
 - EFR32BG 2.4 GHz 10.5 dBm Radio Board (BRD4100A)
 - EFR32BG1P232F256GM48

Bootloader OTA requirements / setup

Launcher - Gecko BootloaderC:\IoT_Fall_2017_Simplicity\bootloader-storage-internal-ble\bootloader-storage-internal-ble.isc - Simplicity Studio™

File Edit Navigate Search Project Run Window Help

Sign In Search Launcher Simplicity IDE Energy Profiler Debug Network Analyzer Configurator

No Devices

J-Link Silicon Labs (440075209)

Preferred SDK: Gecko SDK Suite v1.1.1: Bluetooth 2.4.1.0, Flex 1.2.1.0, MCU 5.2.2.0, Micrium OS 5.0.1 Click [here](#) to change the preferred SDK.

Debug Mode: MCU [Change](#)
 Adapter Firmware Version: 1v2p0b853 Updated adapter firmware available. [Install](#)

[New Project](#) [Recent Projects](#)

Getting Started Documentation Compatible Tools Resources

Demos

Software Examples

- Gecko Bootloader 1.1.1
 - Gecko Bootloader Examples**
 - BGAPI UART DFU Bootloader**
Standalone Bootloader for the BLE
 - Bluetooth in-place OTA DFU Bootloader**
Application Bootloader for in-place
 - EZSP SPI Bootloader**
Standalone Bootloader for
 - Internal Storage Bootloader (multip**
Application Bootloader for all

SDK Documentation

- Bluetooth SDK 2.4.1.0
 - [API References](#)
 - [Application Notes](#)
 - [Fundamentals](#)
 - [Quick Start Guides](#)
 - [Release Notes](#)
 - [User's Guides](#)

Solutions

[New Solution](#)

Enter product name

- Custom Solution
 - EFR32 Blue Gecko Bluetooth Starter Kit (SLWSTK6020A)
 - Wireless Starter Kit Mainboard (BRD4001A Rev A01)
 - EFR32BG 2.4 GHz 10.5 dBm Radio Board (BRD4100A)
 - EFR32BG1P232F256GM48

Bootloader OTA requirements / setup

File Edit Navigate Search Project Run Window Help

Project Explorer

- > efr32mg12p332f1024gl125
- > efr32mg12p432f1024gl125
- > efr32mg12p433f1024gl125
- > emlib
- > GNU ARM v4.9.3 - Default
- > internal_flash
- > mbedtls
- > parser
- > storage-common
- > token-management
- > bootloader-callbacks.c
- > bootloader-callbacks.h
- > bootloader-callback-stubs.c
- > bootloader-configuration.h
- > bootloader-slot-configuration.h
- app-encrypt-key.txt
- app-sign-key.pem
- app-sign-key.pem-tokens.txt
- app-sign-key.pem.pub
- bootloader-storage-internal-ble_postbuild.sh
- bootloader-storage-internal-ble.isc
- > soc-empty [GNU ARM v4.9.3 - Default] [EFR32BG1B232F]

Gecko Bootloader, version:1.4.0

General Plugins Storage Callbacks Other

Plugin configuration

Use this section to select or unselect the plugins that you want to use in your application

- ☐ Communication
 - ☐ BGAPI UART DFU
 - ☐ EZSP-SPI
 - ☐ UART XMODEM
 - ☐ XMODEM Parser, provides API: xmodemParser
- ☒ Core
 - ☒ Bootloader Core, provides API: core
 - ☐ GBL Compression (LZ4)
 - ☒ Image Parser, provides API: imageParser
 - ☐ Image Parser with legacy EBL support, provides API: ir
- ☐ Drivers
 - ☐ Delay, provides API: delayDriver
 - ☐ SPI Master, provides API: spiDriver
 - ☐ SPI Slave, provides API: spiSlaveDriver

Plugin: Bootloader Core

Quality: Unknown plugin quality

Description:

Core library for bootloader

Options: [Reset to def](#)

- ☒ Require signed firmware upgrade files
- ☒ Require encrypted firmware upgrade files
- ☐ Enable secure boot
- ☐ Prevent bootloader write/erase

