Firmware Options and Upgrade Methods User Guide

Lyra (22) and Lyra 24 Series

Version 3.1

Revision History

Version	Date	Notes	Contributors	Approver
1.0	3 May 2022	Initial Release	Florian Baumgartl	Jonathan Kaye
1.1	18 July 2022	Updated all tables. Added details for BOOT pin and SWCLK/SWDIO usage.	Florian Baumgartl, Greg Leach, Raj Khatri	Jonathan Kaye
1.2	6 Sept 2022	Added details for the firmware image files generated with the by Simplicity Studio and how these related to Ezurio provided firmware image files. Added References section and revised all tables. Updated examples and screenshots. Replaced JTAG term with SWD and fixed URLs. Added further instructions and updated wording.	Florian Baumgartl, Rikki Horrigan, Chris Laplante, Greg Leach	Jonathan Kaye
1.3	19 Sept 2022	Added extra Bootloader section and further details about the original (legacy) Wireless Xpress bootloader.	Florian Baumgartl	Jonathan Kaye
2.0	02 Aug 2023	Introduced the Lyra 24 Series. Revised instructions and added further examples.	Florian Baumgartl	Jonathan Kaye
3.0	7 May 2024	Ezurio rebranding	Sue White	Dave Drogowski
3.1	16 Dec 2024	EFR Connect has been replaced with Simplicity Connect (Si Connect) mobile app, and updated its screenshots.	Florian Baumgartl	Dave Drogowski

Contents

1 Introduction 4
2 Requirements 4
3 Lyra (22) and Lyra 24 GA Software Release 5
4 Available Bootloader and Firmware Options 6
4.1 Overview 6
4.2 Simplicity Studio Generated Firmware Files 9
4.2.1 For the SWD Firmware Upgrade Method Only 9
4.2.2 For the Serial DFU (UART) Firmware Upgrade Method Only 9
4.2.3 For the BLE FOTA (EFR Connect mobile app) Firmware Upgrade Method Only 9
4.3 Ezurio Provided Firmware Files 9
5 Hardware Preparation 14
6 Firmware Upgrade Methods 15
6.1 BLE (FOTA) using EFR Connect Mobile App 15
6.1.1 Lyra (22) Example – EFR Connect Mobile App under Android 16
6.1.2 Lyra 24 Example – EFR Connect Mobile App under iOS 20
6.2 Serial DFU (UART) using uart_dfu application 24
6.2.1 Lyra (22) Example – uart_dfu.exe with User Commands and Output 27
6.2.2 Lyra 24 Example – uart_dfu.exe with User Commands and Output 27
6.3 Simplicity Commander & Simplicity Studio 5 (SWD) 28
6.3.1 Lyra (22) Example – Simplicity Commander (Log Window) Output 31
6.3.2 Lyra 24 Example – Simplicity Commander (Log Window) Output 32
6.4 SEGGER J-Link Commander (SWD) 33
6.4.1 Lyra (22) Example – J-Link Commander with User Commands and Output 38
6.4.2 Lyra 24 Example – J-Link Commander with User Commands and Output 40
7 Further Information 41
8 References 42
9 Additional Information 43

1 Introduction

This firmware update user guide shows and explains all available firmware options and upgrade methods for our Lyra (22) and Lyra 24 Series modules. For demonstration purposes one Lyra (22) and 24 Series development board will be flashed and programmed via:

- **BLE (FOTA) using Simplicity Connect (Si Connect) Mobile App** *Recommended for end users and customers.*
- **Serial DFU (UART) using uart_dfu Application** *Recommended for power users and production use.*
- **Simplicity Commander & Simplicity Studio 5 (SWD)** *Recommended for developer and production use.*
- **SEGGER J-Link Commander (SWD)** *Recommended for developer and production use.*

Note(s) This document does not cover how to implement / utilize the Wireless Xpress and AT Command Set functions – or any C code development guidance. For more information, refer to the induvial User Guide documents available under http://www.ezurio.com/lyra-series (Lyra 22) and http://www.ezurio.com/lyra24-series (Lyra 24).

---

2 Requirements

This firmware update user guide is applicable to the following part number(s):

Lyra 22/24 Development Kits

- (453-00090-K1) Lyra (22) Series – PCB (+8 dBm) development board with integrated antenna option,
- (453-00091-K1) Lyra (22) Series – SIP (+6 dBm) development board with various antenna options.

- (453-00170-K1) Lyra 24 Series – SIP (+10 dBm) development board with various antenna options,
- (453-00142-K1) Lyra 24 Series – PCB (+10 dBm) development board with integrated antenna option,
- (453-00145-K1) Lyra 24 Series – PCB (+20 dBm) development board with integrated antenna option,
- (453-00148-K1) Lyra 24 Series – PCB (+20 dBm) development board with RF trace pad option.

Lyra 22/24 Modules & USB Dongles

- (453-00090) Lyra (22) Series – PCB (+8 dBm) module with integrated antenna option,
- (453-00091) Lyra (22) Series – SIP (+6 dBm) module with integrated antenna option.

- (453-00170) Lyra 24 Series – SIP (+10 dBm) module with various antenna options,
- (453-00142) Lyra 24 Series – PCB (+10 dBm) module with integrated antenna option,
- (453-00145) Lyra 24 Series – PCB (+20 dBm) module with integrated antenna option.
- (453-00148) Lyra 24 Series – PCB (+20 dBm) module with RF trace pad option.

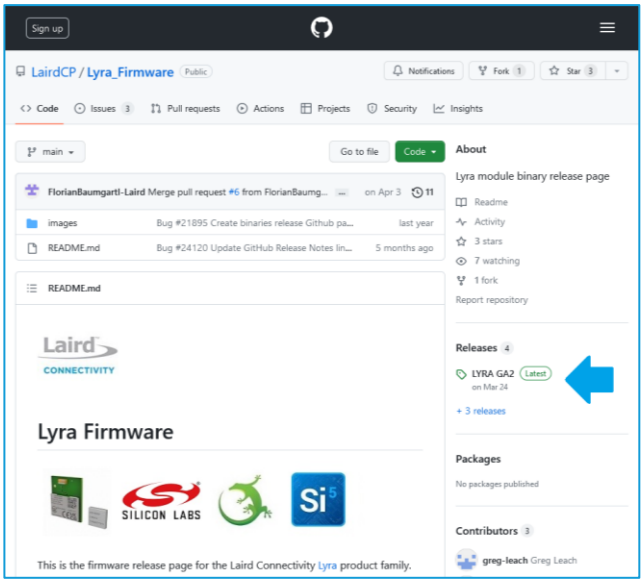
Note: The Lyra 24 Series USB dongle (450-00184) is based on the +20 dBm PCB module with part number 453-00145. In this case, please review and refer to the Lyra 24 USB Dongle – User Guide document for more information.

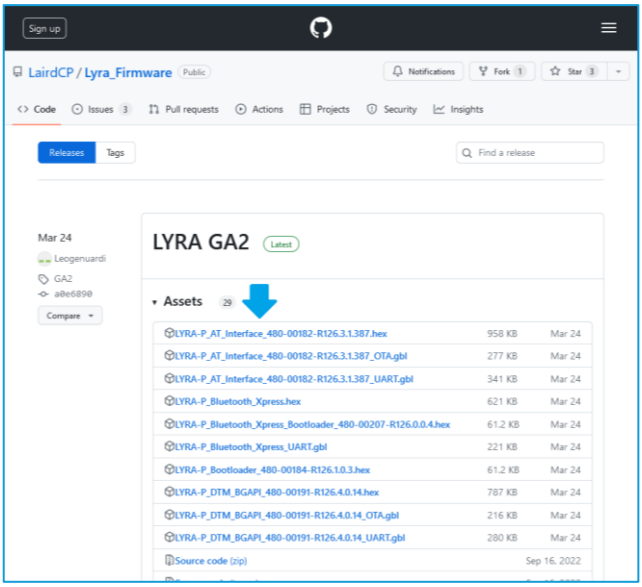
Software Tools & Utilities

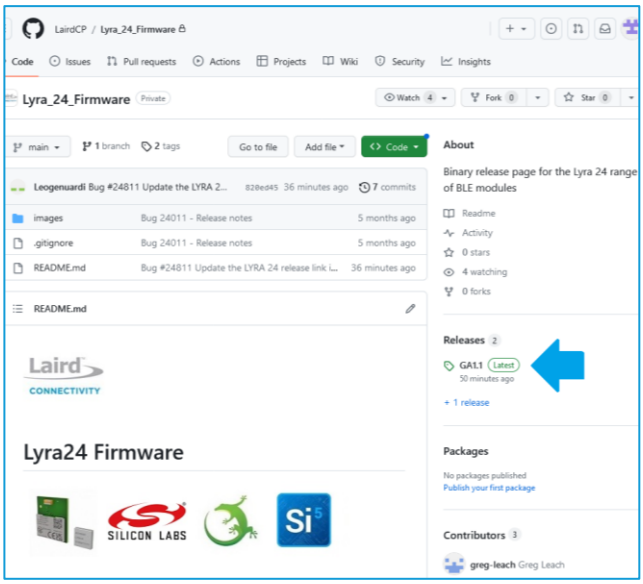
1. uart_dfu utility software, available for Windows. You can use this release version for both Lyra (22) and Lyra 24.
2. (Optional) Simplicity Studio 5 for Windows, Linux or Mac → https://www.silabs.com/developers/simplicity-studio.
3. J-Link Software and Documentation Pack v7.88n or later → https://www.segger.com/downloads/J-Link/.
4. iOS or Android Smartphone with Bluetooth Low Energy support running latest OS version is preferred.
5. Simplicity Commander v1.15.3 or later. Available for Windows, Linux, and Mac.
6. Simplicity Connect (Si Connect) Mobile App for iOS / Android v2.7.1 or later.

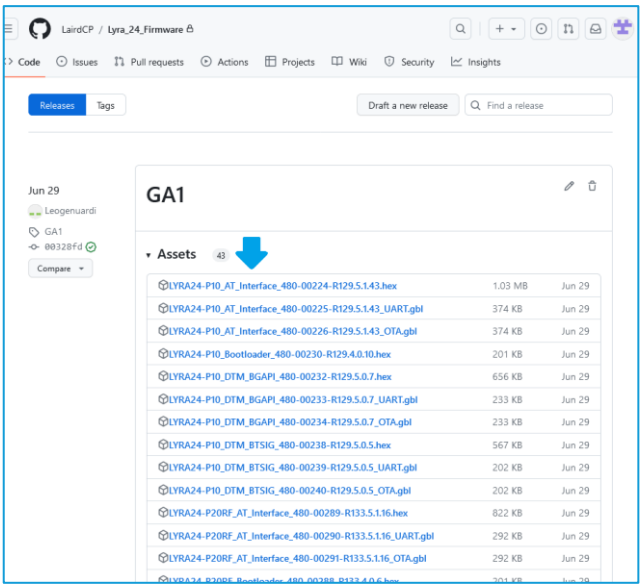
3 Lyra (22) and Lyra 24 GA Software Release

You can obtain the latest available firmware images at https://github.com/LairdCP/Lyra_Firmware (Lyra 22) and https://github.com/LairdCP/Lyra_24_Firmware (Lyra 24). Select the latest available release as shown below or click on **Release** to see a list of all published firmware image releases. To download a specific firmware release, tap **› Assets** and download the individual firmware image files for Lyra.









Note: Older firmware images are unsupported by Ezurio and may contain issues or bugs that have been fixed in newer versions. Ezurio strongly recommends always using the latest firmware images. Use of these unsupported firmware images is at your own risk.

4 Available Bootloader and Firmware Options

4.1 Overview

The bootloader is the first piece of code which gets executed on modules reset and is required for operation of all other applications. It should be the first software part programmed to the target device.

All Lyra DVKs ship completely empty and erased, with no bootloader and no application whereas all Lyra modules leave our production preprogrammed with a bootloader and factory BLE application. Please make sure to first program a bootloader and/or application image on your Lyra DVK / module before further use.

Note that binaries for the bootloader are available for transfer using SWD only. By default, all Lyra modules come with the Ezurio bootloader type preprogrammed as described in Table 1 and Table 2.

Lyra (22) only For compatibility reasons, Ezurio also provides the original (legacy) BGX220 Series Wireless Xpress bootloader on GitHub for usage with legacy Bluetooth Xpress hardware designs. The only difference is in the location of the BOOT pin defined by the bootloader configuration.

It is also possible to create a custom Lyra bootloader project using the Simplicity Studio 5 IDE and the "Internal Storage Bootloader (single image on 512kB device)" example. This e.g. allows you to modify the BOOT pin definition or to remove the GPIO activation for the BOOT pin if needed.

Table 1: Available Bootloaders for the Lyra 22 Series

Bootloader Type and Config	Lyra (22) P BOOT Pin	Lyra (22) S BOOT Pin
Ezurio	PC07	PA06
Bluetooth Xpress	PD02	PD02

Table 2: Available Bootloaders for the Lyra 24 Series

Bootloader Type and Config	Lyra 24 P10 BOOT Pin	Lyra 24 S10 BOOT Pin	Lyra 24 P20 BOOT Pin	Lyra 24 P20 RF BOOT Pin
Ezurio	PC07	PC07	PC07	PC07

The following three tables illustrate and summarize how you can swap between the individually firmware options using the different firmware upgrade methods which are supported by the Lyra 22 and Lyra 24 Series:

Firmware Update Source (From): The *old and current* firmware option which is at this point running on the Lyra module(s).

Firmware Update Target (To): The *new and desired* firmware option which you would like to update the Lyra module(s) to.

Table 3: Lyra (22) – Available Firmware Options for SWD.

SWD		Firmware Update Source (From)				
Firmware Update Target (To)		Factory Production	DTM Certification	Wireless Xpress[1]	AT Command Set Interface	Custom C Application
	DTM	SUPPORTED	IN-PLACE UPDATE(S)	SUPPORTED	SUPPORTED	SUPPORTED
	Wireless Xpress[1]	SUPPORTED	SUPPORTED	IN-PLACE UPDATE(S)	SUPPORTED	SUPPORTED
	AT Command Set Interface	SUPPORTED	SUPPORTED	SUPPORTED	IN-PLACE UPDATE(S)	SUPPORTED
	Custom C Application	SUPPORTED	SUPPORTED	SUPPORTED	SUPPORTED	IN-PLACE UPDATE(S)

Table 4: Lyra (22) – Available Firmware Options for Serial (DFU) UART.

(UART)		Firmware Update Source (From)				
Firmware Update Target (To)		Factory Production	DTM Certification	Wireless Xpress[1]	AT Command Set Interface	Custom C Application
	DTM	SUPPORTED	IN-PLACE UPDATE(S)	SUPPORTED	SUPPORTED	SUPPORTED
	Wireless Xpress[1]	SUPPORTED	SUPPORTED	IN-PLACE UPDATE(S)	SUPPORTED	SUPPORTED
	AT Command Set Interface	SUPPORTED	SUPPORTED	SUPPORTED	IN-PLACE UPDATE(S)	SUPPORTED
	Custom C Application	SUPPORTED	SUPPORTED	SUPPORTED	SUPPORTED	IN-PLACE UPDATE(S)

Table 5: Lyra (22) – Available Firmware Options for Firmware Over-The-Air (FOTA).

F(OTA)		Firmware Update Source (From)				
Firmware Update Target (To)		Factory Production	DTM Certification	Wireless Xpress[1]	AT Command Set Interface	Custom C Application
	DTM	SUPPORTED	N/A	UNSUPPORTED	SUPPORTED	SUPPORTED
	Wireless Xpress[1]	UNSUPPORTED	N/A	IN-PLACE UPDATE(S)	UNSUPPORTED	UNSUPPORTED
	AT Command Set Interface	SUPPORTED	N/A	UNSUPPORTED	IN-PLACE UPDATE(S)	SUPPORTED
	Custom C Application	SUPPORTED	N/A	UNSUPPORTED	SUPPORTED	IN-PLACE UPDATE(S)

Table 6: Lyra 24 – Available Firmware Options for SWD.

SWD		Firmware Update Source (From)				
Firmware Update Target (To)		Factory Production	DTM Certification	Wireless Xpress[1]	AT Command Set Interface	Custom C Application
	DTM	SUPPORTED	IN-PLACE UPDATE(S)	UNSUPPORTED	SUPPORTED	SUPPORTED
	Wireless Xpress[1]	UNSUPPORTED	UNSUPPORTED	UNSUPPORTED	UNSUPPORTED	UNSUPPORTED
	AT Command Set Interface	SUPPORTED	SUPPORTED	UNSUPPORTED	IN-PLACE UPDATE(S)	SUPPORTED
	Custom C Application	SUPPORTED	SUPPORTED	UNSUPPORTED	SUPPORTED	IN-PLACE UPDATE(S)

Table 7: Lyra 24 – Available Firmware Options for Serial (DFU) UART.

(UART)		Firmware Update Source (From)				
Firmware Update Target (To)		Factory Production	DTM Certification	Wireless Xpress[1]	AT Command Set Interface	Custom C Application
	DTM	SUPPORTED	IN-PLACE UPDATE(S)	UNSUPPORTED	SUPPORTED	SUPPORTED
	Wireless Xpress[1]	UNSUPPORTED	UNSUPPORTED	UNSUPPORTED	UNSUPPORTED	UNSUPPORTED
	AT Command Set Interface	SUPPORTED	SUPPORTED	UNSUPPORTED	IN-PLACE UPDATE(S)	SUPPORTED
	Custom C Application	SUPPORTED	SUPPORTED	UNSUPPORTED	SUPPORTED	IN-PLACE UPDATE(S)

Table 8: Lyra 24 – Available Firmware Options for Firmware Over-The-Air (FOTA).

F(OTA)		Firmware Update Source (From)				
		Factory	DTM	Wireless	AT Command	Custom C

F(OTA)		Firmware Update Source (From)				
Firmware Update Target (To)		Production	Certification	Xpress[1]	Set Interface	Application
	DTM	SUPPORTED	N/A	UNSUPPORTED	SUPPORTED	SUPPORTED
	Wireless Xpress[1]	UNSUPPORTED	N/A	UNSUPPORTED	UNSUPPORTED	UNSUPPORTED
	AT Command Set Interface	SUPPORTED	N/A	UNSUPPORTED	IN-PLACE UPDATE(S)	SUPPORTED
	Custom C Application	SUPPORTED	N/A	UNSUPPORTED	SUPPORTED	IN-PLACE UPDATE(S)

The following sections describe the firmware file types that can be generated via Silabs' Simplicity Studio, and how these relate to the firmware files provided by Ezurio via the Lyra (22) [C] and Lyra 24 [C] firmware images on GitHub.
Images are grouped by release number in the Releases subsection of GitHub for Lyra (22) [B] and Lyra 24 [D].

Note: [1] Wireless Xpress is not supported with the Lyra 24 Series and can only run on the Lyra (22) Series.

4.2 Simplicity Studio Generated Firmware Files

Please find below an overview about all firmware images which can be generated via Simplicity Studio.

4.2.1 For the SWD Firmware Upgrade Method Only

[filename].hex → This is an application image in hex format. Can be used with the SWD firmware upgrade method only.

