Community

Search all content... Ctrl + K

STM32 MCUs



B.Montanari
ST Technical Moderator · 1 year ago

How to create an STM32N6 FSBL load and run

23 replies

Summary

The first stage bootloader (FSBL) is a key component in the boot process of STM32N6 microcontrollers. It is responsible for initializing the system, configuring the hardware, and loading the application code from external memory into the internal or external memories for execution. This article provides a quick tutorial on how to use the FSBL in the load and run mode including the process to program the external memory.

It covers two practical examples, the first being a simple blink LED application running directly from the FSBL. The second being a more useful setup, where the FSBL loads an application from external memory and runs it from the internal RAM.

Introduction

At power-on, the boot ROM copies the FSBL binary from the external memory into the internal SRAM. Once the boot ROM task is completed, it will jump to the FSBL project, which is usually responsible for executing the clock and system settings, and configuring the external memories. Finally, it either copies the application binary in internal SRAM or sets the external memory in memory mode. When done, the application itself starts up and runs. If you want to know more about the boot ROM, check this knowledge article.

On STM32N6 MCUs, the first-stage bootloader (FSBL) must be signed or at least have a valid header, so the boot ROM can execute it in a secured-locked state. The FSBL layout includes several key components, and more details are available in this article.

This article uses the STM32N6570-DK as the base for its hands-on portion, but the content can be tailored to any specific STM32N6 hardware. Two examples are shown: the first exam blink LED running directly from the FSBL, and the second example is composed of two bi application and FSBL. Both examples have the header added and programmed on the external memory. It is up to the FSBL to execute the application directly from the internal RAM, or copy from the exter FLASH to the internal RAM and execute the application.

Feedback

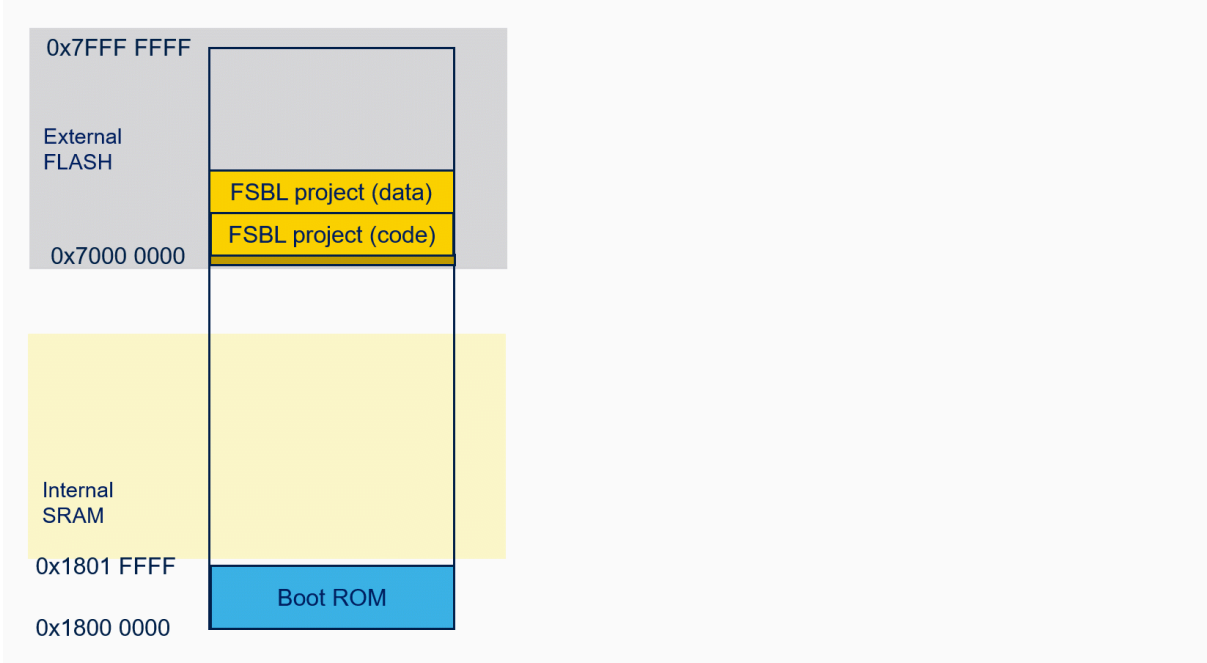
Ask AI

1. FSBL main features

The FSBL can be used in several different ways, each suited to specific application requirements. The following sections describe the two modes that the FSBL is used in this article.

