

# Building the Secure Bootloader Project for your Device

The Secure Bootloader Plus project should build and run in your device as delivered by Driven 2 Design from your order specification. The source code is provided so that you may make your own modifications as required.

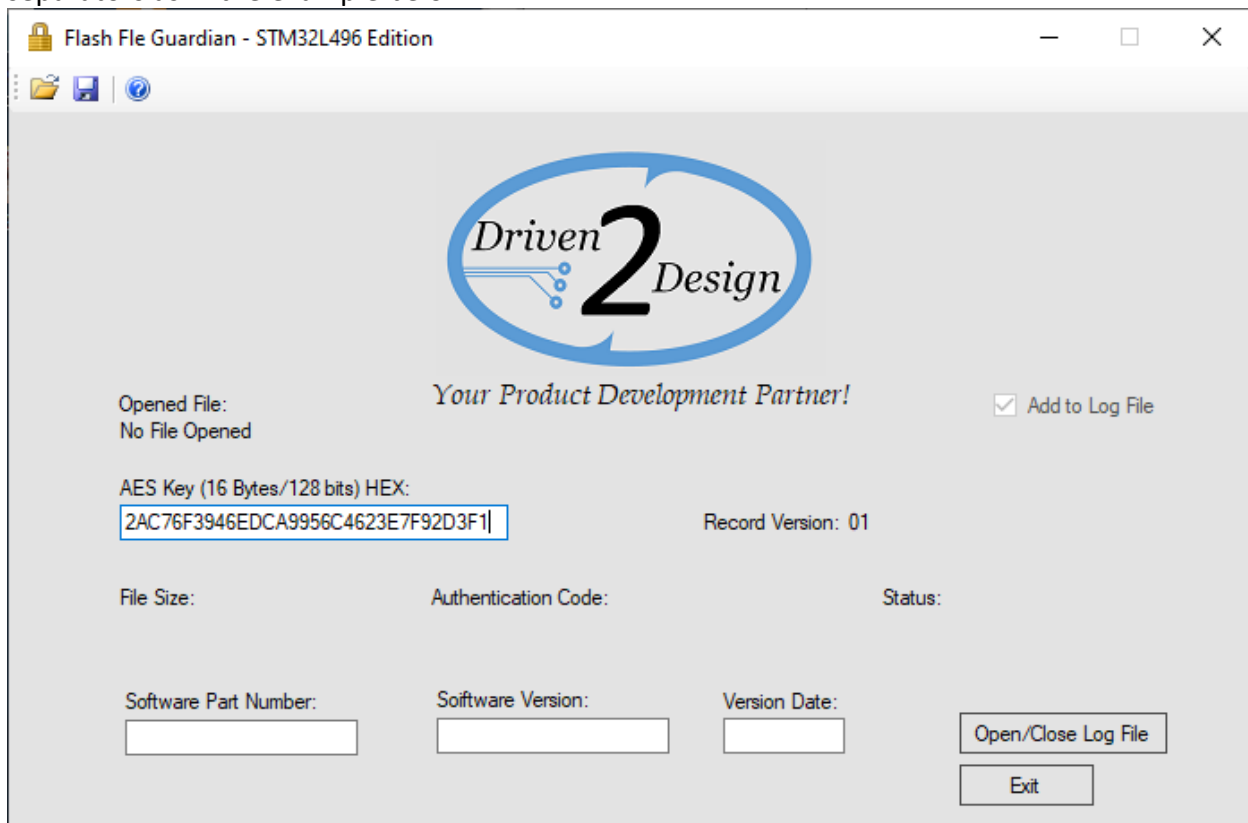
The project is delivered as “SBLp/SBLp XXX xxxxxx.nnn”. The top SBLp directory is the workspace and SBLp XXX xxxxxx.nnn is the project directory. To open the project in CubeIDE double click the CubeIDE Icon. Click the “Browse...” button and navigate to the **/SBLp/** directory then click the “Launch” button. CubeIDE should come up with the project “**SBLp XXX xxxxxx.nnn**” available for opening in the project navigation window on the left of the IDE.

The only item that the user is required to edit is the AES encryption key. This key is a 16 byte (128 bit) value and is defined in the “**Private Macros**” section of Update.h found at “ProjDir/BLApp/Inc”

It is the user’s responsibility to select, define and maintain security of this key value. In Update.h this key is defined as a string of 16 byte values as below. You will need to edit these byte values to the values of your own choosing.

```
// 128 bit AES encryption key as delivered by Driven 2 Design
#define ENCRYPTION_KEY    0xD2, 0xD0, 0xD2, 0xD1, 0xD2, 0xD2, 0xD2, 0xD3, \
                          0xD2, 0xD4, 0xD2, 0xD5, 0xD2, 0xD6, 0xD2, 0xD7
```

When securing your update binary file with Flash File Guardian the key you selected for the bootloader is entered into the AES Key window of Flash File Guardian as a string of hexadecimal characters “2AC76F3946EDCA9956C4623E7F92D3F1” without the C syntax of “0x” or comma separators as in the example below.



Once a key has been used to secure a file (create an .ffg file) you may open the log file and perform a copy and paste of the AES key from the log file instead of retyping all 32 characters again. The AES Key is saved in the log file along with other data such as date, firmware part number and version number. See the Log File Example Page.

Coming out of every reset the bootloader will always be booted and get control of your board. This ensures that your board may always be updated, even if an update that crashes the system is inadvertently released. For this reason you will need to edit the bootloader project's .ioc file in order to ensure your systems startup I/O requirements. This is performed in the bootloader's project .ioc file using ST's STM32CubeIDE tool. The bootloader is not delivered to build under any other tool. It is however a very simple effort to convert the bootloader project to other tools supported by STM32CubeMX but this is not recommended. Your product application code may be developed in the tool of your choice. See the **Bootloader Readme.txt** file for CubeIDE and low level library version information.

It may also be necessary to edit the clock configuration for your board but shouldn't be. Clock configuration data is normally received at order time and then configured for you prior to product delivery.

**\*\*\* CAUTION \*\*\***

When using CubeIDE, from time to time it will make you aware of updates to the low level libraries at startup and ask if you wish to Continue or Migrate. Unless you have your current bootloader project backed up **DO NOT SELECT MIGRATE**.

Selecting Migrate will cause CubeIDE to download and install the new low level libraries for the project's MCU and then regenerate the low level code for the bootloader project with the new library. In the past this often has had negative consequences and the build failed to execute properly in the MCU. To be safe always select **Continue**. If you wish to migrate to the latest firmware package first make a backup of the bootloader project and store it safely somewhere else.

If you have converted the bootloader project to IAR or any other tool then you must use STM32CubeMX to make edits to the bootloader project's .ioc file. The same situation described above for CubeIDE applies to CubeMX. **Selecting Migrate can have negative consequences.**

Example Flash File Guardian Log File

FLASH File Guardian Log File v1.0.0

Created: 01/02/2020 10:04 AM

Creation Date:	Creation Time:	File Name:	Part Number:	Version Number:	Version Date:	Authentication Code:	AES Key:
01/05/2020	05:08 PM	D2D-8726523 v001.000.ffg	D2D-8726523	001.000	01/05/2020	32AB7878	2AC76F3946EDCA9956C4623E7F92D3F1
01/18/2020	05:22 PM	D2D-8726523 v002.000.ffg	D2D-8726523	002.000	01/18/2020	F93C1373	2AC76F3946EDCA9956C4623E7F92D3F1