Other formats such as [filename].bin and [filename].s37 are currently not covered in this Lyra (22) and Lyra 24 Series Firmware Options and Upgrading Methods User Guide document.

4.2.2 For the Serial DFU (UART) Firmware Upgrade Method Only

[filename]_full.gbl → Combined application and Silicon Labs Apploader image, can be used with the Serial DFU (UART) firmware upgrade method only.

[filename]_full-crc.gbl → Combined version with included CRC-32 checksum of the application image and Silicon Labs Apploader image, can be used with the Serial DFU (UART) firmware upgrade method only.

[filename].gbl → Application image, can be used with both Serial DFU (UART) and BLE FOTA (EFR Connect mobile app) firmware upgrade methods.

[filename] -crc.gbl → Application image with included CRC-32 checksum, can be used with both Serial DFU (UART) and BLE FOTA (EFR Connect mobile app) firmware upgrade methods.

apploader.gbl → Silabs Apploader image, can be used with the Serial DFU (UART) firmware upgrade method only.

apploader-crc.gbl → Silabs Apploader image with included CRC-32 checksum, can be used with the Serial DFU (UART) firmware upgrade method only.

4.2.3 For the BLE FOTA (EFR Connect mobile app) Firmware Upgrade Method Only

[filename].gbl → Application image, can be used with both BLE FOTA (EFR Connect mobile app) and Serial DFU (UART) firmware upgrade methods.

[filename]-crc.gbl → Application image with included CRC-32 checksum, can be used with both BLE FOTA (EFR Connect mobile app) and Serial DFU (UART) firmware upgrade methods.

Note: Updating the Apploader image via the BLE FOTA (EFR Connect mobile app) firmware upgrade method is NOT supported, so please make sure to select and perform a partial (not full) software update when using the EFR Connect mobile app as shown in section 6.1 BLE (FOTA) using EFR Connect Mobile App

---

4.3 Ezurio Provided Firmware Files

Please find below an overview about all firmware images which are provided by Ezurio. Files originating from Ezurio are suffixed with a code of the format **V-W.X.Y.Z**, with the parts having the following meaning.

- V: This is the Ezurio part number for the firmware image
- W: This is the module variant identifying code for the Lyra hardware,
- X: This is the major revision number of the Silicon Labs SDK in use
- Y: This is the major revision number of the firmware image
- Z: This is the minor revision number of the firmware image

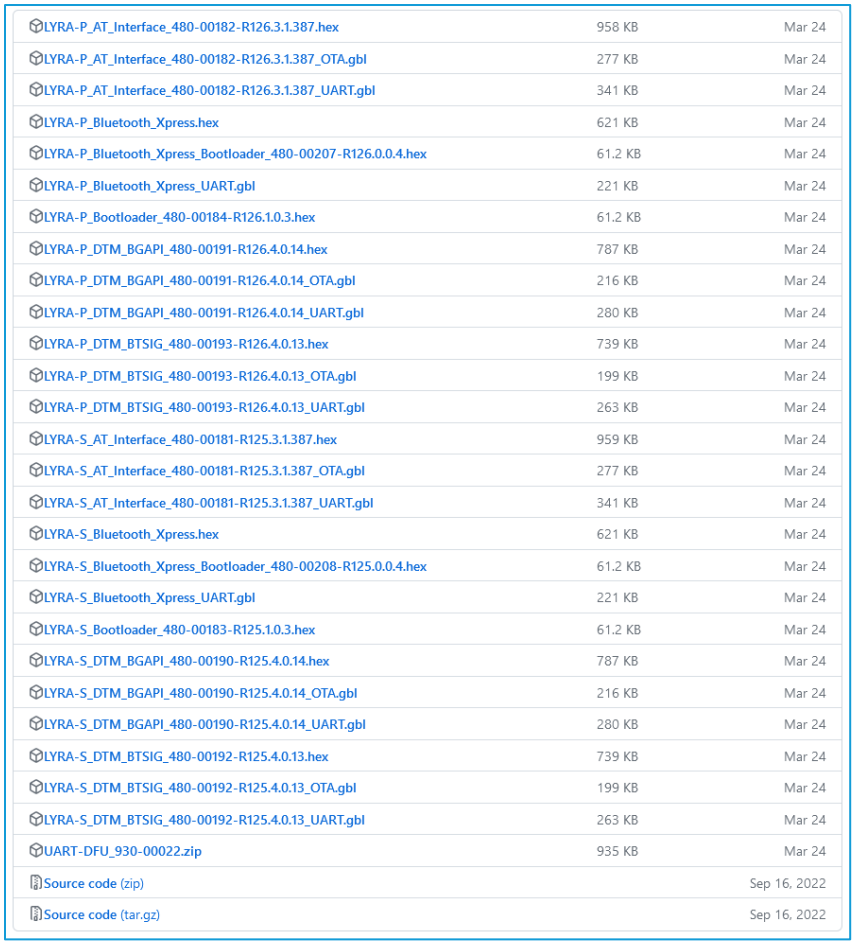
Module Variants (W) for Lyra: 126 Lyra (22) P +10 dBm, 125 Lyra (22) S +6 dBm, 129 Lyra 24P +10 dBm, 131 Lyra 24P +20 dBm, 133 Lyra 24P RF +20 dBm, and 130 Lyra 24S +10 dBm.

Example	LYRA-P_AT_Interface_480-00182-R126.3.1.173.hex	LYRA24-P20_AT_Interface_480-00259-R131.5.1.19.hex
V . . . ≙ . . .	480-00182	480-00259
W . . . ≙ . . .	126	131
X . . . ≙ . . .	3	5
Y . . . ≙ . . .	1	1
Z . . . ≙ . . .	173	19

Figure 1: Ezurio Code Format – Example for Lyra (22) and Lyra 24

Please note these details are omitted from the table below for clarity.

Image name	Equivalent to	Upgrade Method			Details
		SWD	SERIAL	OTA	
LYRA-P_AT_Interface.hex	[filename].hex	Y	N	N	Lyra P +10 dBm AT Interface
LYRA-P_AT_Interface_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA-P_AT_Interface_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA-P_Bluetooth_Xpress.hex	[filename].hex	Y	N	N	Lyra P +10 dBm Wireless Xpress
LYRA-P_Bluetooth_Xpress_Bootloader	[filename].hex	Y	N	N	
LYRA-P_Bluetooth_Xpress_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA-P_Bootloader.hex	[filename].hex	Y	N	N	Lyra P +10 dBm Ezurio Bootloader
LYRA-P_DTM_BGAPI.hex	[filename].hex	Y	N	N	Lyra P +10 dBm BGAPI Format DTM
LYRA-P_DTM_BGAPI_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA-P_DTM_BGAPI_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA-P_DTM_BTSIG.hex	[filename].hex	Y	N	N	Lyra P +10 dBm BTSIG Format DTM
LYRA-P_DTM_BTSIG _OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA-P_DTM_BTSIG _UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA-S_AT_Interface.hex	[filename].hex	Y	N	N	Lyra S +6 dBm AT Interface
LYRA-S_AT_Interface_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA-S_AT_Interface_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA-S_Bluetooth_Xpress.hex	[filename].hex	Y	N	N	Lyra S +6 dBm Wireless Xpress
LYRA-S_Bluetooth_Xpress_Bootloader	[filename].hex	Y	N	N	
LYRA-S_Bluetooth_Xpress_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA-S_Bootloader_.hex	[filename].hex	Y	N	N	Lyra S +6 dBm Ezurio Bootloader
LYRA-S_DTM_BGAPI.hex	[filename].hex	Y	N	N	Lyra S +6 dBm BGAPI Format DTM
LYRA-S_DTM_BGAPI_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA-S_DTM_BGAPI_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA-S_DTM_BTSIG.hex	[filename].hex	Y	N	N	Lyra S +6 dBm BTSIG Format DTM
LYRA-S_DTM_BTSIG_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA-S_DTM_BTSIG_UART.gbl	[filename]_full-crc.gbl	N	Y	N	



LYRA-P_AT_Interface_480-00182-R126.3.1.387.hex	958 KB	Mar 24
LYRA-P_AT_Interface_480-00182-R126.3.1.387_OTA.gbl	277 KB	Mar 24
LYRA-P_AT_Interface_480-00182-R126.3.1.387_UART.gbl	341 KB	Mar 24
LYRA-P_Bluetooth_Xpress.hex	621 KB	Mar 24
LYRA-P_Bluetooth_Xpress_Bootloader_480-00207-R126.0.0.4.hex	61.2 KB	Mar 24
LYRA-P_Bluetooth_Xpress_UART.gbl	221 KB	Mar 24
LYRA-P_Bootloader_480-00184-R126.1.0.3.hex	61.2 KB	Mar 24
LYRA-P_DTM_BGAPI_480-00191-R126.4.0.14.hex	787 KB	Mar 24
LYRA-P_DTM_BGAPI_480-00191-R126.4.0.14_OTA.gbl	216 KB	Mar 24
LYRA-P_DTM_BGAPI_480-00191-R126.4.0.14_UART.gbl	280 KB	Mar 24
LYRA-P_DTM_BTSIG_480-00193-R126.4.0.13.hex	739 KB	Mar 24
LYRA-P_DTM_BTSIG_480-00193-R126.4.0.13_OTA.gbl	199 KB	Mar 24
LYRA-P_DTM_BTSIG_480-00193-R126.4.0.13_UART.gbl	263 KB	Mar 24
LYRA-S_AT_Interface_480-00181-R125.3.1.387.hex	959 KB	Mar 24
LYRA-S_AT_Interface_480-00181-R125.3.1.387_OTA.gbl	277 KB	Mar 24
LYRA-S_AT_Interface_480-00181-R125.3.1.387_UART.gbl	341 KB	Mar 24
LYRA-S_Bluetooth_Xpress.hex	621 KB	Mar 24
LYRA-S_Bluetooth_Xpress_Bootloader_480-00208-R125.0.0.4.hex	61.2 KB	Mar 24
LYRA-S_Bluetooth_Xpress_UART.gbl	221 KB	Mar 24
LYRA-S_Bootloader_480-00183-R125.1.0.3.hex	61.2 KB	Mar 24
LYRA-S_DTM_BGAPI_480-00190-R125.4.0.14.hex	787 KB	Mar 24
LYRA-S_DTM_BGAPI_480-00190-R125.4.0.14_OTA.gbl	216 KB	Mar 24
LYRA-S_DTM_BGAPI_480-00190-R125.4.0.14_UART.gbl	280 KB	Mar 24
LYRA-S_DTM_BTSIG_480-00192-R125.4.0.13.hex	739 KB	Mar 24
LYRA-S_DTM_BTSIG_480-00192-R125.4.0.13_OTA.gbl	199 KB	Mar 24
LYRA-S_DTM_BTSIG_480-00192-R125.4.0.13_UART.gbl	263 KB	Mar 24
UART-DFU_930-00022.zip	935 KB	Mar 24
Source code (zip)		Sep 16, 2022
Source code (tar.gz)		Sep 16, 2022

Figure 2: GitHub Snapshot of all Lyra (22) GA 2 firmware files.

Image name	Equivalent to	Upgrade Method			Details
		SWD	SERIAL	OTA	
LYRA24-P10_AT_Interface.hex	[filename].hex	Y	N	N	Lyra 24P +10 dBm AT Interface
LYRA24-P10_AT_Interface_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-P10_AT_Interface_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-P10_Bootloader.hex	[filename].hex	Y	N	N	Lyra 24P +10 dBm Ezurio Bootloader
LYRA24-P10_DTM_BGAPI.hex	[filename].hex	Y	N	N	Lyra 24P +10 dBm BGAPI Format DTM
LYRA24-P10_DTM_BGAPI_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-P10_DTM_BGAPI_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-P10_DTM_BTSIG.hex	[filename].hex	Y	N	N	Lyra 24P +10 dBm BTSIG Format DTM
LYRA24-P10_DTM_BTSIG _OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-P10_DTM_BTSIG _UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-P20_AT_Interface.hex	[filename].hex	Y	N	N	Lyra 24P +20 dBm AT Interface
LYRA24-P20_AT_Interface_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-P20_AT_Interface_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-P20_Bootloader.hex	[filename].hex	Y	N	N	Lyra 24P +20 dBm Ezurio Bootloader
LYRA24-P20_DTM_BGAPI.hex	[filename].hex	Y	N	N	Lyra 24P +20 dBm BGAPI Format DTM
LYRA24-P20_DTM_BGAPI_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-P20_DTM_BGAPI_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-P20_DTM_BTSIG.hex	[filename].hex	Y	N	N	Lyra 24P +20 dBm BTSIG format DTM
LYRA24-P20_DTM_BTSIG _OTA.gbl	[filename]-crc.gbl	N	Y	Y	

Image name	Equivalent to	Upgrade Method			Details
		SWD	SERIAL	OTA	
LYRA24-P20_DTM_BTSIG _UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-P20RF_AT_Interface.hex	[filename].hex	Y	N	N	Lyra 24P RF +20 dBm AT Interface
LYRA24-P20RF_AT_Interface_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-P20RF_AT_Interface_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-P20RF_Bootloader.hex	[filename].hex	Y	N	N	Lyra 24P RF +20 dBm Ezurio Bootloader
LYRA24-P20RF_DTM_BGAPI.hex	[filename].hex	Y	N	N	Lyra 24P RF +20 dBm BGAPI format DTM
LYRA24-P20RF_DTM_BGAPI_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-P20RF_DTM_BGAPI_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-P20RF_DTM_BTSIG.hex	[filename].hex	Y	N	N	Lyra 24P RF +20 dBm BTSIG Format DTM
LYRA24-P20RF_DTM_BTSIG _OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-P20RF_DTM_BTSIG _UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-S10_AT_Interface.hex	[filename].hex	Y	N	N	Lyra 24S +10 dBm AT Interface
LYRA24-S10_AT_Interface_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-S10_AT_Interface_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-S10_Bootloader.hex	[filename].hex	Y	N	N	Lyra 24S +10 dBm Ezurio Bootloader
LYRA24-S10_DTM_BGAPI.hex	[filename].hex	Y	N	N	Lyra 24S +10 dBm BGAPI Format DTM
LYRA24-S10_DTM_BGAPI_OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-S10_DTM_BGAPI_UART.gbl	[filename]_full-crc.gbl	N	Y	N	
LYRA24-S10_DTM_BTSIG.hex	[filename].hex	Y	N	N	Lyra 24S +10 dBm BTSIG Format DTM
LYRA24-S10_DTM_BTSIG _OTA.gbl	[filename]-crc.gbl	N	Y	Y	
LYRA24-S10_DTM_BTSIG _UART.gbl	[filename]_full-crc.gbl	N	Y	N	

LYRA24-P10_AT_Interface_480-00224-R129.5.1.43.hex	1.03 MB	Jun 29
LYRA24-P10_AT_Interface_480-00225-R129.5.1.43_UART.gbl	374 KB	Jun 29
LYRA24-P10_AT_Interface_480-00226-R129.5.1.43_OTA.gbl	374 KB	Jun 29
LYRA24-P10_Bootloader_480-00230-R129.4.0.10.hex	201 KB	Jun 29
LYRA24-P10_DTM_BGAPI_480-00232-R129.5.0.7.hex	656 KB	Jun 29
LYRA24-P10_DTM_BGAPI_480-00233-R129.5.0.7_UART.gbl	233 KB	Jun 29
LYRA24-P10_DTM_BGAPI_480-00234-R129.5.0.7_OTA.gbl	233 KB	Jun 29
LYRA24-P10_DTM_BTSIG_480-00238-R129.5.0.5.hex	567 KB	Jun 29
LYRA24-P10_DTM_BTSIG_480-00239-R129.5.0.5_UART.gbl	202 KB	Jun 29
LYRA24-P10_DTM_BTSIG_480-00240-R129.5.0.5_OTA.gbl	202 KB	Jun 29
LYRA24-P20RF_AT_Interface_480-00289-R133.5.1.16.hex	822 KB	Jun 29
LYRA24-P20RF_AT_Interface_480-00290-R133.5.1.16_UART.gbl	292 KB	Jun 29
LYRA24-P20RF_AT_Interface_480-00291-R133.5.1.16_OTA.gbl	292 KB	Jun 29
LYRA24-P20RF_Bootloader_480-00288-R133.4.0.6.hex	201 KB	Jun 29
LYRA24-P20RF_DTM_BGAPI_480-00292-R133.5.0.3.hex	657 KB	Jun 29
LYRA24-P20RF_DTM_BGAPI_480-00293-R133.5.0.3_UART.gbl	234 KB	Jun 29
LYRA24-P20RF_DTM_BGAPI_480-00294-R133.5.0.3_OTA.gbl	234 KB	Jun 29
LYRA24-P20RF_DTM_BTSIG_480-00295-R133.5.0.4.hex	568 KB	Jun 29
LYRA24-P20RF_DTM_BTSIG_480-00296-R133.5.0.4_UART.gbl	202 KB	Jun 29
LYRA24-P20RF_DTM_BTSIG_480-00297-R133.5.0.4_OTA.gbl	202 KB	Jun 29
LYRA24-P20_AT_Interface_480-00259-R131.5.1.17.hex	822 KB	Jun 29
LYRA24-P20_AT_Interface_480-00260-R131.5.1.17_UART.gbl	292 KB	Jun 29
LYRA24-P20_AT_Interface_480-00261-R131.5.1.17_OTA.gbl	292 KB	Jun 29
LYRA24-P20_Bootloader_480-00262-R131.4.0.4.hex	201 KB	Jun 29
LYRA24-P20_DTM_BGAPI_480-00263-R131.5.0.4.hex	657 KB	Jun 29
LYRA24-P20_DTM_BGAPI_480-00264-R131.5.0.4_UART.gbl	234 KB	Jun 29
LYRA24-P20_DTM_BGAPI_480-00265-R131.5.0.4_OTA.gbl	234 KB	Jun 29
LYRA24-P20_DTM_BTSIG_480-00164-R131.5.0.5_UART.gbl	202 KB	Jun 29
LYRA24-P20_DTM_BTSIG_480-00165-R131.5.0.5_OTA.gbl	202 KB	Jun 29
LYRA24-P20_DTM_BTSIG_480-00266-R131.5.0.5.hex	568 KB	Jun 29
LYRA24-S10_AT_Interface_480-00270-R130.5.1.16.hex	1.03 MB	Jun 29
LYRA24-S10_AT_Interface_480-00271-R130.5.1.16_UART.gbl	374 KB	Jun 29
LYRA24-S10_AT_Interface_480-00272-R130.5.1.16_OTA.gbl	374 KB	Jun 29
LYRA24-S10_Bootloader_480-00231-R130.4.0.2.hex	201 KB	Jun 29
LYRA24-S10_DTM_BGAPI_480-00235-R130.5.0.5.hex	756 KB	Jun 29
LYRA24-S10_DTM_BGAPI_480-00236-R130.5.0.5_UART.gbl	269 KB	Jun 29
LYRA24-S10_DTM_BGAPI_480-00237-R130.5.0.5_OTA.gbl	269 KB	Jun 29
LYRA24-S10_DTM_BTSIG_480-00241-R130.5.0.5.hex	567 KB	Jun 29
LYRA24-S10_DTM_BTSIG_480-00242-R130.5.0.5_UART.gbl	202 KB	Jun 29
LYRA24-S10_DTM_BTSIG_480-00243-R130.5.0.5_OTA.gbl	202 KB	Jun 29
UART-DFU_930-00022.zip	935 KB	Jun 29
Source code (zip)		Mar 30
Source code (tar.gz)		Mar 30

Figure 3: GitHub Snapshot of all Lyra 24 GA 1 firmware files.