1.1 Application as part of the FSBL

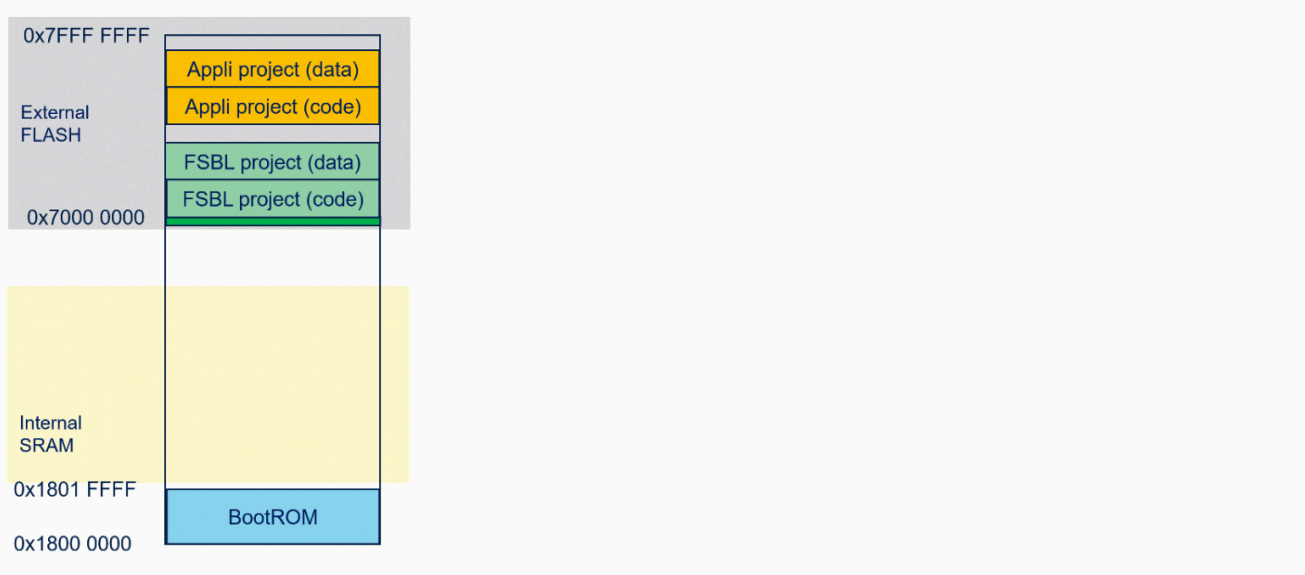
In this mode, the boot ROM fetches the FSBL from external serial NOR flash memory. As the FSBL and application are contained in the same binary, there is a 511 KB size limitation in this mode. This is due to the total area of 512 KB that the boot ROM copies from the external memory into the internal SRAM. The visual representation can be observed in the small animation below:



1.2 FSBL with load & run application

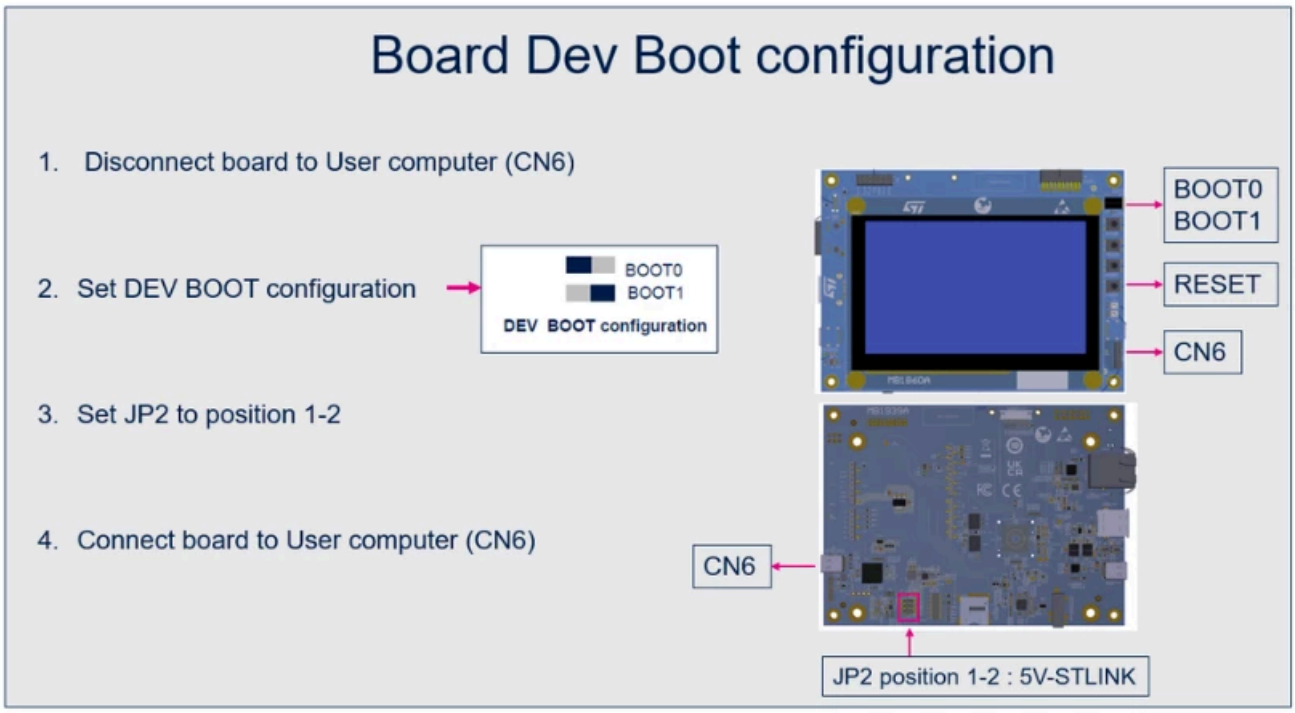
In this mode, the boot ROM fetches the FSBL from external flash memory. This time, the FSBL proceeds to configure the external memory and fetch a second binary stored in it, then copy it into the internal SRAM. Once the binary is loaded, the FSBL jumps to the new position in RAM, where the application code is located and starts the execution. This mode is applicable to a few examples available in the STM32Cube_FW_N6. The interesting aspect is that the 511 KB size limitation is no longer applicable, as the user code can be placed in the remaining area of the internal RAM.

Ask AI



2. Hands-on with blink LED

This article assumes you have installed STM32CubeMX (6.13 or later), the latest version of the STM32N6 HAL driver, STM32CubeProgrammer (2.18 or later), and STM32CubeIDE (1.17.0 or later). The hardware used to showcase is the STM32N6570-DK and make sure you have it in DEV boot mode to program the code:



2.1 First demo with simple FSBL blink code

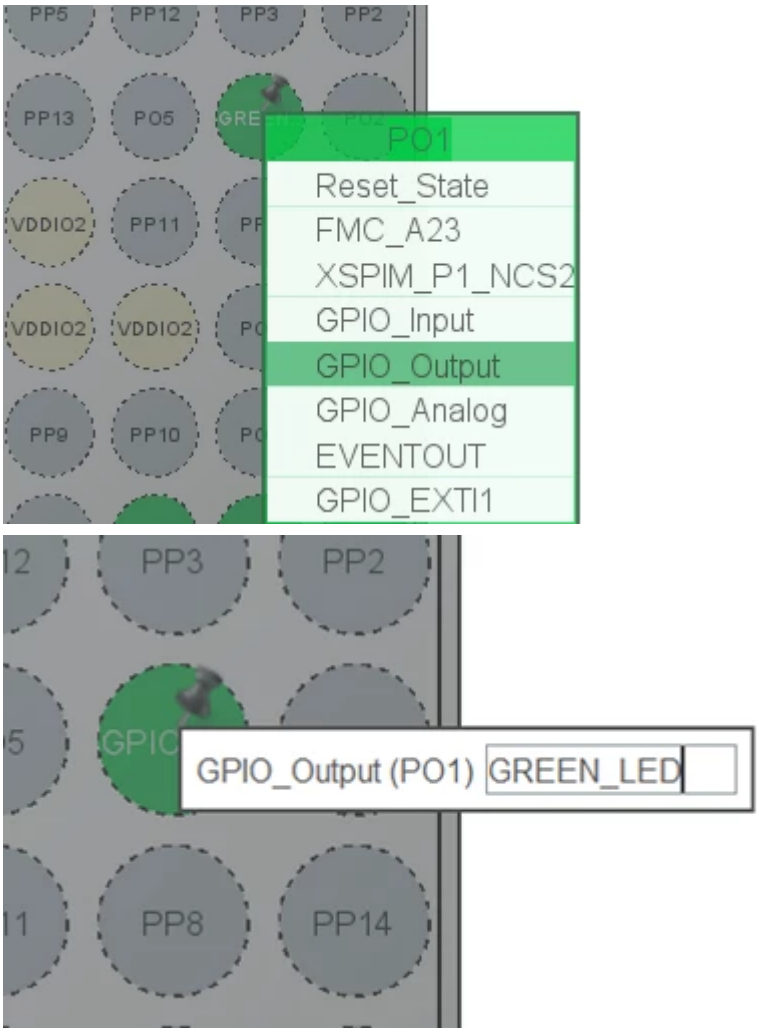
The project needs to configure a few peripherals to properly work, including the green l ted with the PO1 and its active HIGH to validate the first step. Also, the FSBL and applic same binary. So, let us start with this initial configuration first.

Create a new project using the STM32CubeMX and select the **[STM32N657X0H3Q]**. Select the op use the **[Secure Domain only]**.