Details (double-click on files to show content):

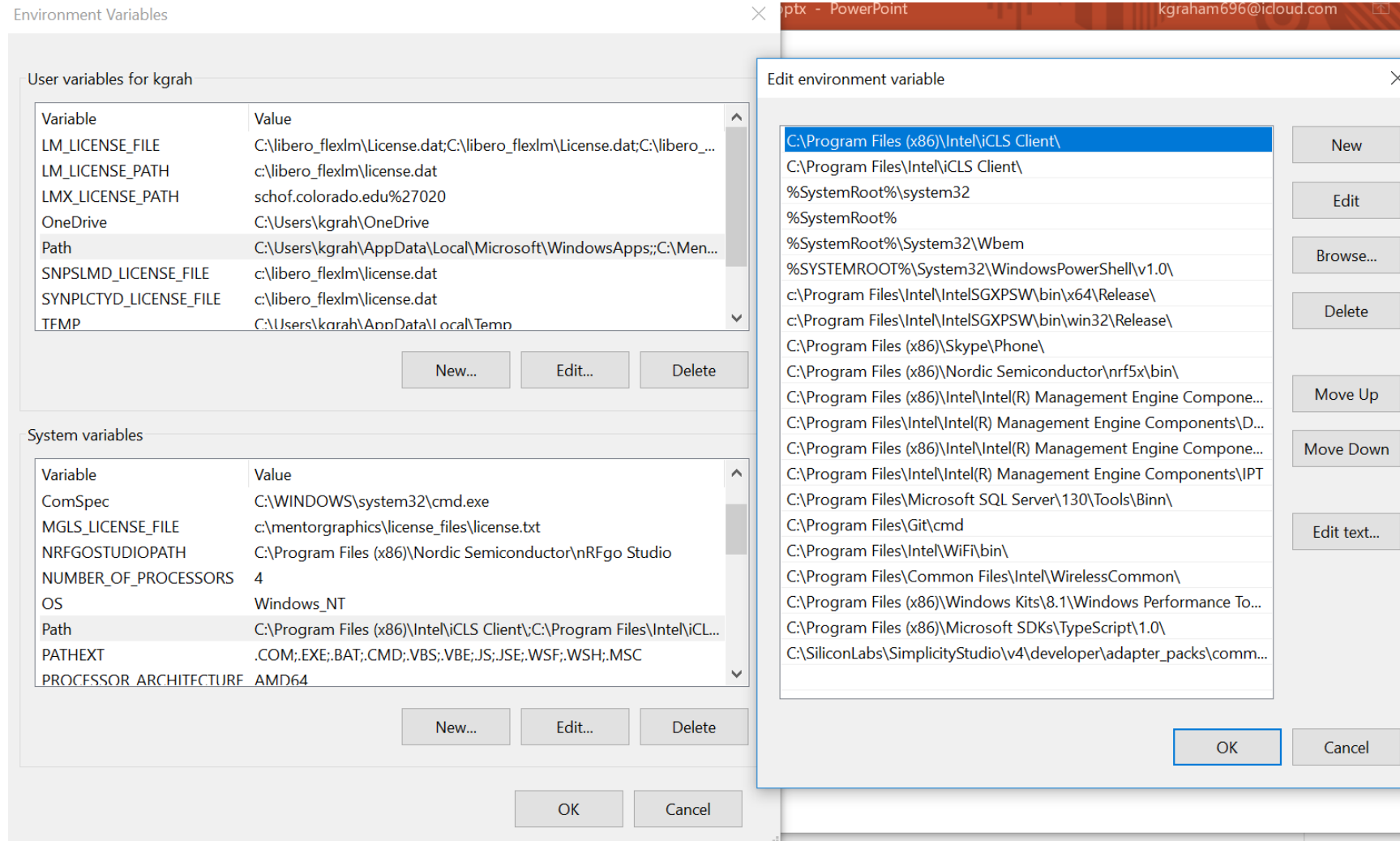
Bootloader OTA requirements / setup

- **Bluetooth SDK 2.4.01 Gecko SDK Suite 1.1.0 MCU 5.2.1.0**
- For the OTA DFU we need the Internal Storage Bootloader (single image) (EFR32BG12 parts) or the Bluetooth in-place OTA DFU Bootloader (EFR32BG1 parts) configuration.
- Open Simplicity Studio, and select your device in the Devices tab
- Check the Preferred SDK at the top of the main window
- Click New Project button
- Select Gecko Bootloader application type, click Next
- Select the latest installed SDK, click Next
- Select Internal Storage Bootloader (single image) or Bluetooth in-place OTA DFU Bootloader sample application, click Next
- Name your project, click Next
- Check your device and choose toolchain. Click Finish
- Open the Plugins tab
- Click on Bootloader Core, On the right side tick the checkboxes Require signed firmware upgrade files Require encrypted firmware upgrade files Enable secure boot
- Click Generate in the upper right corner
- Build your bootloader project.

Bootloader OTA requirements / setup

- **Generate and flash security keys.** The private key is used to generate the public key. All applications developed to be used with this bootloader must be signed using the private key. [Before going further add \SiliconLabs\SimplicityStudio\v4\developer\adapterpacks\commander\commander.exe to system environment variables](#)

Bootloader OTA requirements / setup




Bootloader OTA requirements / setup

- **Generate and flash security keys.** The private key is used to generate the public key. All applications developed to be used with this bootloader must be signed using the private key. Before going further add
`\SiliconLabs\SimplicityStudio\v4\developer\adapterpacks\commander\commander.exe` to system environment variables
- Run the following commands
- `$ commander gbl keygen --type ecc-p256 --outfile app-sign-key.pem`
- `$ commander gbl keygen --type aes-ccm --outfile app-encrypt-key.txt`
- `$ commander flash --tokengroup znet --tokenfile app-encrypt-key.txt --tokenfile app-sign-key.pem-tokens.txt --device EFR32BG1B232F256GM56`

Bootloader OTA requirements / setup

- **Create Bluetooth app with secure OTA DFU capability**