5 Hardware Preparation

1. Download and extract the latest available Lyra Series GA software release for your matching Lyra (22) and/or Lyra 24 hardware version.
2. Based on your firmware selection, set the **Switch (SW1)** on your Lyra DVK either in the *AT / BGx* or *C-DEV* position.
3. Connect your Lyra DVK to your PC via the included type A to micro type B USB cable, and make sure that the blue "J-Link" LED is active.
4. Do not forget to mount and attach the included external antenna to your Lyra (22) or 24 SIP development board; skip for Lyra PCB versions.

Notes(s) The (left-handed) AT / BGx position should be used when programming the Lyra Bootloader, AT Command Set or Bluetooth Xpress firmware. The (right-handed) C-DEV position should be used when performing native C development.

Versions of the Lyra (22) development board silkscreen prior to v2.3 may list SW1 options as *BGx or AT/SWO*. On these boards, *BGx* should be understood to be *AT/BGx* and *AT/SWO* should be understood to be *C-DEV*. Refer to the "Why does the Software Development Selection Switch (SW1) on my Lyra development board look different?" FAQ for additional information.

All Lyra development boards will show up and identified as a **JLink CDC UART Port (SEGGER)** device in the Windows device manager under "Ports (COM & LPT)". If the driver is not installed automatically when the DVK is plugged into your Windows PC, then please manually install Simplicity Studio 5 or the latest version of the J-Link Software and Documentation Pack.

6 Firmware Upgrade Methods

6.1 BLE (FOTA) using EFR Connect Mobile App

The BLE (FOTA) firmware upgrade method uses the Bluetooth wireless link and your Smartphone to load new firmware onto the Lyra module. We recommend the BLE (FOTA) firmware upgrade method, when possible, especially for deployed Lyra modules in the field or for Lyra modules controlled by end users.

The most common way to perform an update is to use the **Silicon Labs EFR Connect Mobile App**. This mobile application is available in both iOS and Android app stores – see https://www.silabs.com/developers/efr-connect-mobile-app.

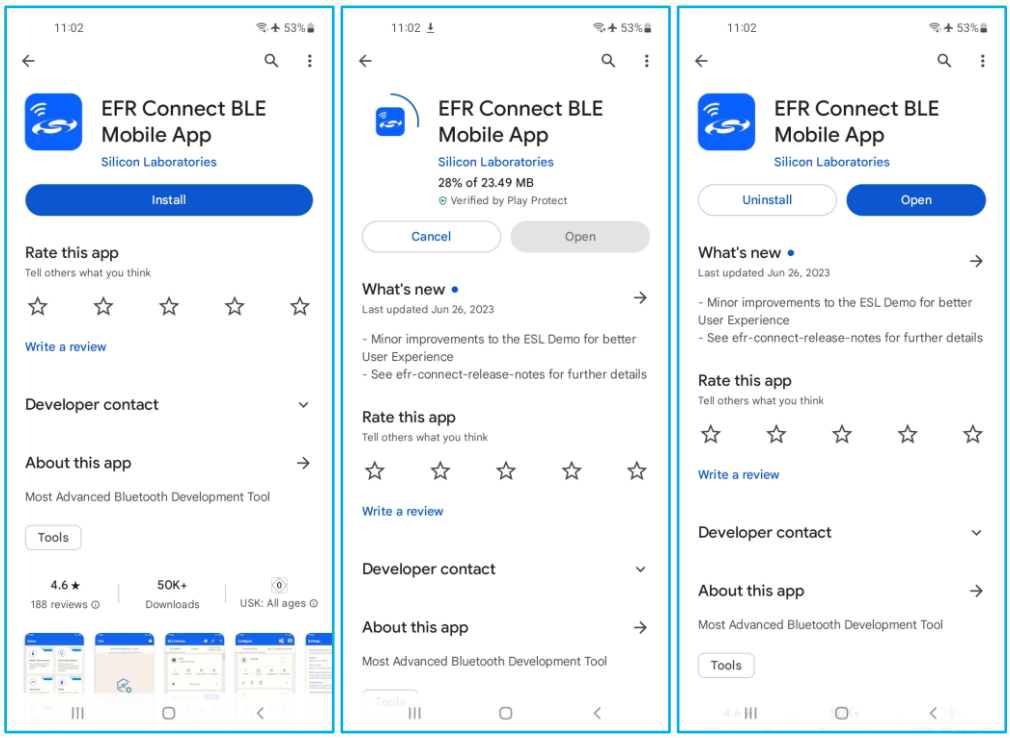
Lyra (22) only You can also update our Lyra (22) modules running the Wireless Xpress firmware with the **BGX Commander Mobile App**. Please see here for more information and instructions.

Before continuing, please first download and install the EFR Connect Mobile App for Android / iOS on your smartphone:

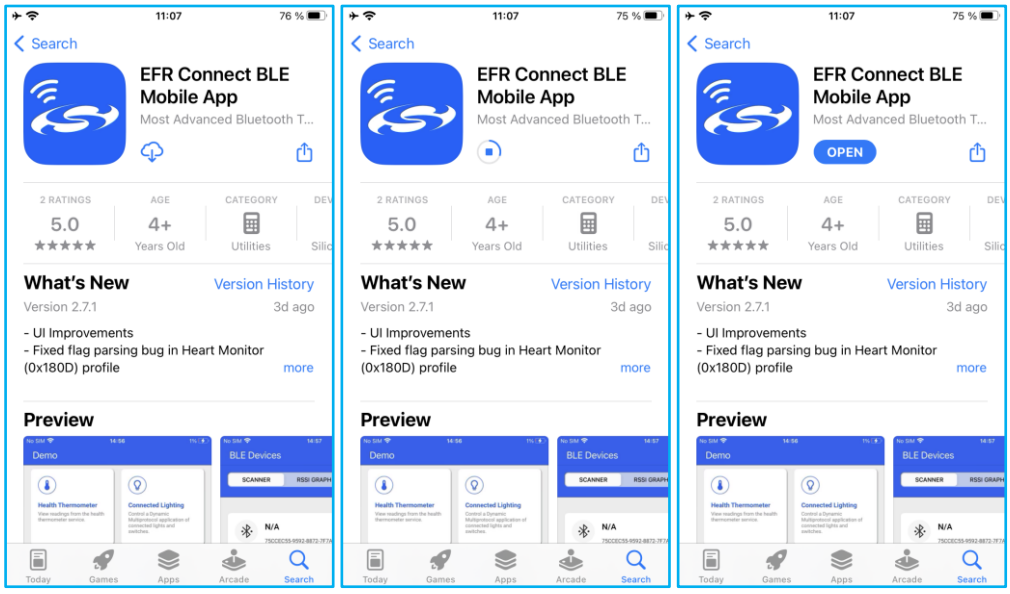
EFR Connect

Google Play Store

Android



EFR Connect
Apple Store
iOS

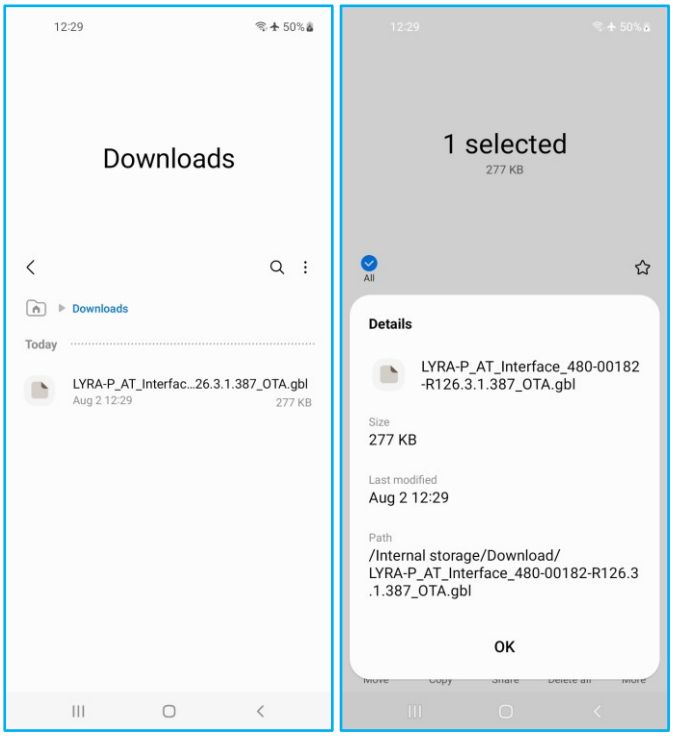


6.1.1 Lyra (22) Example – EFR Connect Mobile App under Android

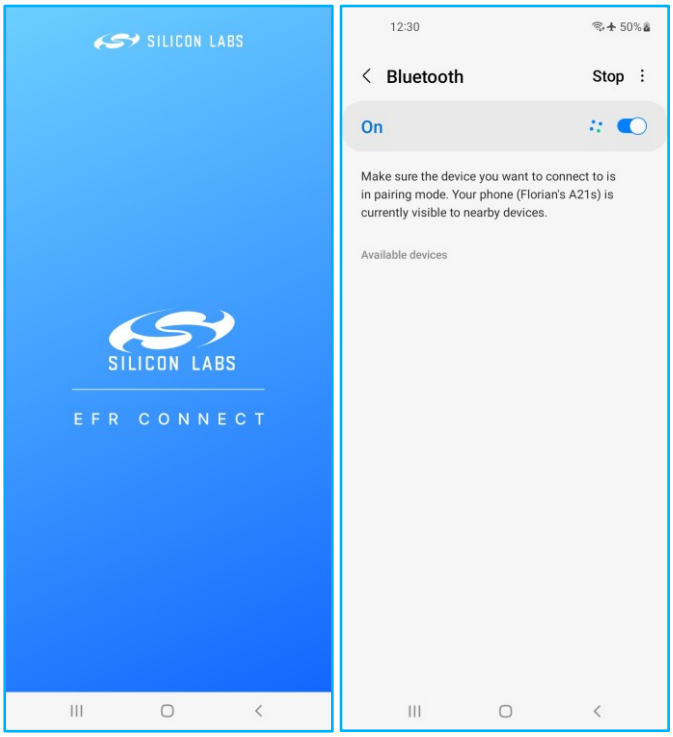
1. Upload the OTA firmware image file(s) to your smartphone, so that the EFR Connect mobile app can find and access it.

   For Android we recommend saving a copy to the "Downloads" (Internal Storage) folder by using your PC.

   In our example we are using the **LYRA-P_AT_Interface_480-00182-R126.3.1.387_OTA**.



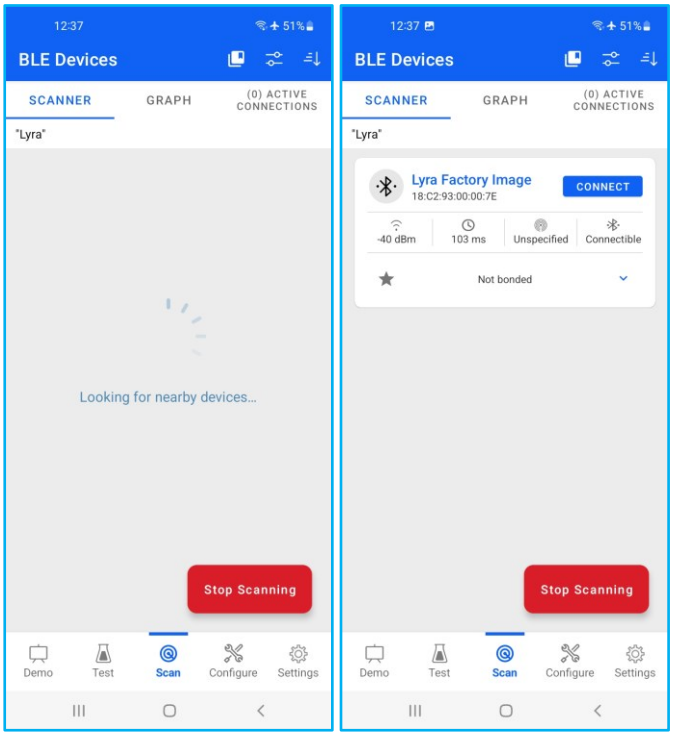
2. Start the EFR Connect mobile app on your smartphone and make sure that Bluetooth has been enabled in the Android settings.



3. Open the *Scan* tab and search for your Lyra development board or module. In most cases, it will appear in the list as **OTA**, **Lyra Factory Image** or with the **LYRA-** prefix.

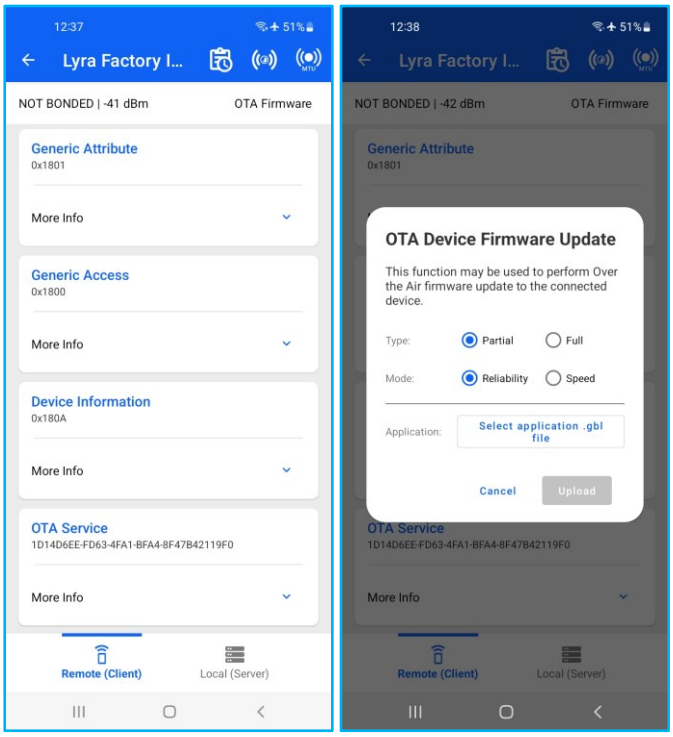
   Please simply ignore all other BLE devices which are found in this list. You can also use and enable a filter like in this example with “Lyra”.

   Tap on **Connect**. It may take a few seconds until the BLE connection has been fully established.



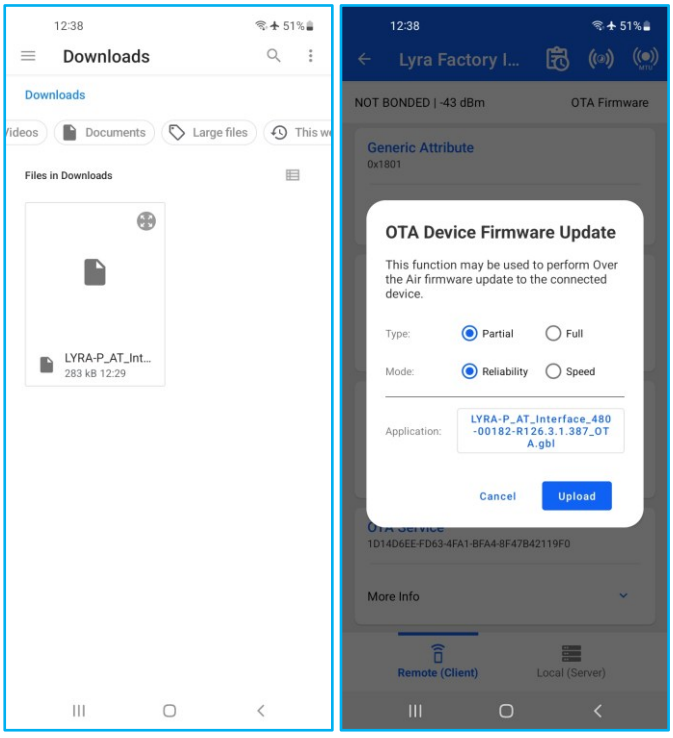
4. Verify that you can see the "OTA Service" with UUID **1D14D6EE-FD63-4FA1-BFA4-8F47B42119F0**.

   Next, switch to the **OTA Firmware** window which can found in the top-right corner.

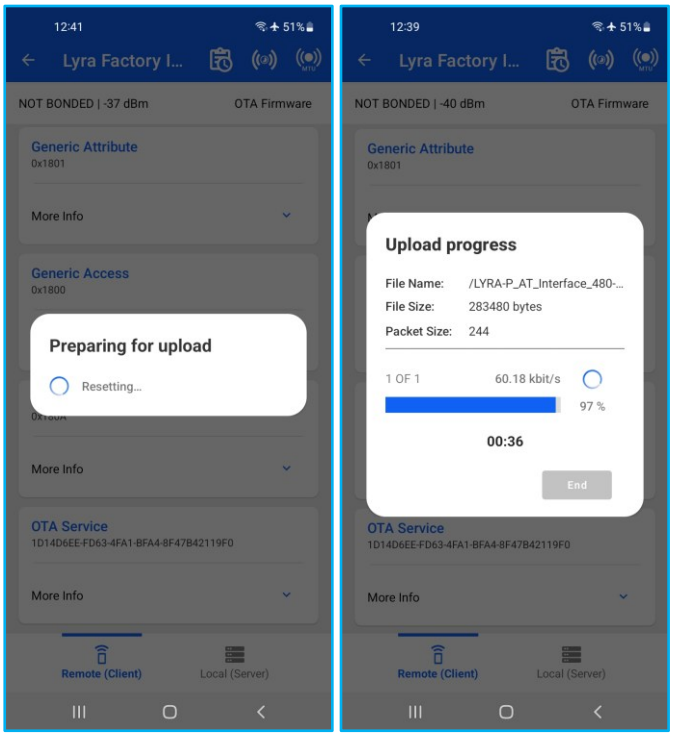


5. Apply the following settings in the **OTA Device Firmware Update** window:

   - **Type:** *Partial*
   - **Mode:** *Reliability* (Recommended)
   - **Application:** *LYRA-P_AT_Interface_480-00182-R126.3.1.387_OTA.*

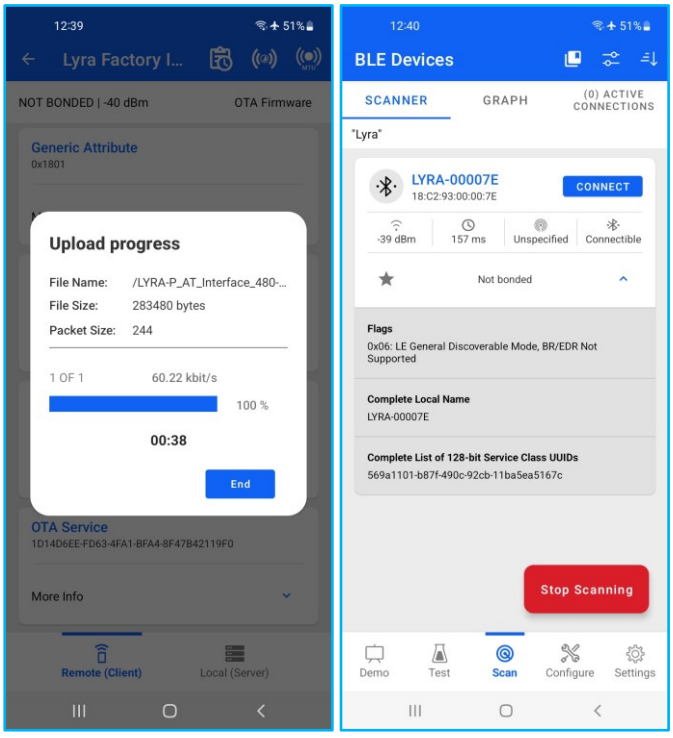


6. Start the update by pressing the **Upload** button. Keep the EFR Connect mobile app open and in foreground while the update is running. The whole update process usually does not take longer than 3 minutes.