Ask AI

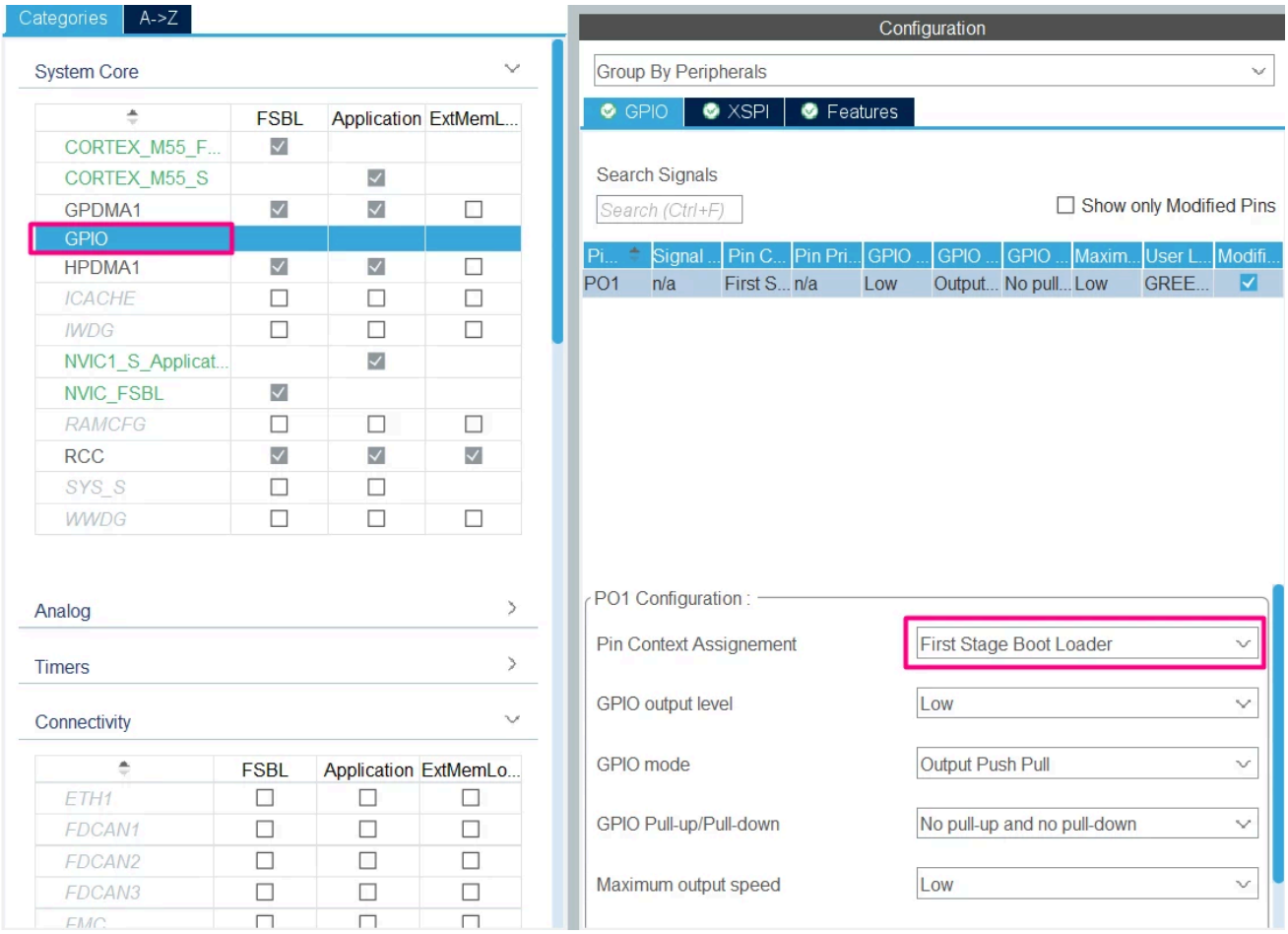
2.1.1 Configure the LED pin

Locate the PO1 and configure it as GPIO_Output and use the label to name it **GREEN_LED**.



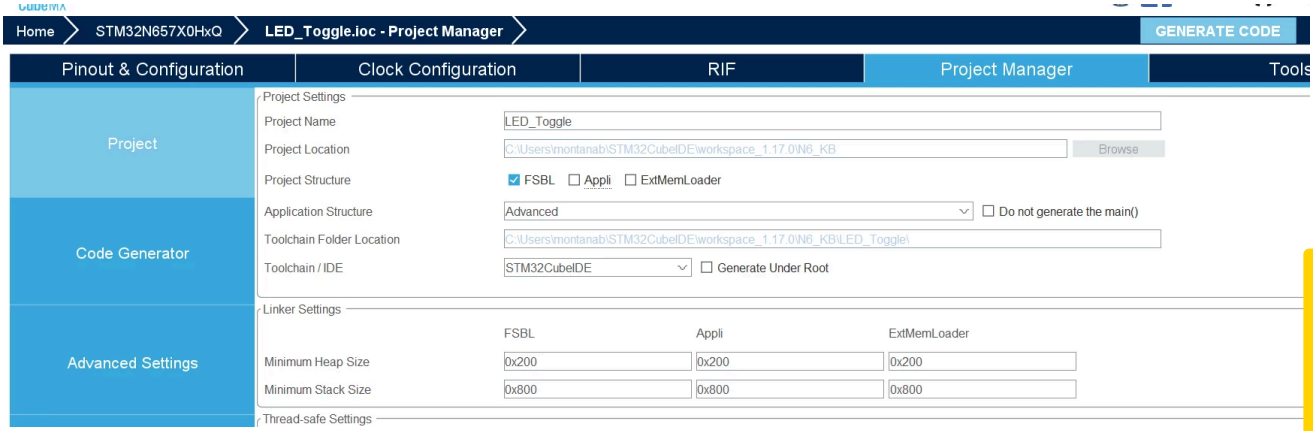
We need to assign the GPIO to be used by the FSBL

Ask AI



2.1.2 Code generation and editing

In the **[Project Manager]** tab, ensure the **[FSBL]** checkbox is selected and generate the code for your preferred toolchain.