- Since OTA DFU is not fully implemented in the Bootloader, a Bluetooth application has to be created and flashed to the device first along with the Bootloader to support the upgrade. This can be any app that supports restarting the device in DFU mode. The easiest is to use the SoC-Empty software example.
- Open Simplicity Studio, and select your device in the Devices tab
- Check the Preferred SDK at the top of the main window
- Click New Project button
- Select Bluetooth SDK, click Next
- Select the latest installed SDK, click Next
- Select SOC - Empty sample application, click Next
- Name your project, click Next
- Check your device and choose toolchain. Click Finish
- Build you project
- Copy app-sign-key.pem and app-encrypt-key.txt into the Bluetooth project directory.
- Run create_bl_files.bat found in your Bluetooth project

Bootloader OTA requirements / setup

- Copy the bootloader image that ends with combined.s37 from the output folder(GNU ARM) of your bootloader project to the output_gbl folder of your Bluetooth project
- Run the following command in the output_gbl folder:
- `$ commander convert bootloader-storage-internal-ble-combined.s37 stack-signed.gbl app-signed.gbl --outfile bootloader+stack+app.hex`
- Open Commander and flash the hex file to the device. ****Do not erase the flash before flashing the hex file as it will erase the tokens flashed previously

Bootloader OTA requirements / setup

- **Create Bluetooth app to be uploaded via OTA**
- Build your project (iBeacon or Thermometer Example)
- Copy app-sign-key.pem and app-encrypt-key.txt into the Bluetooth project folder of this project
- Run create_bl_files.bat found in your Bluetooth project folder
- This will create stack-signed-encrypted.gbl and app-signed-encrypted.gbl files into the output_gbl folder. These are the signed and encrypted upgrade files, which can be sent OTA to the target device

Bootloader OTA requirements / setup

- **Performing the OTA**
- **(Android)** Copy the .gbl files to your phone and store it in /SiliconLabs_BGApp/OTAFiles/*(ProjectFolder)*/
(iPhone) Copy the .gbl file to your Google Drive
- Download the Blue Gecko App on your phone
- Go to Bluetooth Browser and Connect to your device
- After connecting select OTA in the menu in the top right corner.
- Select the OTA files and do a full or app only OTA depending on the changes in the project