7. Click **End.** Make sure to reset or power cycle your Lyra development board (by pressing the **RST** button) once the update is completed.

   It is also recommended to scan again in the EFR Connect mobile app, otherwise Lyra may does not appear.

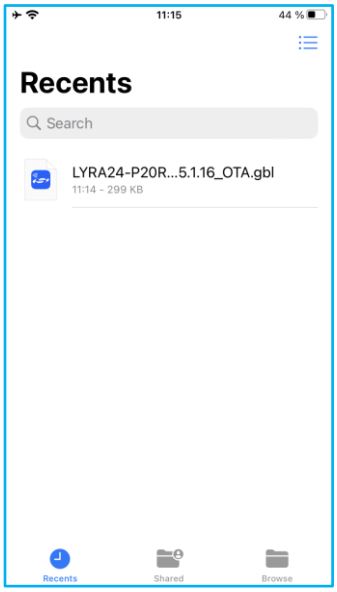


6.1.2 Lyra 24 Example – EFR Connect Mobile App under iOS

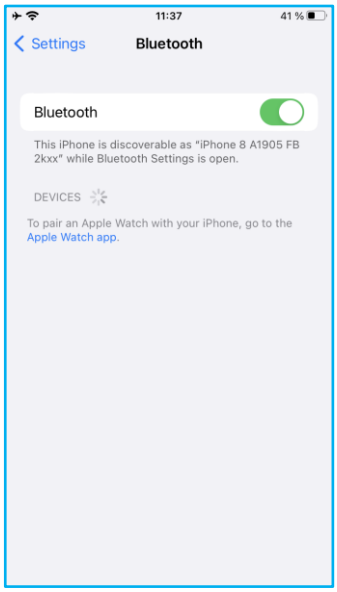
1. Upload the OTA firmware image file(s) to your smartphone, so that the EFR Connect mobile app can find and access it.

   For Apple we recommend saving a copy to a folder on your iCloud Drive.

   In our example we are using the **LYRA24-P20RF_AT_Interface_480-00291-R133.5.1.16_OTA.gbl.**



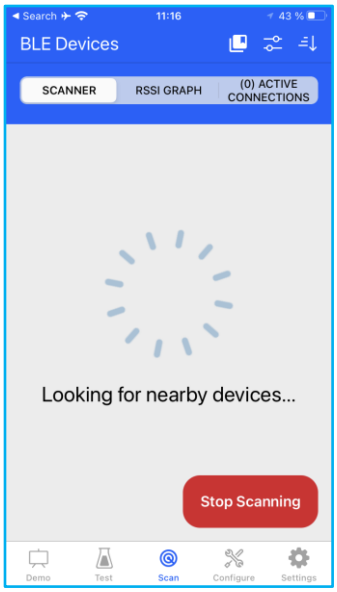
2. Start the EFR Connect mobile app on your smartphone and make sure that Bluetooth has been enabled in the iOS settings.

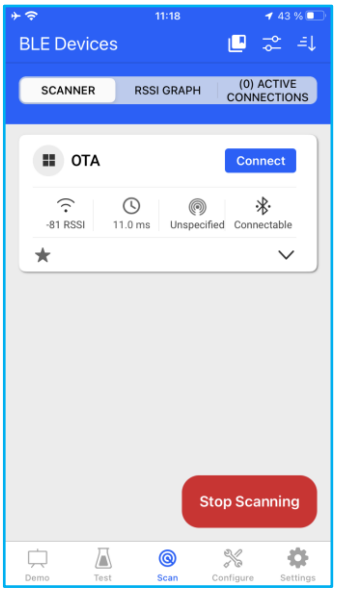


3. Open the *Scan* tab and search for your Lyra development board or module. In most cases, it will appear in the list as **OTA**, **Lyra Factory Image** or with the **LYRA-** prefix.

   Please simply ignore all other BLE devices which are found in this list. You can also use and enable a filter like in this example with "OTA".

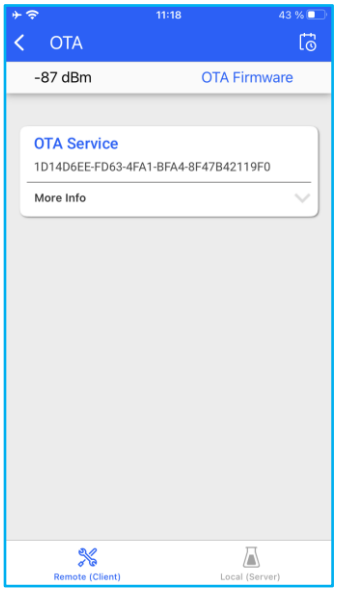
   Tap on **Connect**. It may take a few seconds until the BLE connection has been fully established.

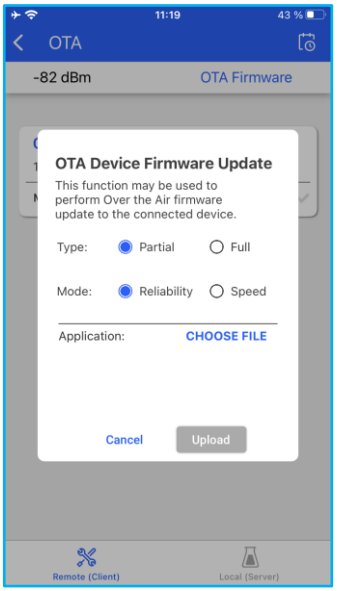




4. Verify that you can see the "OTA Service" with UUID **1D14D6EE-FD63-4FA1-BFA4-8F47B42119F0**.

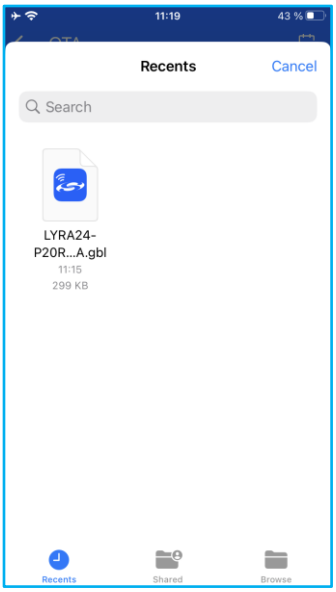
   Next, switch to the **OTA Firmware** window which can found in the top-right corner.

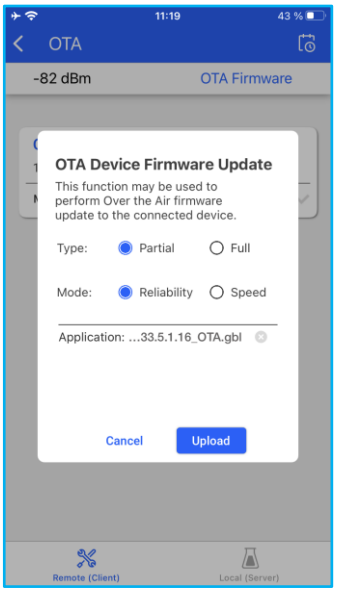




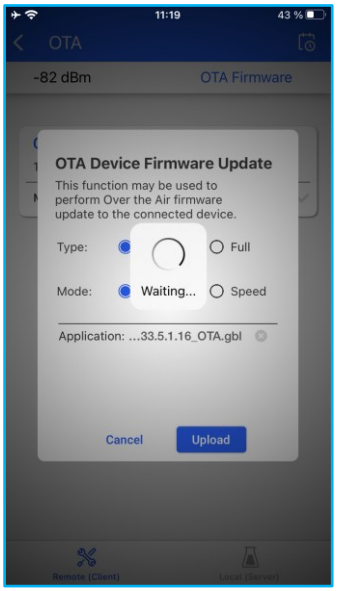
5. Apply the following settings in the **OTA Device Firmware Update** window:

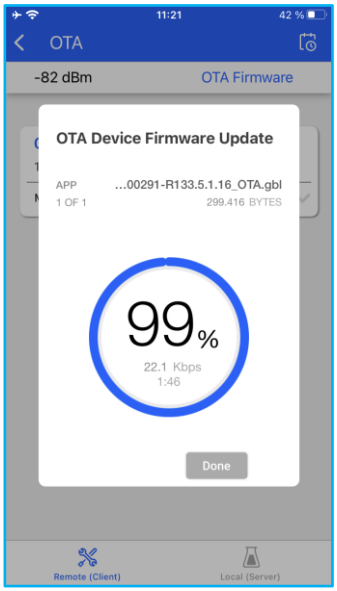
   - **Type:** *Partial*
   - **Mode:** *Reliability* (Recommended)
   - **Application:** *LYRA24-P20RF_AT_Interface_480-00291-R133.5.1.16_OTA.gbl.*





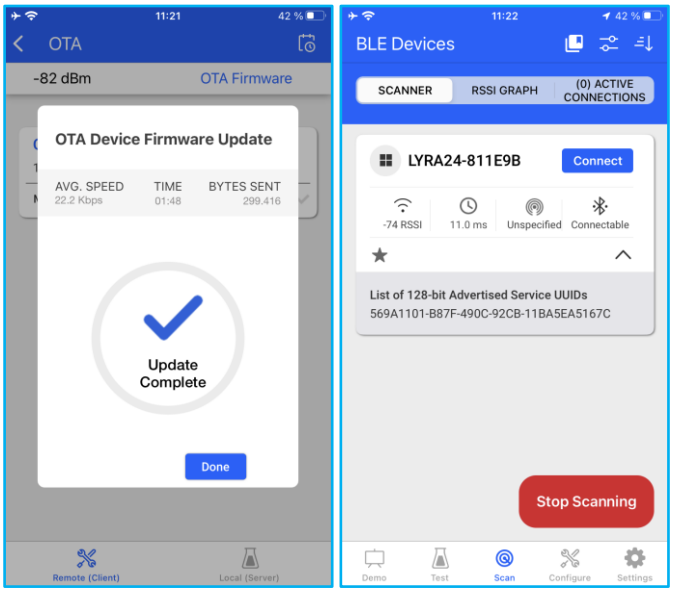
6. Start the update by pressing the **Update** button. Keep the EFR Connect mobile app open and in foreground while the update is running. The whole update process usually does not take longer than 3 minutes.





7. Click **Done.** Make sure to reset or power cycle your Lyra development board (by pressing the **RST** button) once the update is completed.

   It is also recommended to scan again in the EFR Connect mobile app, otherwise Lyra may does not appear.



6.2 Serial DFU (UART) using uart_dfu application

The Serial DFU (UART) firmware upgrade method allows the firmware to be updated via the serial interface using a pre-compiled application called **uart_dfu.** This is a common and powerful way to update firmware for our Lyra Series modules.

For this firmware upgrade method, it is mandatory to trigger the **BOOT** pin. The BOOT pin is an active-low digital input to force the module into DFU bootloader state upon device reset. It is used to determine when execution of the bootloader is required. Upon reset, execution of the bootloader begins. The state of the BOOT pin is read immediately upon start-up of the bootloader. If **LOW**, execution of the bootloader continues, facilitating firmware update via the UART. If the BOOT pin is **HIGH**, the bootloader will stop execution and pass control to the main application firmware.

Note: Due to different bootloaders pre-programmed in factory, it is very important to verify the BOOT pin against your current hardware design, especially when migrating from a BGX220P/BGM220P or BGX220S/BGM220S module to avoid any potential conflicts or unexpected behaviours in your existing application. Please refer to Table 1, Table 2, or the respective Lyra datasheets and DVK schematics for more information.

---

1. Download and extract the **Serial DFU (UART) Utility** (uart_dfu) provided by the Ezurio team from the Lyra firmware releases GitHub page [B] (Lyra 22) or [D] (Lyra 24) and copy the latest UART firmware image file(s) in the folder of the **uart_dfu.exe**. Make sure that the file names end with the "_UART.gbl" prefix and format as shown below.

   Depending on your Simplicity Studio 5 installation / configuration the source code and Makefile for the original version can be found in the Gecko SDK (GSDK). Please make sure that the latest version for the GSDK is installed in this case. In Windows, the application can be built using, for example, MinGW or Cygwin. Under Linux or Mac the program can be compiled with the GCC toolchain. Please refer to this FAQ for build instructions under Windows and/or Linux.

   - C:\SiliconLabs\SimplicityStudio\v5\developer\sdks\gecko_sdk_suite\<VERSION>\app\bluetooth\example_host\uart_dfu
   - C:\Users\<USERNAME>\SimplicityStudio\SDKs\gecko_sdk\app\bluetooth\example_host\bt_host_uart_dfu.

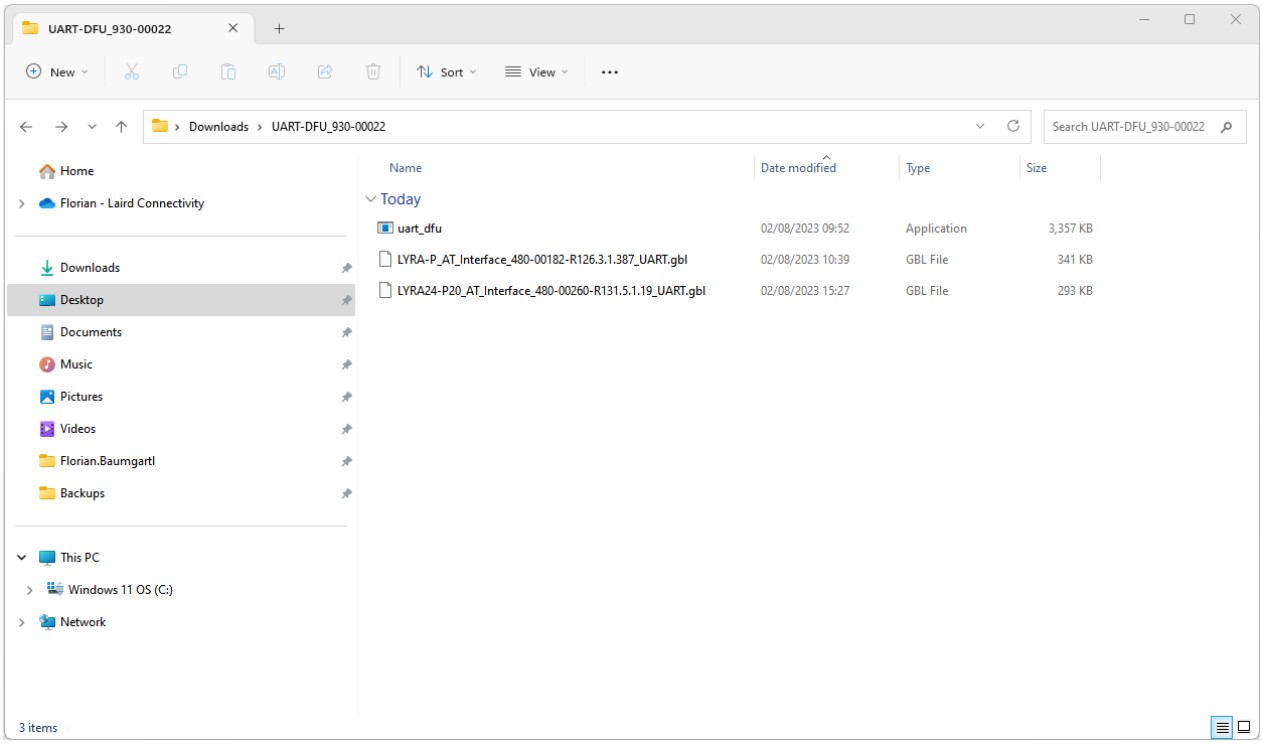


Figure 4: Serial DFU (UART) Utility and Firmware Image Files

2. Before continuing, we need to verify the **COM Port** for our Lyra Series – Development Kit via the **Windows Device Manager**. Press **Windows+R** which will open the **Run** window. Enter **devmgmt.msc** and then click **OK**. Find the COM Port for the **JLink CDC UART Port Device** in the **Ports (COM & LPT)** tab which is surrounded in brackets. In our example it is *COM5*.

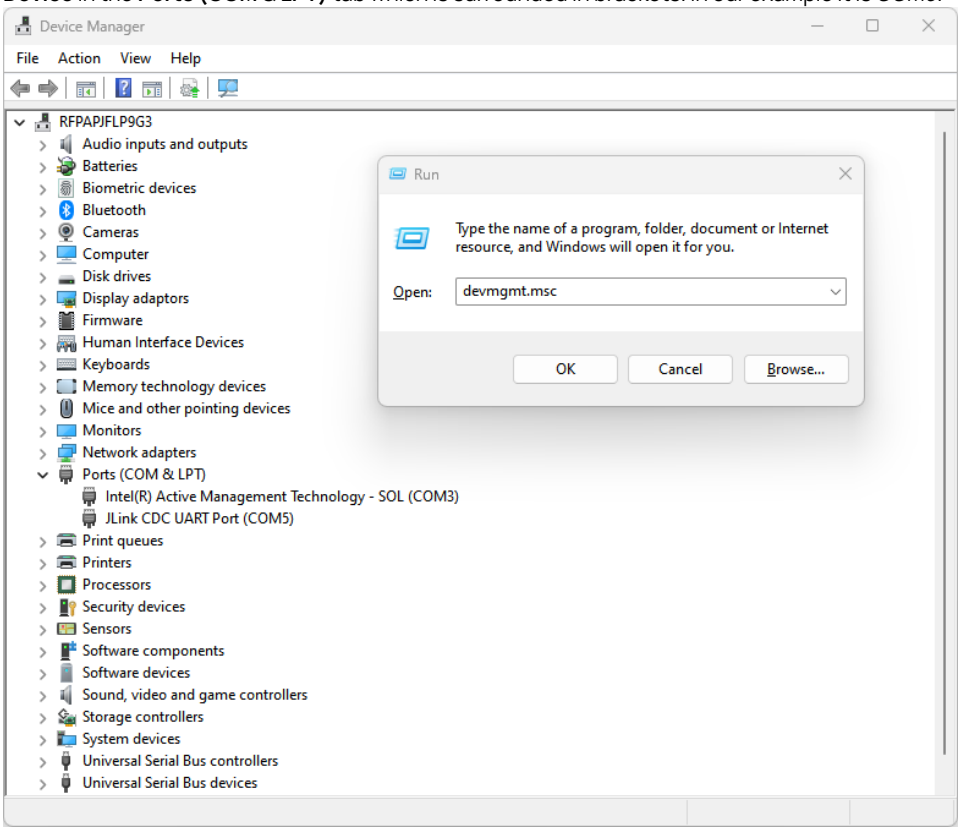


Figure 5: Windows Device Manager and Run Window.