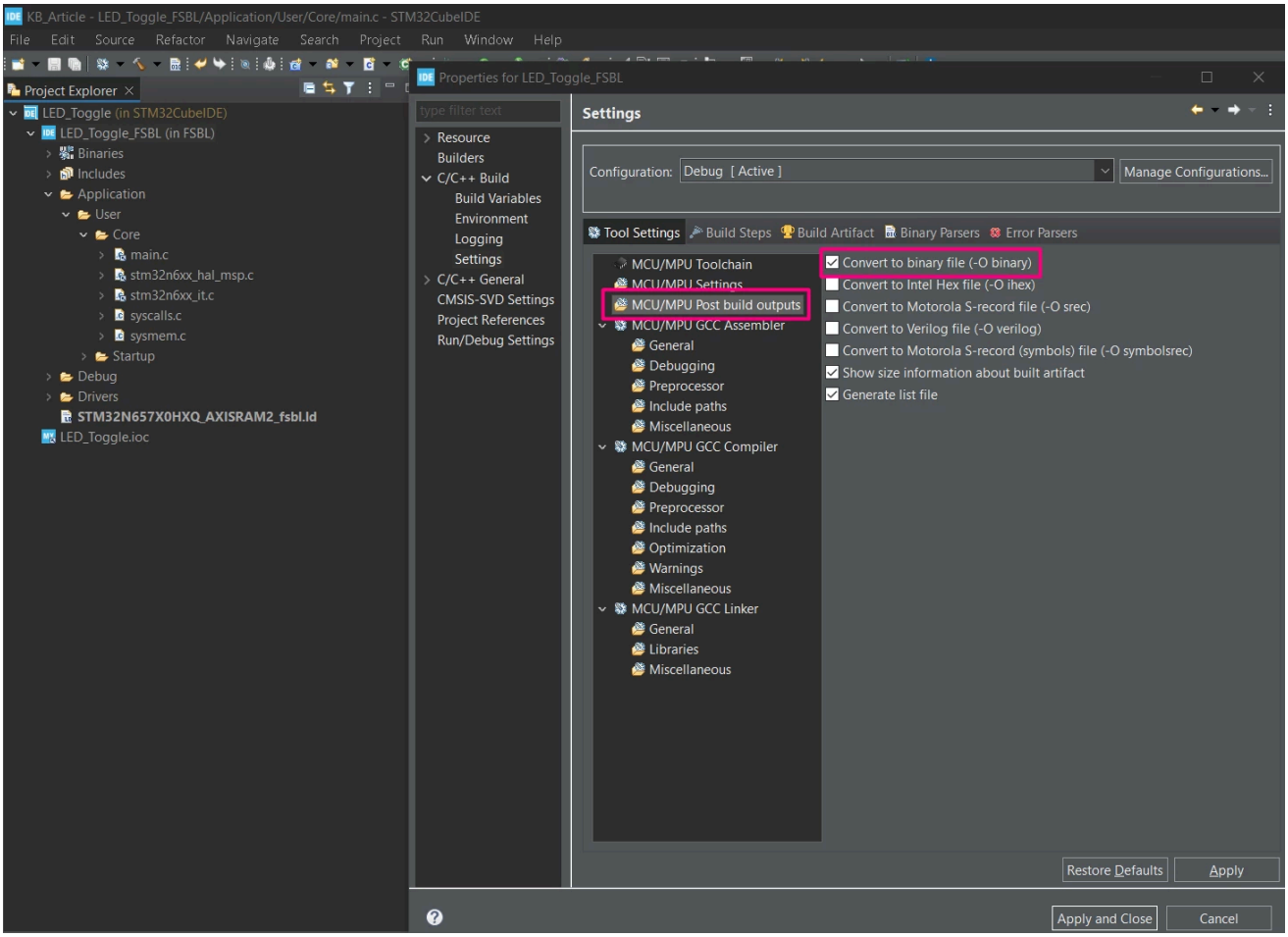


Once the project is created, add these two lines in the main loop:

```
while (1)
{
/* USER CODE END WHILE */
/* USER CODE BEGIN 3 */
HAL_GPIO_TogglePin(GREEN_LED_GPIO_Port, GREEN_LED_Pin);
HAL_Delay(100);
}
/* USER CODE END 3 */
```

Ask AI

Make sure that your project settings are configured to generate the *.bin as well, since we use it to run the scripts.