BLE OTA demo

The screenshot displays the SImplicity IDE interface. The top menu bar includes File, Edit, Navigate, Search, Project, Run, Window, and Help. The toolbar contains various icons for file operations and development tools. The Project Explorer on the left shows a project structure with folders like Binaries, Includes, bgapi, device, emlib, GNU ARM v4.9.3 - Default, inc, kit_flashpwr, linker, output_ebl, output_gbl, and src, which contains files like adc.c, adc.h, and bluetooth.c. The main workspace shows the 'Bluetooth SDK, version:2.4.1.0' and the 'BLE GATT Configurator' window. The configurator has tabs for General, Profiles, Services, Characteristics, and Descriptors. The 'General' tab is active, showing 'Source filters' with 'SIG' and 'Silicon Labs' checked. Below this is a list of services including Alert Notification, Automation IO, Blood Pressure, Continuous Glucose Monitoring, Cycling Power, Cycling Speed and Cadence, Find Me, Glucose, Health Thermometer, and Heart Rate. On the right, a tree view shows the 'Custom BLE GATT' configuration with expandable sections for Generic Access, Device Information, Silicon Labs OTA, Tx Power, Health Thermometer, and Client Characteristic Configuration. A 'Generate' button and a 'Preview' button are at the top right of the configurator.

File Edit Navigate Search Project Run Window Help

Launcher SImplicity IDE Energy Profiler Debug Network Analyzer Configurator

Project Explorer

- Assignment_1 [GNU ARM v4.9.3 - Default] [EFR32BG1B23]
- Assignment_2 [GNU ARM v4.9.3 - Default] [EFR32BG1B23]
- Assignment_3 [GNU ARM v4.9.3 - Default] [EFR32BG1B23]
- Assignment_4 [GNU ARM v4.9.3 - Default] [EFR32BG1B23]
- Assignment_5 [GNU ARM v4.9.3 - Default] [EFR32BG1B23]
- Assignment_6 [GNU ARM v4.9.3 - Default] [EFR32BG1B23]
- Assignment_OTA [GNU ARM v4.9.3 - Default] [EFR32BG1B23]
- Binaries
- Includes
- bgapi
- device
- emlib
- GNU ARM v4.9.3 - Default
- inc
- kit_flashpwr
- linker
- output_ebl
- output_gbl
- src
 - adc.c
 - adc.h
 - bluetooth.c

Bluetooth SDK, version:2.4.1.0

Generate Preview

General Other

BLE GATT Configurator

Source filters

☒ SIG ☒ Silicon Labs

Profiles Services Characteristics Descriptors

type filter text

- Alert Notification
- Automation IO
- Blood Pressure
- Continuous Glucose Monitoring
- Cycling Power
- Cycling Speed and Cadence
- Find Me
- Glucose
- Health Thermometer
- Heart Rate

Custom BLE GATT

- Generic Access
 - Device Name
 - Appearance
- Device Information
 - Manufacturer Name String
 - Model Number String
 - System ID
- Silicon Labs OTA
 - Silicon Labs OTA Control
- Tx Power
 - Tx Power Level
- Health Thermometer
 - Temperature Measurement
- Client Characteristic Configuration

Select a GATT item to configure

Silicon Labs' OTA resources

- To learn more about setting up your Application for OTA:
 - <https://www.silabs.com/documents/login/application-notes/an1045-bt-ota-dfu.pdf>
- To learn more about Gecko Bootloader, please refer to these documents:
 - <https://www.silabs.com/documents/public/user-guides/ug266-gecko-bootloader-user-guide.pdf>
 - https://www.silabs.com/community/wireless/bluetooth/knowledge-base.entry.html/2017/06/22/secure_ota_dfu-Wb22
 - https://www.silabs.com/community/wireless/bluetooth/knowledge-base.entry.html/2017/04/12/adding_gecko_bootloa-osqt