3. Open the **File Explorer** in Windows and navigate to the uart_dfu folder which contains all UART firmware image files as well as the uart_dfu.exe. Click in the address bar or alternatively press **Alt+D**. Once the address bar is highlighted type **cmd** in the bar and press **Enter**. This will automatically set up a cmd window for the uart_dfu folder with the correct paths.

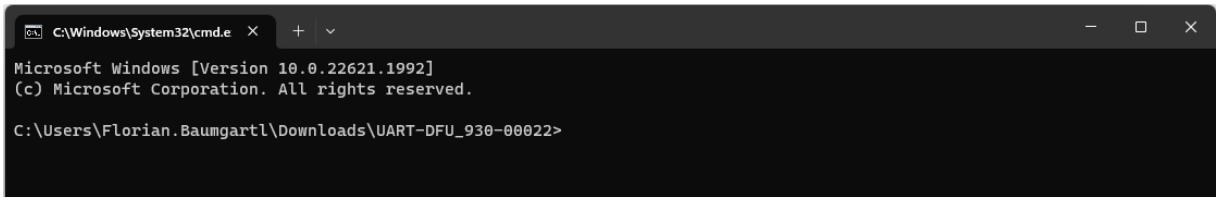


Figure 6: Windows CMD Window

4. Enter **uart_dfu.exe** without any arguments to test if the application works on your system. This will print the required input parameters for the application as shown in Figure 7.

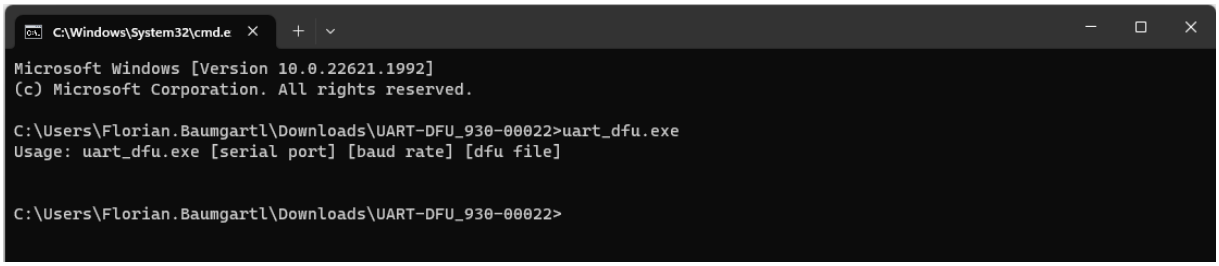


Figure 7: Windows CMD showing the uart.exe Usage Information.

The input parameters are described as follows.

- `[serial_port]`: This is the COM port where the Lyra DVK being updated resides, of the form COMx, where x is the COM port number.
- `[baud rate]`: This is the baud rate of the COM port in decimal numeric format. Default value is 115200.
- `[dfu file]`: This is the path and filename where the update file resides. Note that if the update file resides in the same directory location as the uart_dfu.exe application, only the filename is required.

For a Lyra DVK residing on **COM5** with **115200** baud, and an upgrade filename **LYRA-P_AT_Interface_480-00182-R126.3.1.387_UART.gbl**, with the Ezurio provided bootloader, the following command would be used to start the firmware upgrade via Serial DFU (UART), as shown in Figure 8.

```
uart_dfu.exe COM5 115200 LYRA-P_AT_Interface_480-00182-R126.3.1.387_UART.gbl
```

Note in this case the upgrade file resides in the same location as the uart_dfu.exe application. If the upgrade file resides in a different location, the full path and file name should be provided within quotation characters.

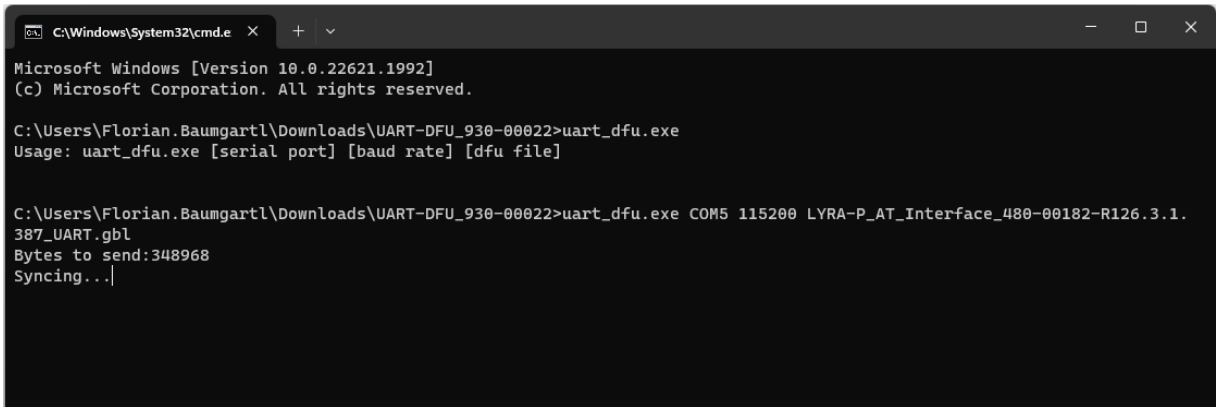


Figure 8: Windows CMD showing that the Firmware Upgrade via uart.exe was initiated and started.

5. Wait until you see the *Syncing…* message. Now press both **BTN0** and **RST** buttons on the Lyra Series – Development Kit simultaneously for at least 3 seconds. After this period of time, release the **RST** button only while continuing to hold **BTN0**. When you see a progress bar and a similar output as below, release **BTN0**.

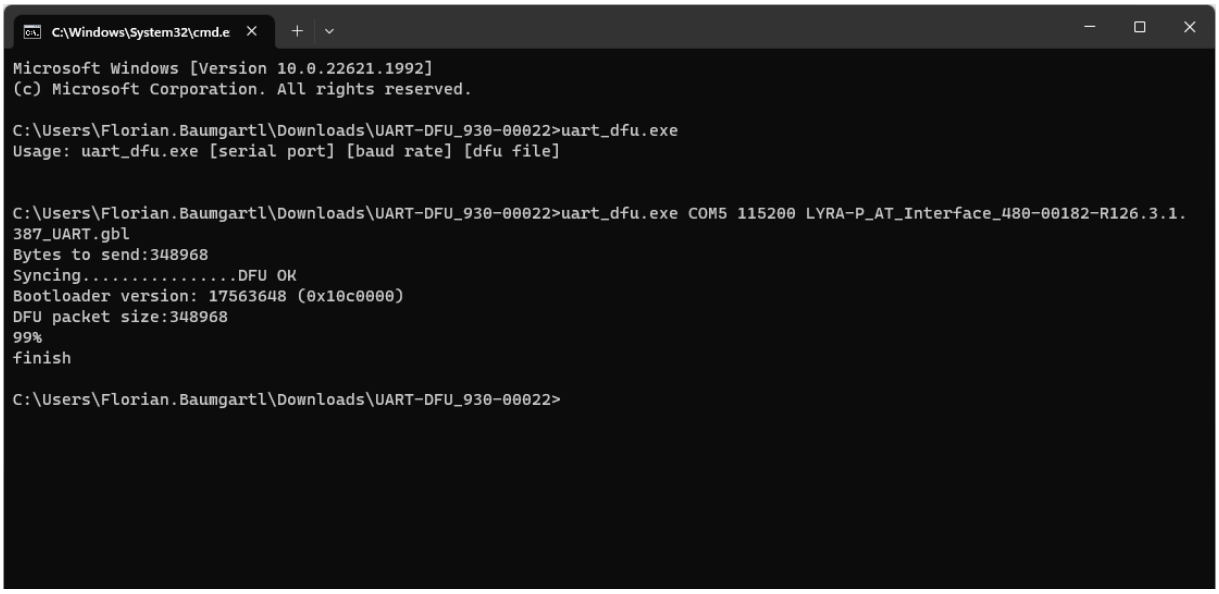


Figure 9: Windows CMD showing that the Firmware Upgrade via uart.exe successfully completed.

6. Please make sure to reset or power cycle the Lyra Series – Development Kit (by pressing the **RST** button) once you see the *finish* message, meaning that the update completed successfully.

6.2.1 Lyra (22) Example – uart_dfu.exe with User Commands and Output

User entered commands in this example are **highlighted**:

```
Microsoft Windows [Version 10.0.22621.1992]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Florian.Baumgartl\Downloads\UART-DFU_930-00022>uart_dfu.exe
Usage: uart_dfu.exe [serial port] [baud rate] [dfu file]


C:\Users\Florian.Baumgartl\Downloads\UART-DFU_930-00022>uart_dfu.exe COM5 115200 "LYRA-P_AT_Interface_480-00182-R126.3.1.387_UART.gbl"
Bytes to send:348968
Syncing..ID:04a00020
..DFU OK
Bootloader version: 17563648 (0x10c0000)
DFU packet size:348968
99%
finish

C:\Users\Florian.Baumgartl\Downloads\UART-DFU_930-00022>
```

6.2.2 Lyra 24 Example – uart_dfu.exe with User Commands and Output

User entered commands in this example are **highlighted**:

```
Microsoft Windows [Version 10.0.22621.1992]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Florian.Baumgartl\Downloads\UART-DFU_930-00022>uart_dfu.exe
Usage: uart_dfu.exe [serial port] [baud rate] [dfu file]


C:\Users\Florian.Baumgartl\Downloads\UART-DFU_930-00022>uart_dfu.exe COM4 115200 LYRA24-P20_AT_Interface_480-00260-R131.5.1.19_UART.gbl
Bytes to send:299380
Syncing.......ID:010000a0
.DFU OK
Bootloader version: 33685505 (0x2020001)
DFU packet size:299380
99%
finish

C:\Users\Florian.Baumgartl\Downloads\UART-DFU_930-00022>
```

6.3 Simplicity Commander & Simplicity Studio 5 (SWD)

Simplicity Commander is part of the **Simplicity Studio 5** and a single, all-purpose tool that can be used in a production environment to program our Lyra modules via a Graphical User Interface (GUI) or a Command Line Interface (CLI). It is also available as a standalone utility – without the need of the Simplicity Studio 5 IDE.

In Simplicity Studio 5, by default, the Simplicity Commander can open from the Launcher perspective. Click "Tools" in the top left corner, and then select the Simplicity Commander. For this the **Gecko SDK – 32-bit and Wireless MCUs** technology / software component must be installed.

The primary audience for this firmware upgrade method are software developers who develop their own C code application using the Bluetooth SDK. You can find additional usage information for both Simplicity Studio 5 and Simplicity Commander in the latest Simplicity Studio 5 User's Guide and UG162: Simplicity Commander Reference Guide.

Note: SWCLK and SWDIO pins are mandatory for this firmware upgrade method. Please verify your existing hardware design against them and make sure that PA01 (SWCLK) / PA02 (SWDIO) pins are exposed and/or available for use. This is also highly recommended for customers migrating from a BGX220P/BGM220P or BGX220S/BGM220S module. All Lyra (22) and Lyra 24 modules share the same pins for SWCLK and SWDIO. Please refer to our respective Lyra datasheets or DVK schematics for more information.

1. Before starting, please go to https://community.silabs.com/SL_CommunitiesSelfReg and register for a Silicon Labs account. Remember to confirm your account registration via email. This is important because it will grant you access to the Simplicity Commander as well as to all Simplicity Studio 5 features and Software Development Toolkits (SDK).

2. Download the Simplicity Commander Package from https://www.silabs.com/developers/simplicity-studio#commander and extract it on your PC. See the included README.txt for additional help. As a software dependency, the **J-Link Software and Documentation Pack** must be installed and available on your system – refer to section 6.4 SEGGER J-Link Commander (SWD) for more information.

   If Simplicity Studio 5 is installed, then you can simply ignore this step and locate the Simplicity Commander under *C:\SiliconLabs\SimplicityStudio\v5\developer\adapter_packs\commander*. Your installation path may be different.

3. Now open your **File Explorer** in Windows, navigate to the Simplicity Commander folder, and run the **commander.exe**. In our example, the folder path is: *C:\Users\<USERNAME>\Downloads\SimplicityCommander-Windows\SimplicityCommander-Windows\Commander_win32_x64_1v15p3b1357\Simplicity Commander*.

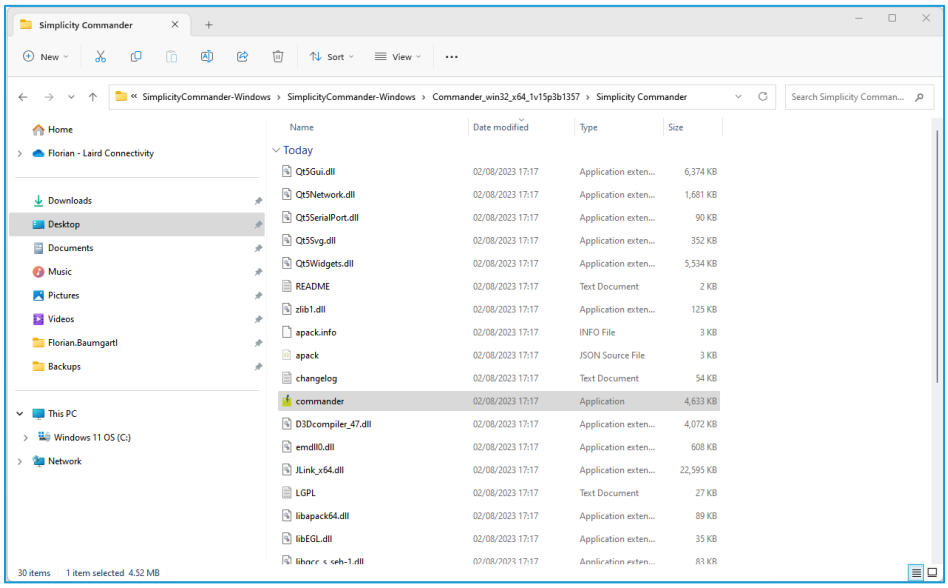


Figure 10: File Explorer showing the Simplicity Commander . . .

4. Go to **Select Kit...** and choose your Lyra Series – Development Kit. In our example it has the **Serial Number** *449005007*. Only one Lyra DVK should be connected to be correctly identified.

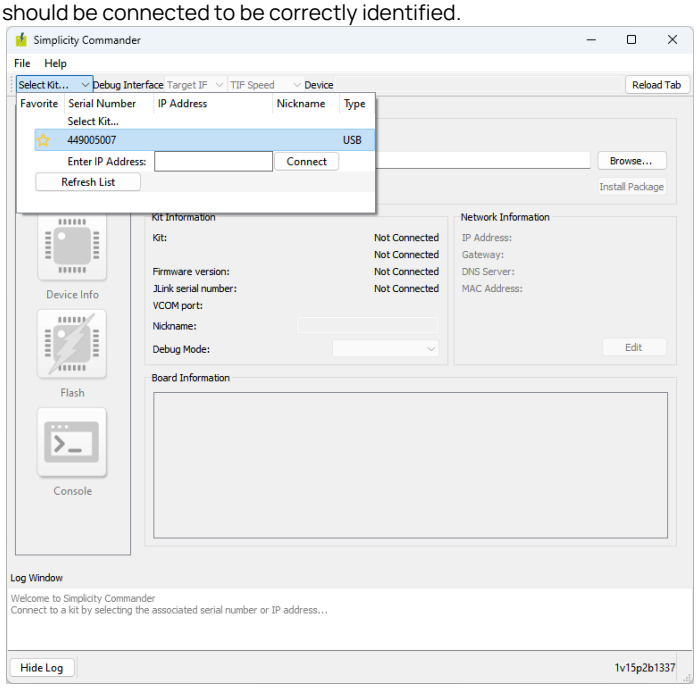


Figure 11: Selecting the Lyra DVK in Simplicity Commander

5. Once connected, you should see a *Connected to 449005007* message in the Log Window area. Also, make sure that the **Debug Mode** is set to **MCU** as shown below. If the Debug Mode MCU does not work for you, try **OUT** instead.

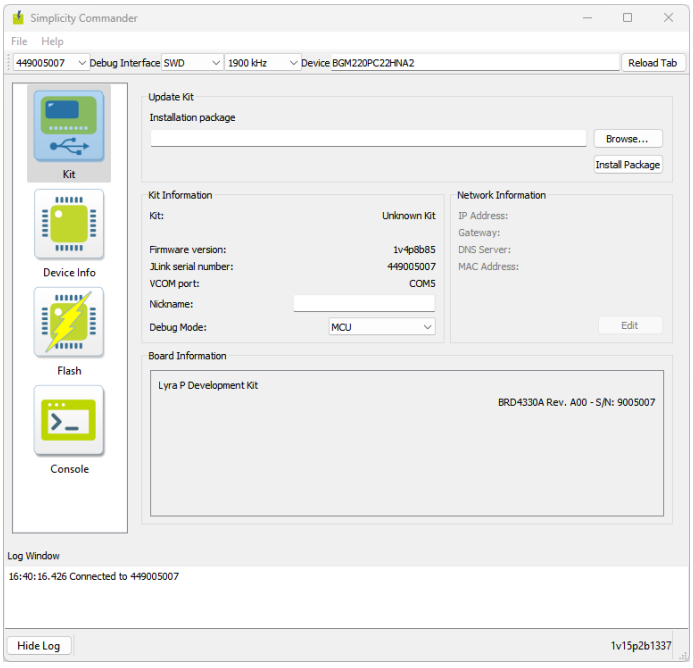


Figure 12: Lyra DVK is connected in Simplicity Commander

6. Next, navigate to the **Flash** page and click the **Erase Chip** button. You should see *Flash was erased successfully message* in the Log Window area. Please leave all other settings unchanged and to their default values.

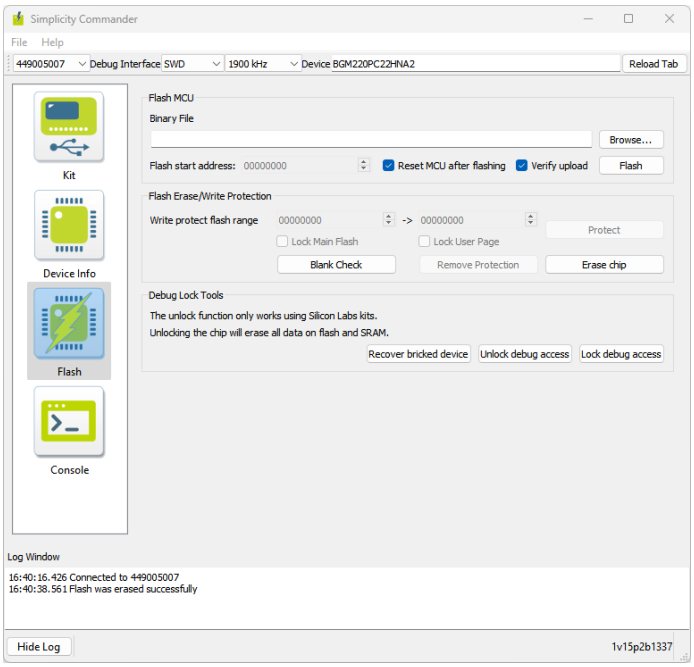


Figure 13: Chip erase in Simplicity Commander

7. Go to **Browse** and select the bootloader image first. Make sure to change your bootloader image file path accordingly if needed. In our example it is *C:\Users\<USERNAME>\Downloads\Lyra 22\LYRA-P_Bootloader_480-00184-R126.1.0.3.hex*. Press the **Flash** Button. This usually does not take longer than 3 seconds.

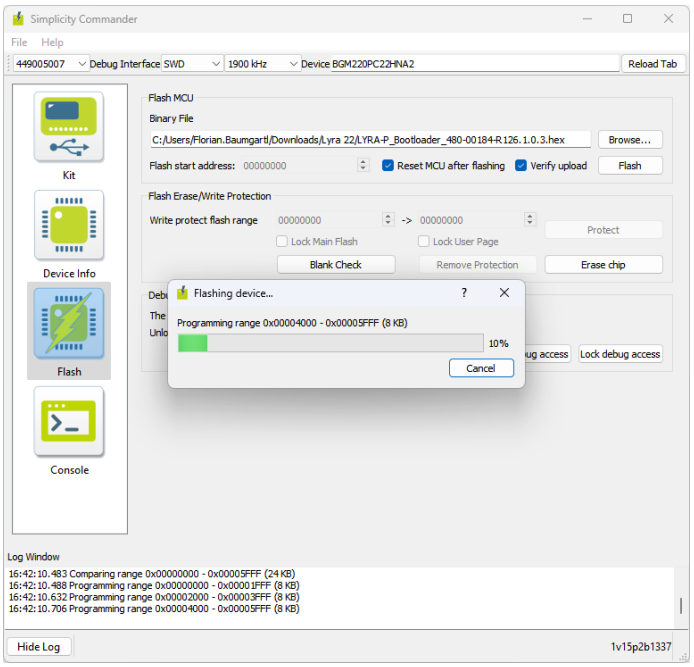


Figure 14: Flashing bootloader image in Simplicity Commander

8. Once programming of the bootloader image is finished, select the main application image file, and press again the **Flash** Button. In our example it is *C:\Users\<USERNAME>\Downloads\Lyra 22\LYRA-P_AT_Interface_480-00182-R126.3.1.387.hex*. This usually does not take longer than 10 seconds.

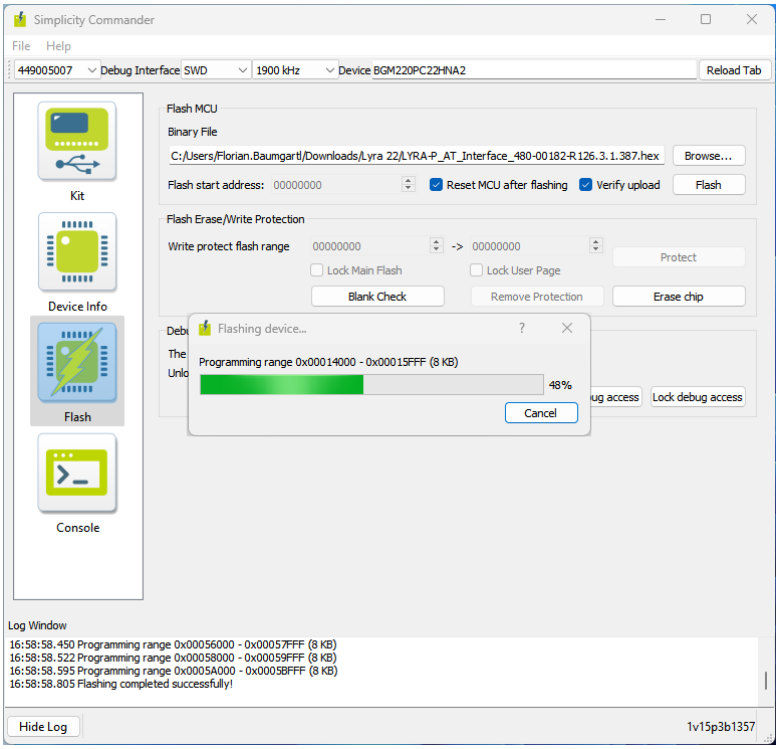


Figure 15: Flashing main application image in Simplicity Commander

9. Please make sure to reset or power cycle the Lyra Series – Development Kit (by pressing the **RST** button) once the programming of both bootloader and main application has been completed successfully.

6.3.1 Lyra (22) Example – Simplicity Commander (Log Window) Output

```
10:54:00.053 Connected to 449005007
10:54:04.801 Flash was erased successfully
10:54:14.237 Writing 24576 bytes starting at address 0x00000000
10:54:14.454 Comparing range 0x00000000 - 0x00005FFF (24 KB)
10:54:14.459 Programming range 0x00000000 - 0x00001FFF (8 KB)
10:54:14.603 Programming range 0x00002000 - 0x00003FFF (8 KB)
10:54:14.676 Programming range 0x00004000 - 0x00005FFF (8 KB)
10:54:14.901 Flashing completed successfully!
10:54:25.281 Writing 352256 bytes starting at address 0x00006000
10:54:25.497 Comparing range 0x00006000 - 0x00025FFF (128 KB)
10:54:25.514 Comparing range 0x00026000 - 0x00045FFF (128 KB)
10:54:25.532 Comparing range 0x00046000 - 0x0005BFFF (88 KB)
10:54:25.545 Programming range 0x00006000 - 0x00007FFF (8 KB)
10:54:25.690 Programming range 0x00008000 - 0x00009FFF (8 KB)
10:54:25.762 Programming range 0x0000A000 - 0x0000BFFF (8 KB)
10:54:25.835 Programming range 0x0000C000 - 0x0000DFFF (8 KB)
10:54:25.908 Programming range 0x0000E000 - 0x0000FFFF (8 KB)
10:54:25.980 Programming range 0x00010000 - 0x00011FFF (8 KB)
10:54:26.053 Programming range 0x00012000 - 0x00013FFF (8 KB)
10:54:26.126 Programming range 0x00014000 - 0x00015FFF (8 KB)
10:54:26.199 Programming range 0x00016000 - 0x00017FFF (8 KB)
10:54:26.271 Programming range 0x00018000 - 0x00019FFF (8 KB)
10:54:26.344 Programming range 0x0001A000 - 0x0001BFFF (8 KB)
10:54:26.417 Programming range 0x0001C000 - 0x0001DFFF (8 KB)
10:54:26.490 Programming range 0x0001E000 - 0x0001FFFF (8 KB)
10:54:26.562 Programming range 0x00020000 - 0x00021FFF (8 KB)
10:54:26.635 Programming range 0x00022000 - 0x00023FFF (8 KB)
10:54:26.708 Programming range 0x00024000 - 0x00025FFF (8 KB)
10:54:26.781 Programming range 0x00026000 - 0x00027FFF (8 KB)
10:54:26.853 Programming range 0x00028000 - 0x00029FFF (8 KB)
```

```
10:54:26.926 Programming range 0x0002A000 - 0x0002BFFF (8 KB)
10:54:26.999 Programming range 0x0002C000 - 0x0002DFFF (8 KB)
10:54:27.071 Programming range 0x0002E000 - 0x0002FFFF (8 KB)
10:54:27.144 Programming range 0x00030000 - 0x00031FFF (8 KB)
10:54:27.217 Programming range 0x00032000 - 0x00033FFF (8 KB)
10:54:27.289 Programming range 0x00034000 - 0x00035FFF (8 KB)
10:54:27.362 Programming range 0x00036000 - 0x00037FFF (8 KB)
10:54:27.435 Programming range 0x00038000 - 0x00039FFF (8 KB)
10:54:27.507 Programming range 0x0003A000 - 0x0003BFFF (8 KB)
10:54:27.580 Programming range 0x0003C000 - 0x0003DFFF (8 KB)
10:54:27.653 Programming range 0x0003E000 - 0x0003FFFF (8 KB)
10:54:27.726 Programming range 0x00040000 - 0x00041FFF (8 KB)
10:54:27.799 Programming range 0x00042000 - 0x00043FFF (8 KB)
10:54:27.872 Programming range 0x00044000 - 0x00045FFF (8 KB)
10:54:27.944 Programming range 0x00046000 - 0x00047FFF (8 KB)
10:54:28.016 Programming range 0x00048000 - 0x00049FFF (8 KB)
10:54:28.089 Programming range 0x0004A000 - 0x0004BFFF (8 KB)
10:54:28.162 Programming range 0x0004C000 - 0x0004DFFF (8 KB)
10:54:28.235 Programming range 0x0004E000 - 0x0004FFFF (8 KB)
10:54:28.308 Programming range 0x00050000 - 0x00051FFF (8 KB)
10:54:28.380 Programming range 0x00052000 - 0x00053FFF (8 KB)
10:54:28.453 Programming range 0x00054000 - 0x00055FFF (8 KB)
10:54:28.526 Programming range 0x00056000 - 0x00057FFF (8 KB)
10:54:28.598 Programming range 0x00058000 - 0x00059FFF (8 KB)
10:54:28.671 Programming range 0x0005A000 - 0x0005BFFF (8 KB)
10:54:28.894 Flashing completed successfully!
```

6.3.2 Lyra 24 Example – Simplicity Commander (Log Window) Output

```
11:00:33.049 Connected to 449006017
11:00:51.340 Flash was erased successfully
11:00:57.655 Writing 73728 bytes starting at address 0x08000000
11:00:57.869 Comparing range 0x08000000 - 0x08011FFF (72 KB)
11:00:57.876 Programming range 0x08000000 - 0x08001FFF (8 KB)
11:00:58.020 Programming range 0x08002000 - 0x08003FFF (8 KB)
11:00:58.092 Programming range 0x08004000 - 0x08005FFF (8 KB)
11:00:58.165 Programming range 0x08006000 - 0x08007FFF (8 KB)
11:00:58.238 Programming range 0x08008000 - 0x08009FFF (8 KB)
11:00:58.310 Programming range 0x0800A000 - 0x0800BFFF (8 KB)
11:00:58.383 Programming range 0x0800C000 - 0x0800DFFF (8 KB)
11:00:58.455 Programming range 0x0800E000 - 0x0800FFFF (8 KB)
11:00:58.528 Programming range 0x08010000 - 0x08011FFF (8 KB)
11:00:58.764 Flashing completed successfully!
11:01:03.112 Writing 303104 bytes starting at address 0x08012000
11:01:03.328 Comparing range 0x08012000 - 0x08031FFF (128 KB)
11:01:03.342 Comparing range 0x08032000 - 0x08051FFF (128 KB)
11:01:03.356 Comparing range 0x08052000 - 0x0805BFFF (40 KB)
11:01:03.364 Programming range 0x08012000 - 0x08013FFF (8 KB)
11:01:03.508 Programming range 0x08014000 - 0x08015FFF (8 KB)
11:01:03.581 Programming range 0x08016000 - 0x08017FFF (8 KB)
11:01:03.654 Programming range 0x08018000 - 0x08019FFF (8 KB)
11:01:03.726 Programming range 0x0801A000 - 0x0801BFFF (8 KB)
11:01:03.799 Programming range 0x0801C000 - 0x0801DFFF (8 KB)
11:01:03.872 Programming range 0x0801E000 - 0x0801FFFF (8 KB)
11:01:03.944 Programming range 0x08020000 - 0x08021FFF (8 KB)
11:01:04.017 Programming range 0x08022000 - 0x08023FFF (8 KB)
11:01:04.090 Programming range 0x08024000 - 0x08025FFF (8 KB)
11:01:04.162 Programming range 0x08026000 - 0x08027FFF (8 KB)
11:01:04.236 Programming range 0x08028000 - 0x08029FFF (8 KB)
11:01:04.308 Programming range 0x0802A000 - 0x0802BFFF (8 KB)
11:01:04.380 Programming range 0x0802C000 - 0x0802DFFF (8 KB)
11:01:04.453 Programming range 0x0802E000 - 0x0802FFFF (8 KB)
11:01:04.526 Programming range 0x08030000 - 0x08031FFF (8 KB)
11:01:04.599 Programming range 0x08032000 - 0x08033FFF (8 KB)
11:01:04.671 Programming range 0x08034000 - 0x08035FFF (8 KB)
11:01:04.744 Programming range 0x08036000 - 0x08037FFF (8 KB)
11:01:04.816 Programming range 0x08038000 - 0x08039FFF (8 KB)
11:01:04.889 Programming range 0x0803A000 - 0x0803BFFF (8 KB)
11:01:04.962 Programming range 0x0803C000 - 0x0803DFFF (8 KB)
11:01:05.034 Programming range 0x0803E000 - 0x0803FFFF (8 KB)
11:01:05.107 Programming range 0x08040000 - 0x08041FFF (8 KB)
11:01:05.179 Programming range 0x08042000 - 0x08043FFF (8 KB)
11:01:05.253 Programming range 0x08044000 - 0x08045FFF (8 KB)
```

```
11:01:05.324 Programming range 0x08046000 - 0x08047FFF (8 KB)
11:01:05.397 Programming range 0x08048000 - 0x08049FFF (8 KB)
11:01:05.469 Programming range 0x0804A000 - 0x0804BFFF (8 KB)
11:01:05.542 Programming range 0x0804C000 - 0x0804DFFF (8 KB)
11:01:05.615 Programming range 0x0804E000 - 0x0804FFFF (8 KB)
11:01:05.687 Programming range 0x08050000 - 0x08051FFF (8 KB)
11:01:05.760 Programming range 0x08052000 - 0x08053FFF (8 KB)
11:01:05.833 Programming range 0x08054000 - 0x08055FFF (8 KB)
11:01:05.906 Programming range 0x08056000 - 0x08057FFF (8 KB)
11:01:05.979 Programming range 0x08058000 - 0x08059FFF (8 KB)
11:01:06.051 Programming range 0x0805A000 - 0x0805BFFF (8 KB)
11:01:06.289 Flashing completed successfully!
```

6.4 SEGGER J-Link Commander (SWD)

J-Link Commander is a tool provided by SEGGER that interfaces with devices attached to compatible J-Link debugger like our Lyra Series – Development Kits or any other compatible J-Link debuggers. It for example allows you to erase and flash our bootloader and main application images in an easy way through a command-line interface (CLI). It perfectly fits and suits for all software engineers, hardware engineers, and release engineers who are familiar with programming system-on-chips (SoCs) and modules. This firmware upgrade method is also recommended for module recovery.

Note: SWCLK and SWDIO pins are mandatory for this firmware upgrade method. Please verify your existing hardware design against them and make sure that PA01 (SWCLK) / PA02 (SWDIO) pins are exposed and/or available for use. This is also highly recommended for customers migrating from a BGX220P/BGM220P or BGX220S/BGM220S module. All Lyra (22) and Lyra 24 modules share the same pins for SWCLK and SWDIO. Please refer to our respective Lyra datasheets or DVK schematics for more information.

---

1. Please download and install the latest version of the **J-Link Software and Documentation Pack** which is available under https://www.segger.com/downloads/J-Link/. It's highly recommended not to work with any beta version(s). Further details, resources and instructions about this tool can be found here.
2. Navigate to the default **C:\Program Files\SEGGER\JLink** folder and start the **SEGGER J-Link Commander** by running the **JLink.exe** application. Depending on your installation, paths may differ here.

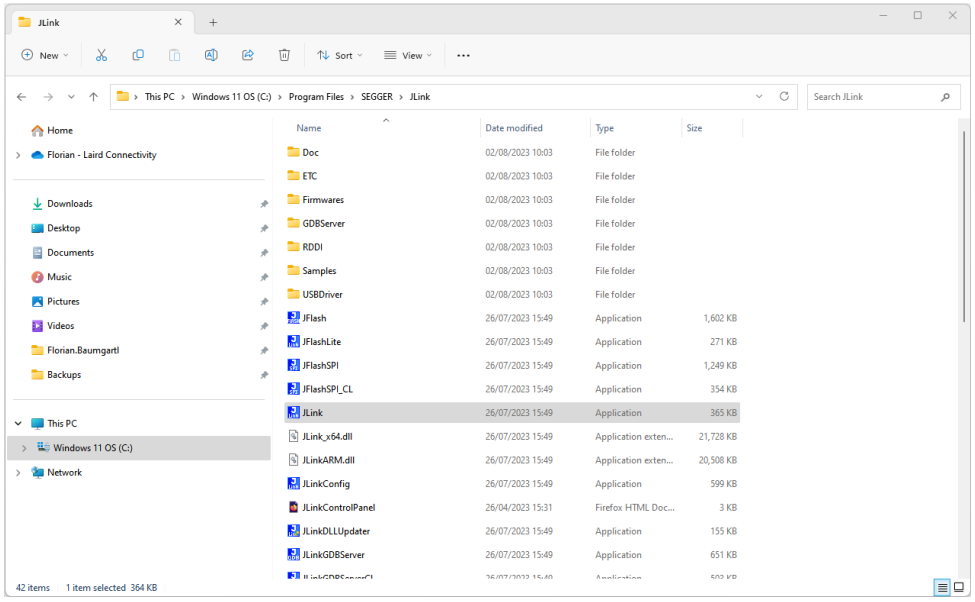


Figure 16: SEGGER J-Link Software and Documentation Pack after installation

3. Start by connecting to our Lyra Series – Development Kit. This is very easy, simply enter **connect** in the *J-Link>* line.

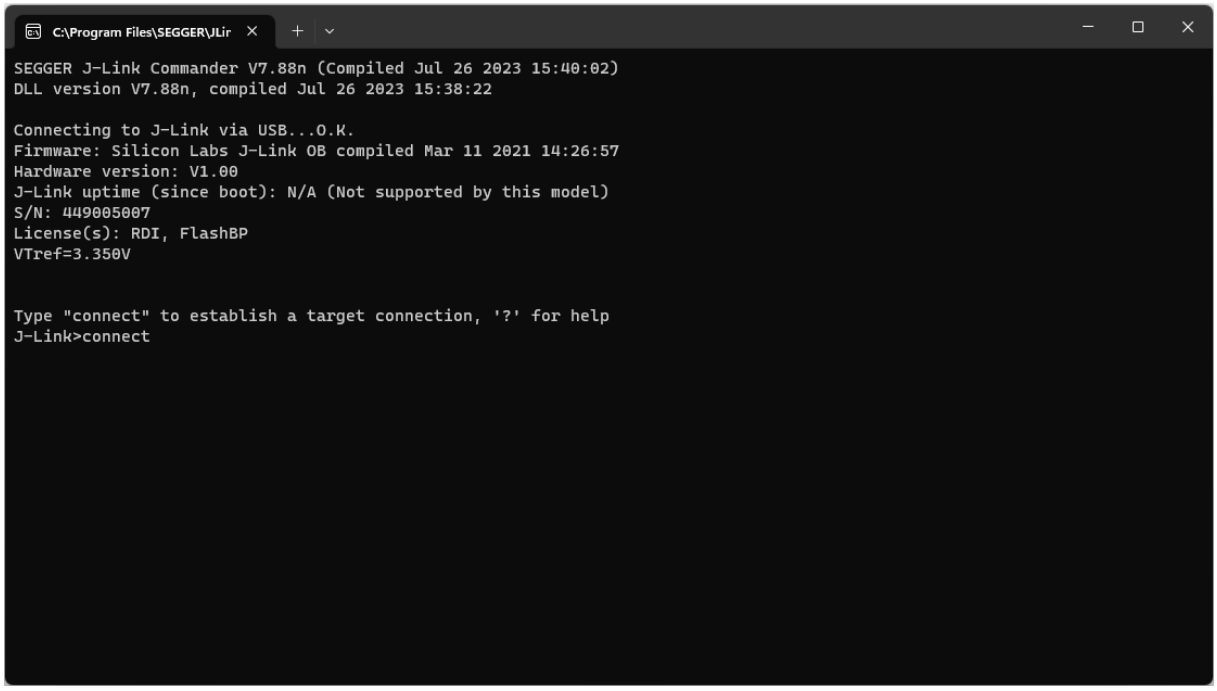


Figure 17: J-Link Commander & Lyra DVK connection establishment.

4. J-Link Commander then prompts for a **target device / core**. Only one Lyra development board should be connected to be correctly identified. Enter **EFR32BG22CXXXF512** (Lyra 22) or **EFR32BG24BXXXF1536** (Lyra 24) in the *Device>* line or enter **?** which will open an additional specification dialog with a complete device list:

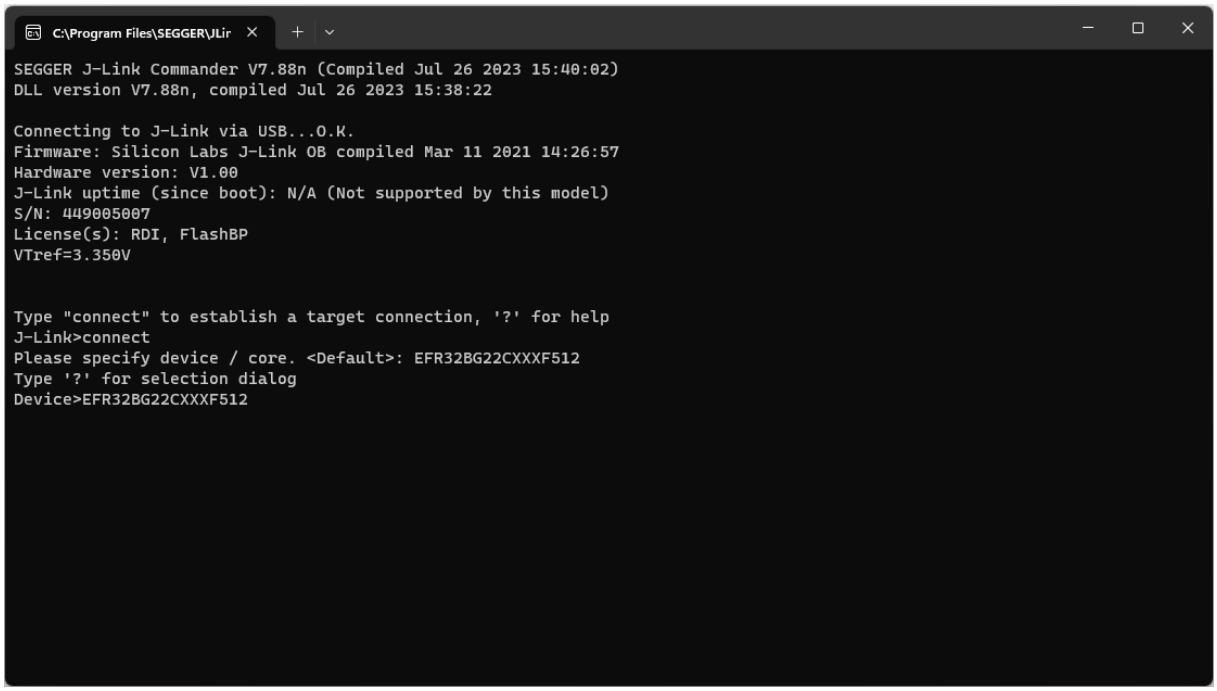


Figure 18: J-Link Commander is asking for target device / core.

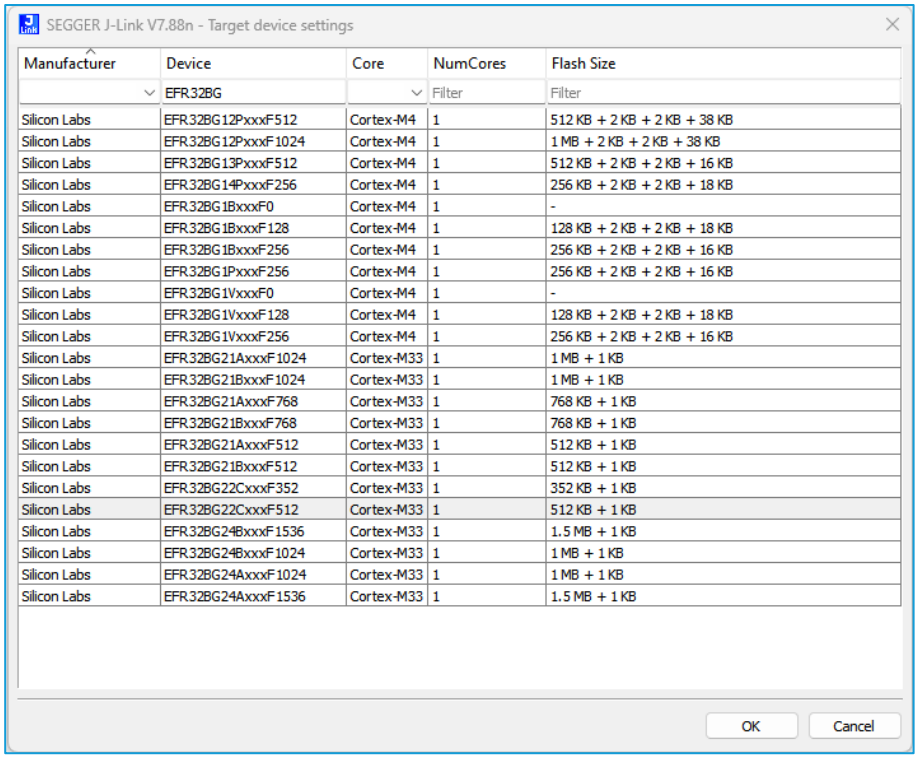


Figure 19: J-Link Commander full device list window when entering ? command.

5. Next, J-Link Commander will ask for the correct **target interface**. Enter **S** in the *TIF>* line.

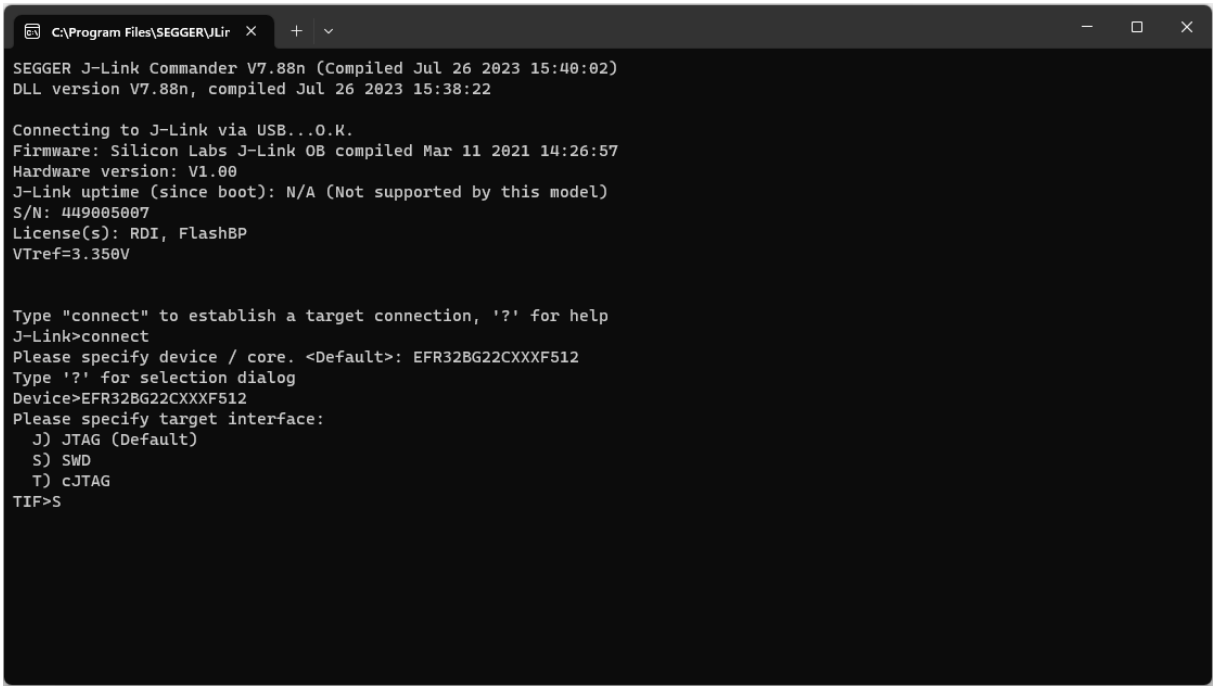


Figure 20: SEGGER J-Link Commander asking for target interface.

6. J-Link Commander also needs the correct **target interface speed** to proceed. The default value of *4000 kHz* can be selected here. Simply hit **Enter** or manual enter 4000 in the *Speed>* line to confirm the selection.

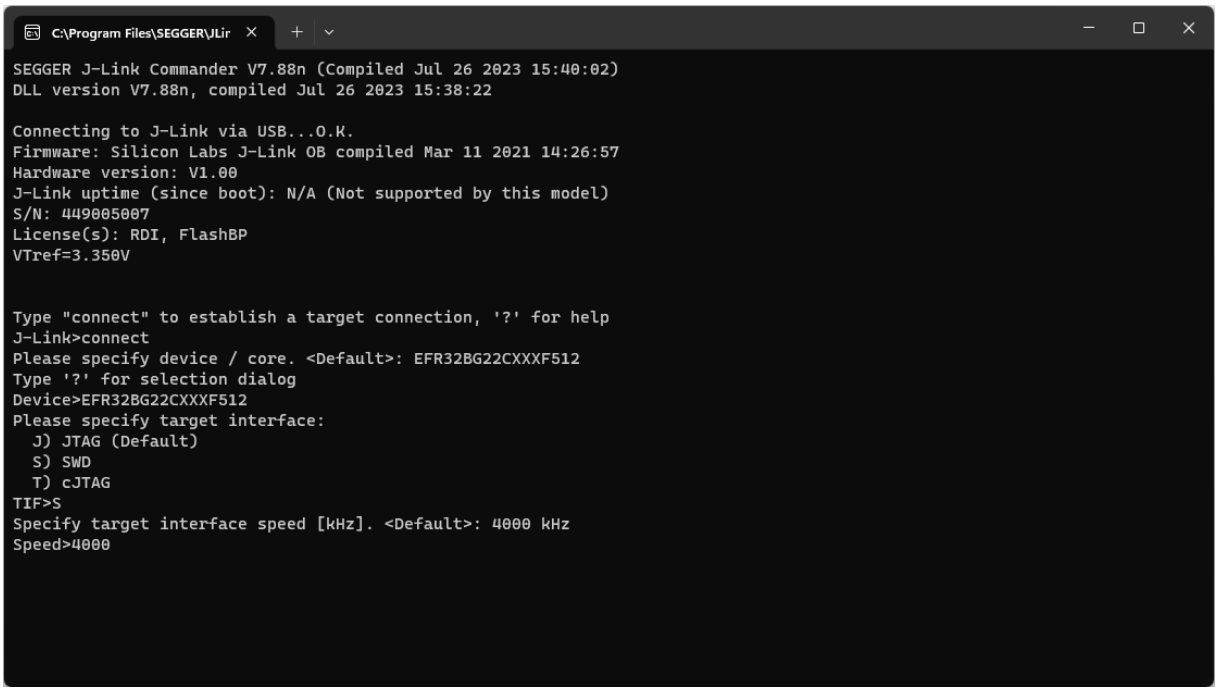


Figure 21: SEGGER J-Link Commander asking for target interface speed.

7. Check for a *Cortex-M33 identified.* message indicating a successful SWD connection to the Lyra DVK:

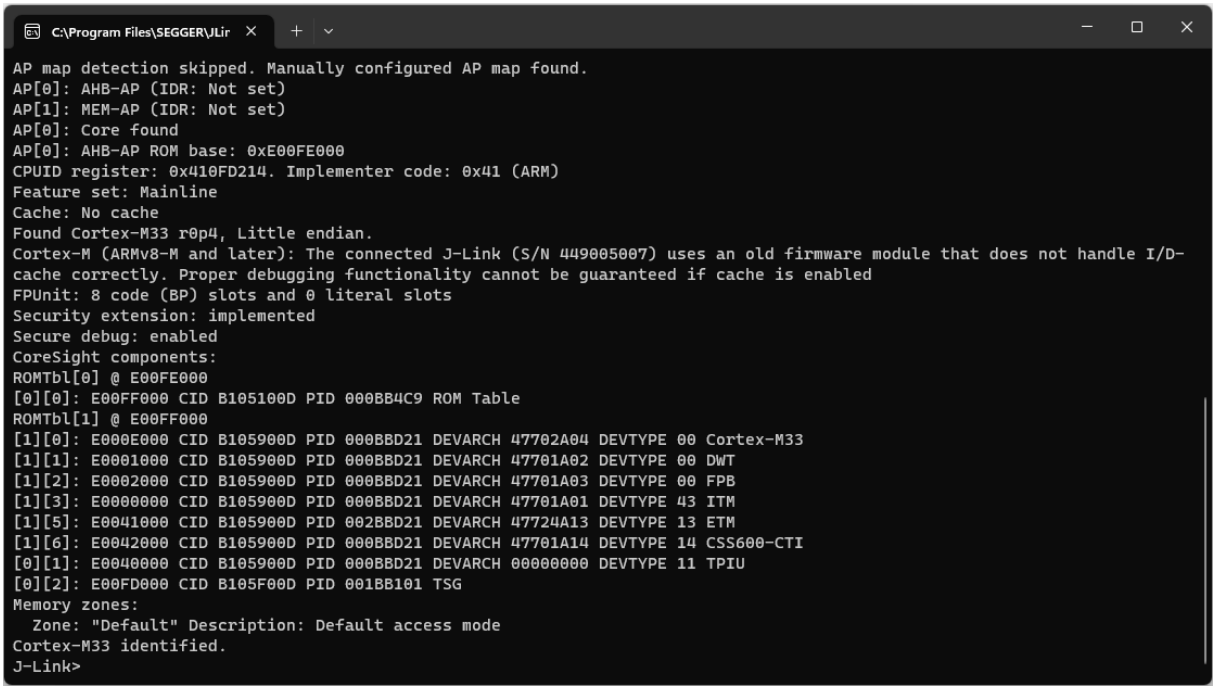


Figure 22: SEGGER J-Link Commander successfully connected to the Lyra DVK.

8. If possible, it is highly recommended to first erase all flash sectors with the **erase** command:

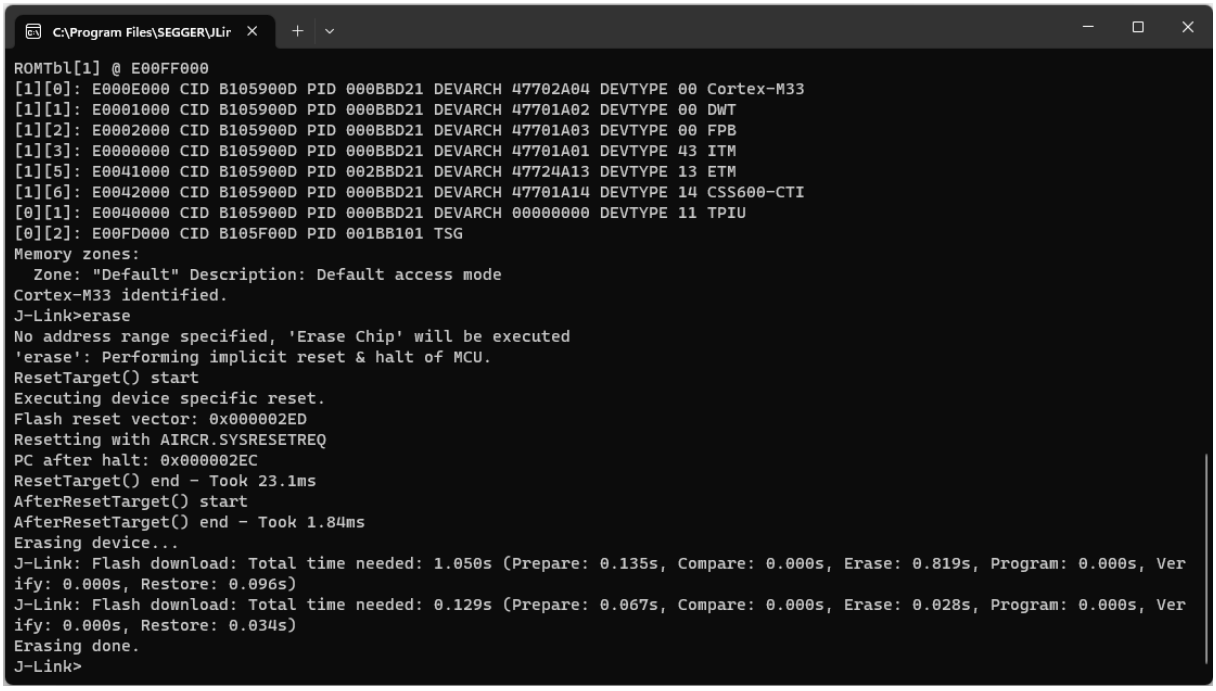


Figure 23: Erasing all flash sectors via SEGGER J-Link Commander …

9. Let's start programming the bootloader image by using the **loadfile <filepath>** command. Please make sure to change the *<filepath>* accordingly. In our example we use the **loadfile "C:\Users\<USERNAME>\Downloads\Lyra 22\LYRA-P_Bootloader_480-00184-R126.1.0.3.hex"** command.

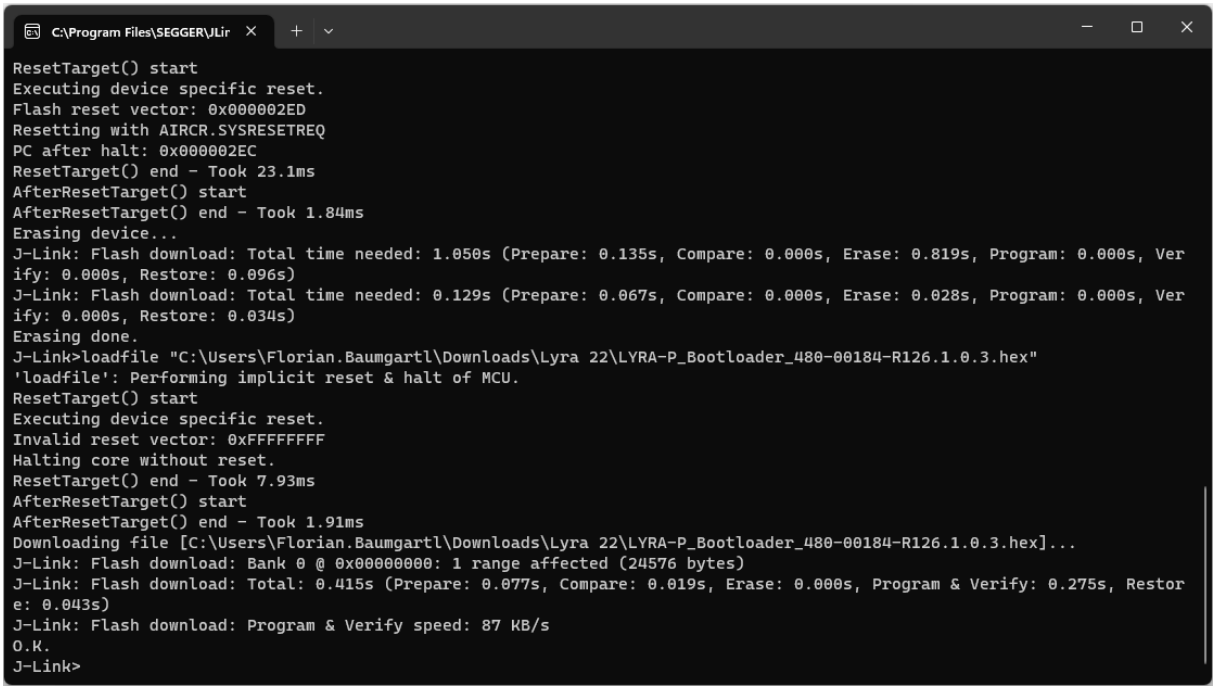


Figure 24: Flashing bootloader image via SEGGER J-Link Commander ...

10. Continue with programming the main application image if the previous command responded with a *O.K.* message. You can use the same **loadfile <filepath>** command by simply replacing *<filepath>* with your main application image path. In this example **loadfile "C:\Users\<USERNAME>\Downloads\Lyra 22\LYRA-P_AT_Interface_480-00182-R126.3.1.387.hex"** is used.

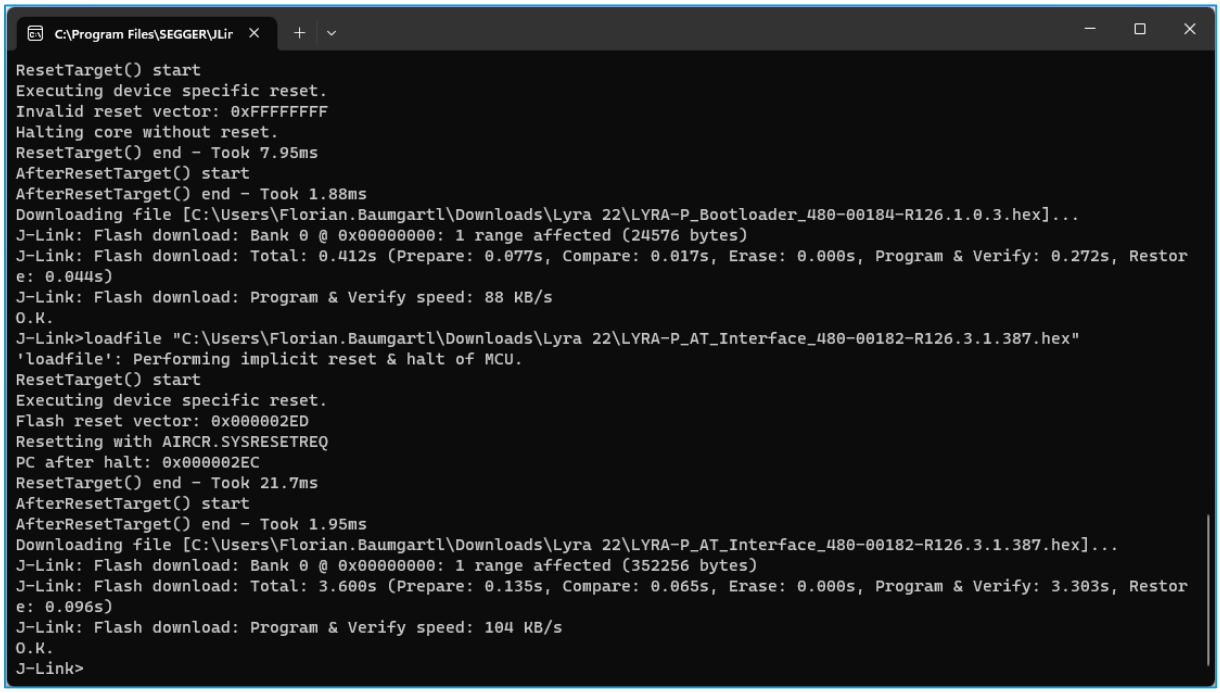


Figure 25: Flashing main application image via SEGGER J-Link Commander ...

11. Please make sure to reset or power cycle the Lyra Series – Development Kit (by pressing the **RST** button) once both commands responded with a *O.K.* message, meaning that the programming completed successfully.

6.4.1 Lyra (22) Example – J-Link Commander with User Commands and Output

User entered commands in this example are **highlighted**:

```
SEGGER J-Link Commander V7.88n (Compiled Jul 26 2023 15:40:02)
DLL version V7.88n, compiled Jul 26 2023 15:38:22

Connecting to J-Link via USB...O.K.
Firmware: Silicon Labs J-Link OB compiled Mar 11 2021 14:26:57
Hardware version: V1.00
J-Link uptime (since boot): N/A (Not supported by this model)
S/N: 449005007
License(s): RDI, FlashBP
VTref=3.349V


Type "connect" to establish a target connection, '?' for help
J-Link>connect
Please specify device / core. <Default>: EFR32BG22CXXXF512
Type '?' for selection dialog
Device>EFR32BG22CXXXF512
Please specify target interface:
  J) JTAG (Default)
  S) SWD
  T) cJTAG
TIF>S
Specify target interface speed [kHz]. <Default>: 4000 kHz
Speed>4000
Device "EFR32BG22CXXXF512" selected.


Connecting to target via SWD
ConfigTargetSettings() start
ConfigTargetSettings() end - Took 4us
```

```
InitTarget() start
SWD selected. Executing JTAG -> SWD switching sequence.
InitTarget() end - Took 54.8ms
Found SW-DP with ID 0x6BA02477
DPv0 detected
CoreSight SoC-400 or earlier
AP map detection skipped. Manually configured AP map found.
AP[0]: AHB-AP (IDR: Not set)
AP[1]: MEM-AP (IDR: Not set)
AP[0]: Core found
AP[0]: AHB-AP ROM base: 0xE00FE000
CPUID register: 0x410FD214. Implementer code: 0x41 (ARM)
Feature set: Mainline
Cache: No cache
Found Cortex-M33 r0p4, Little endian.
Cortex-M (ARMv8-M and later): The connected J-Link (S/N 449005007) uses an old firmware module that does not
handle I/D-cache correctly. Proper debugging functionality cannot be guaranteed if cache is enabled
FPUnit: 8 code (BP) slots and 0 literal slots
Security extension: implemented
Secure debug: enabled
CoreSight components:
ROMTbl[0] @ E00FE000
[0][0]: E00FF000 CID B105100D PID 000BB4C9 ROM Table
ROMTbl[1] @ E00FF000
[1][0]: E000E000 CID B105900D PID 000BBD21 DEVARCH 47702A04 DEVTYPE 00 Cortex-M33
[1][1]: E0001000 CID B105900D PID 000BBD21 DEVARCH 47701A02 DEVTYPE 00 DWT
[1][2]: E0002000 CID B105900D PID 000BBD21 DEVARCH 47701A03 DEVTYPE 00 FPB
[1][3]: E0000000 CID B105900D PID 000BBD21 DEVARCH 47701A01 DEVTYPE 43 ITM
[1][5]: E0041000 CID B105900D PID 002BBD21 DEVARCH 47724A13 DEVTYPE 13 ETM
[1][6]: E0042000 CID B105900D PID 000BBD21 DEVARCH 47701A14 DEVTYPE 14 CSS600-CTI
[0][1]: E0040000 CID B105900D PID 000BBD21 DEVARCH 00000000 DEVTYPE 11 TPIU
[0][2]: E00FD000 CID B105F00D PID 001BB101 TSG
Memory zones:
  Zone: "Default" Description: Default access mode
Cortex-M33 identified.
J-Link>erase
No address range specified, 'Erase Chip' will be executed
'erase': Performing implicit reset & halt of MCU.
ResetTarget() start
Executing device specific reset.
Flash reset vector: 0x000002ED
Resetting with AIRCR.SYSRESETREQ
PC after halt: 0x000002EC
ResetTarget() end - Took 23.3ms
AfterResetTarget() start
AfterResetTarget() end - Took 2.01ms
Erasing device...
J-Link: Flash download: Total time needed: 1.062s (Prepare: 0.138s, Compare: 0.000s, Erase: 0.826s, Program:
0.000s, Verify: 0.000s, Restore: 0.097s)
J-Link: Flash download: Total time needed: 0.126s (Prepare: 0.067s, Compare: 0.000s, Erase: 0.025s, Program:
0.000s, Verify: 0.000s, Restore: 0.033s)
Erasing done.
J-Link>loadfile "C:\Users\Florian.Baumgartl\Downloads\Lyra 22\LYRA-P_Bootloader_480-00184-R126.1.0.3.hex"
'loadfile': Performing implicit reset & halt of MCU.
ResetTarget() start
Executing device specific reset.
Invalid reset vector: 0xFFFFFFFF
Halting core without reset.
ResetTarget() end - Took 7.07ms
AfterResetTarget() start
AfterResetTarget() end - Took 1.86ms
Downloading file [C:\Users\Florian.Baumgartl\Downloads\Lyra 22\LYRA-P_Bootloader_480-00184-R126.1.0.3.hex]...
J-Link: Flash download: Bank 0 @ 0x00000000: 1 range affected (24576 bytes)
J-Link: Flash download: Total: 0.403s (Prepare: 0.077s, Compare: 0.012s, Erase: 0.000s, Program & Verify:
0.271s, Restore: 0.043s)
J-Link: Flash download: Program & Verify speed: 88 KB/s
O.K.
J-Link>loadfile "C:\Users\Florian.Baumgartl\Downloads\Lyra 22\LYRA-P_AT_Interface_480-00182-R126.3.1.387.hex"
'loadfile': Performing implicit reset & halt of MCU.
ResetTarget() start
Executing device specific reset.
Flash reset vector: 0x000002ED
Resetting with AIRCR.SYSRESETREQ
```

```
PC after halt: 0x000002EC
ResetTarget() end - Took 21.2ms
AfterResetTarget() start
AfterResetTarget() end - Took 1.94ms
Downloading file [C:\Users\Florian.Baumgartl\Downloads\Lyra 22\LYRA-P_AT_Interface_480-00182-
R126.3.1.387.hex]...
J-Link: Flash download: Bank 0 @ 0x00000000: 1 range affected (352256 bytes)
J-Link: Flash download: Total: 3.588s (Prepare: 0.134s, Compare: 0.057s, Erase: 0.000s, Program & Verify:
3.299s, Restore: 0.096s)
J-Link: Flash download: Program & Verify speed: 104 KB/s
O.K.
J-Link>
```

6.4.2 Lyra 24 Example – J-Link Commander with User Commands and Output

User entered commands in this example are **highlighted**:

```
SEGGER J-Link Commander V7.88n (Compiled Jul 26 2023 15:40:02)
DLL version V7.88n, compiled Jul 26 2023 15:38:22

Connecting to J-Link via USB...O.K.
Firmware: Silicon Labs J-Link OB compiled Mar 11 2021 14:26:57
Hardware version: V1.00
J-Link uptime (since boot): N/A (Not supported by this model)
S/N: 449006017
License(s): RDI, FlashBP
VTref=3.352V


Type "connect" to establish a target connection, '?' for help
J-Link>connect
Please specify device / core. <Default>: EFR32BG24BXXXF1536
Type '?' for selection dialog
Device>EFR32BG24BXXXF1536
Please specify target interface:
  J) JTAG (Default)
  S) SWD
  T) cJTAG
TIF>S
Specify target interface speed [kHz]. <Default>: 4000 kHz
Speed>4000
Device "EFR32BG24BXXXF1536" selected.


Connecting to target via SWD
ConfigTargetSettings() start
ConfigTargetSettings() end - Took 4us
InitTarget() start
SWD selected. Executing JTAG -> SWD switching sequence.
InitTarget() end - Took 67.5ms
Found SW-DP with ID 0x6BA02477
DPv0 detected
CoreSight SoC-400 or earlier
AP map detection skipped. Manually configured AP map found.
AP[0]: AHB-AP (IDR: Not set)
AP[1]: MEM-AP (IDR: Not set)
AP[0]: Core found
AP[0]: AHB-AP ROM base: 0xE00FE000
CPUID register: 0x410FD214. Implementer code: 0x41 (ARM)
Feature set: Mainline
Cache: No cache
Found Cortex-M33 r0p4, Little endian.
Cortex-M (ARMv8-M and later): The connected J-Link (S/N 449006017) uses an old firmware module that does not
handle I/D-cache correctly. Proper debugging functionality cannot be guaranteed if cache is enabled
FPUnit: 8 code (BP) slots and 0 literal slots
Security extension: implemented
Secure debug: enabled
CoreSight components:
ROMTbl[0] @ E00FE000
[0][0]: E00FF000 CID B105100D PID 000BB4C9 ROM Table
ROMTbl[1] @ E00FF000
[1][0]: E000E000 CID B105900D PID 000BBD21 DEVARCH 47702A04 DEVTYPE 00 Cortex-M33
[1][1]: E0001000 CID B105900D PID 000BBD21 DEVARCH 47701A02 DEVTYPE 00 DWT
```

```
[1][2]: E0002000 CID B105900D PID 000BBD21 DEVARCH 47701A03 DEVTYPE 00 FPB
[1][3]: E0000000 CID B105900D PID 000BBD21 DEVARCH 47701A01 DEVTYPE 43 ITM
[1][5]: E0041000 CID B105900D PID 002BBD21 DEVARCH 47724A13 DEVTYPE 13 ETM
[1][6]: E0042000 CID B105900D PID 000BBD21 DEVARCH 47701A14 DEVTYPE 14 CSS600-CTI
[0][1]: E0040000 CID B105900D PID 000BBD21 DEVARCH 00000000 DEVTYPE 11 TPIU
[0][2]: E00FD000 CID B105F00D PID 001BB101 TSG
Memory zones:
  Zone: "Default" Description: Default access mode
Cortex-M33 identified.
J-Link>erase
No address range specified, 'Erase Chip' will be executed
'erase': Performing implicit reset & halt of MCU.
ResetTarget() start
Executing device specific reset.
Flash reset vector: 0x08000391
Resetting with AIRCR.SYSRESETREQ
PC after halt: 0x08000390
ResetTarget() end - Took 35.7ms
AfterResetTarget() start
AfterResetTarget() end - Took 1.90ms
Erasing device...
J-Link: Flash download: Total time needed: 2.724s (Prepare: 0.134s, Compare: 0.000s, Erase: 2.493s, Program:
0.000s, Verify: 0.000s, Restore: 0.096s)
J-Link: Flash download: Total time needed: 0.127s (Prepare: 0.070s, Compare: 0.000s, Erase: 0.022s, Program:
0.000s, Verify: 0.000s, Restore: 0.034s)
Erasing done.
J-Link>loadfile "C:\Users\Florian.Baumgartl\Downloads\Lyra 24\LYRA24-P20RF_Bootloader_480-00288-R133.4.0.6.hex"
'loadfile': Performing implicit reset & halt of MCU.
ResetTarget() start
Executing device specific reset.
Invalid reset vector: 0xFFFFFFFF
Halting core without reset.
ResetTarget() end - Took 7.50ms
AfterResetTarget() start
AfterResetTarget() end - Took 1.83ms
Downloading file [C:\Users\Florian.Baumgartl\Downloads\Lyra 24\LYRA24-P20RF_Bootloader_480-00288-
R133.4.0.6.hex]...
J-Link: Flash download: Bank 0 @ 0x08000000: 1 range affected (73728 bytes)
J-Link: Flash download: Total: 0.936s (Prepare: 0.095s, Compare: 0.016s, Erase: 0.000s, Program & Verify:
0.762s, Restore: 0.060s)
J-Link: Flash download: Program & Verify speed: 94 KB/s
O.K.
J-Link>loadfile "C:\Users\Florian.Baumgartl\Downloads\Lyra 24\LYRA24-P20RF_AT_Interface_480-00289-
R133.5.1.16.hex"
'loadfile': Performing implicit reset & halt of MCU.
ResetTarget() start
Executing device specific reset.
Flash reset vector: 0x08000391
Resetting with AIRCR.SYSRESETREQ
PC after halt: 0x08000390
ResetTarget() end - Took 34.1ms
AfterResetTarget() start
AfterResetTarget() end - Took 1.88ms
Downloading file [C:\Users\Florian.Baumgartl\Downloads\Lyra 24\LYRA24-P20RF_AT_Interface_480-00289-
R133.5.1.16.hex]...
J-Link: Flash download: Bank 0 @ 0x08000000: 1 range affected (303104 bytes)
J-Link: Flash download: Total: 3.151s (Prepare: 0.133s, Compare: 0.046s, Erase: 0.000s, Program & Verify:
2.874s, Restore: 0.096s)
J-Link: Flash download: Program & Verify speed: 103 KB/s
O.K.
J-Link>
```

7 Further Information

Further information and resources (i*ncluding Firmware, Certificates, Reports, Software, Application Notes, User Guide and more*) relating to the Lyra Series module is available on our product page of the Ezurio website under:

http://www.ezurio.com/lyra-series

http://www.ezurio.com/lyra24-series

8 References

Ref	Details
[A]	Lyra (22) Firmware GitHub https://github.com/LairdCP/Lyra_Firmware
[B]	Lyra (22) Firmware GitHub (Releases) https://github.com/LairdCP/Lyra_Firmware/releases
[C]	Lyra 24 Firmware GitHub https://github.com/LairdCP/Lyra_24_Firmware
[D]	Lyra 24 Firmware GitHub (Releases) https://github.com/LairdCP/Lyra_24_Firmware/releases

9 Additional Information

Please contact your local sales representative or our support team for further assistance:

Headquarters	Ezurio 50 S. Main St. Suite 1100 Akron, OH 44308 USA
Website	http://www.ezurio.com
Technical Support	http://www.ezurio.com/resources/support
Sales Contact	http://www.ezurio.com/contact

Note: Information contained in this document is subject to change.

Ezurio's products are subject to standard Terms & Conditions.