Enter in debug mode and validate that your code is working. Now that you have, we need to add the FSBL header to ensure that the boot ROM is capable of reading and copying it from the external memory. To do this, we use the STM32CubeProgrammer's CLI. The next assumes the STM32CubeProgrammer was installed in the default path, if not, make sure to adjust it accordingly.

2.1.3 Adding the header

To facilitate the process, its possible to type [cmd] in the binary folder. For example *..\LED_Toggle\STM32CubeIDE\FSBL\Debug*. This will pop up the cmd in the selected path and its possible to use the command line interface from STM32CubeProgrammer and its Signing Tool. Starting from STM32CubeProgrammer version 2.21.0, a new parameter is needed at the end **"-align". Please make sure you use the updated command, as follows**:

```
"C:\Program Files\STMicroelectronics\STM32Cube\STM32CubeProgrammer\bin\STM32_SigningTool_CLI.exe" -bin LED_Toggle_FSBL.bin -nk -of 0x80000000 -t fsbl -o Project-trusted.bin -hv 2.3 -dump Project-trusted.bin -align
```

Ask AI

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. All rights reserved.

C:\Users\montanab\STM32CubeIDE\workspace_1.17.0\N6_KB\LED_Toggle\STM32CubeIDE\FSBL\Debug>"C:\Program Files\STMicroelectr
onics\STM32Cube\STM32CubeProgrammer\bin\STM32_SigningTool_CLI.exe" -bin LED_Toggle_FSBL.bin -nk -of 0x80000000 -t fsbl -
o Project-trusted.bin -hv 2.3 -dump Project-trusted.bin
      ------------------------------------------------------------------
                        STM32 Signing Tool v2.18.0
      ------------------------------------------------------------------

Adding head padding bytes for payload with header v2.3
Header version 2.3 preparation...
Extracting Entry point value from the input file...
Entry point value : 0x34180c11
The headred image file generated successfully: Project-trusted.bin

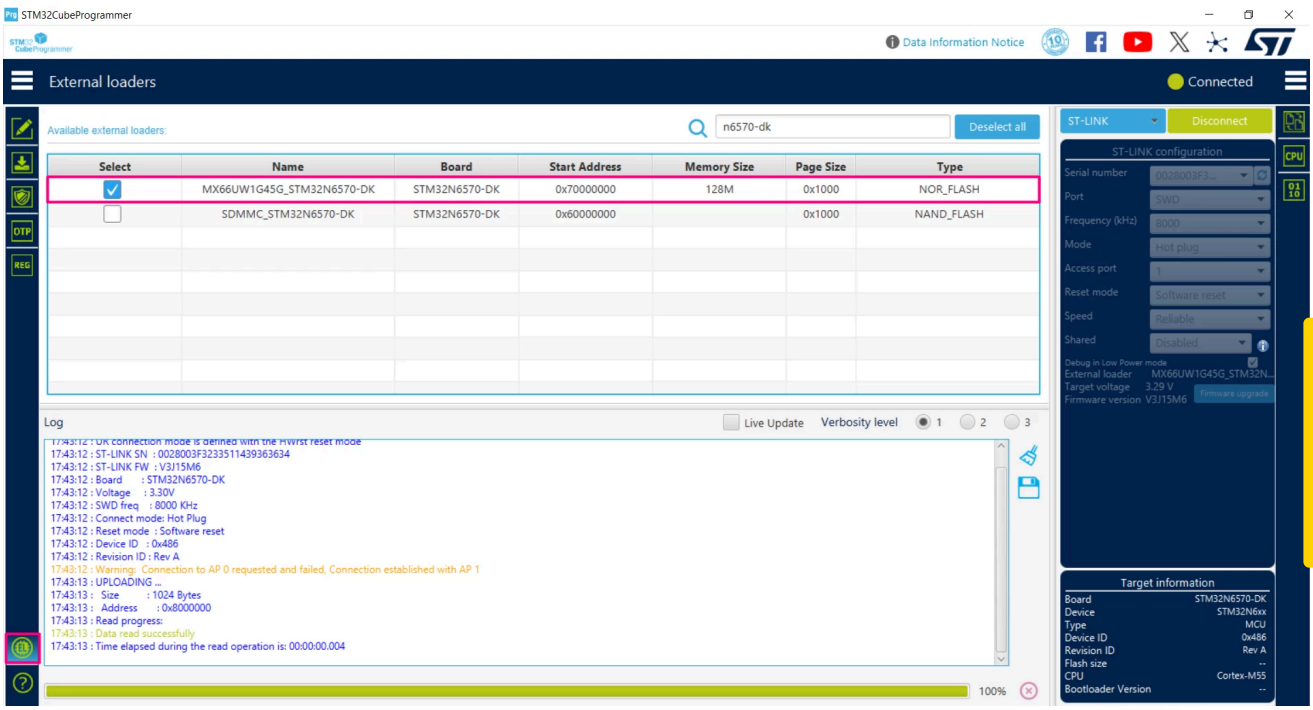
Header description:

    Magic: 0x53544d32
    Signature: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
               00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
               00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
    Checksum: 0x2efb9c
    Header version: 0x20300
    Size: 0x6f20
    Load address: 0xffffffff
    Entry point: 0x34180c11
    Image version: 0x0
    Extension: 0x80000000
```

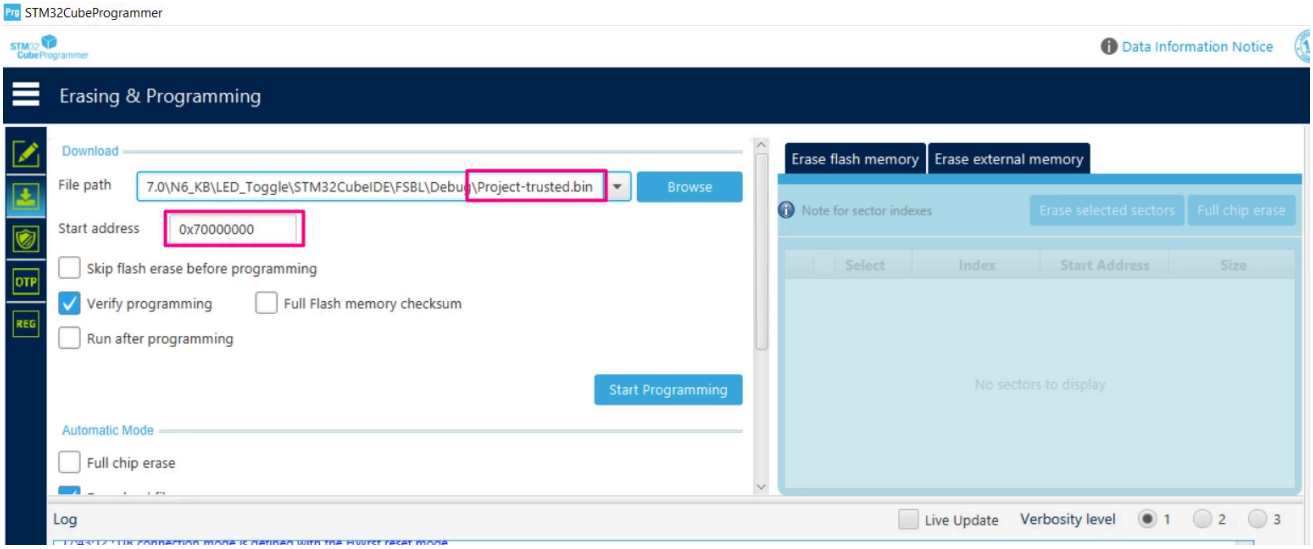
This creates the header for the binary, which can now be loaded into the external FLASH using STM32CubeProgrammer.

2.1.4 Programming the binary

Make sure that the external loader for the flash memory is enabled and use the address [0x7000 0000] to program the binary.



Ask AI



TIP: you might need to power cycle the board in case a failure message appears when clicking [Start Programming].

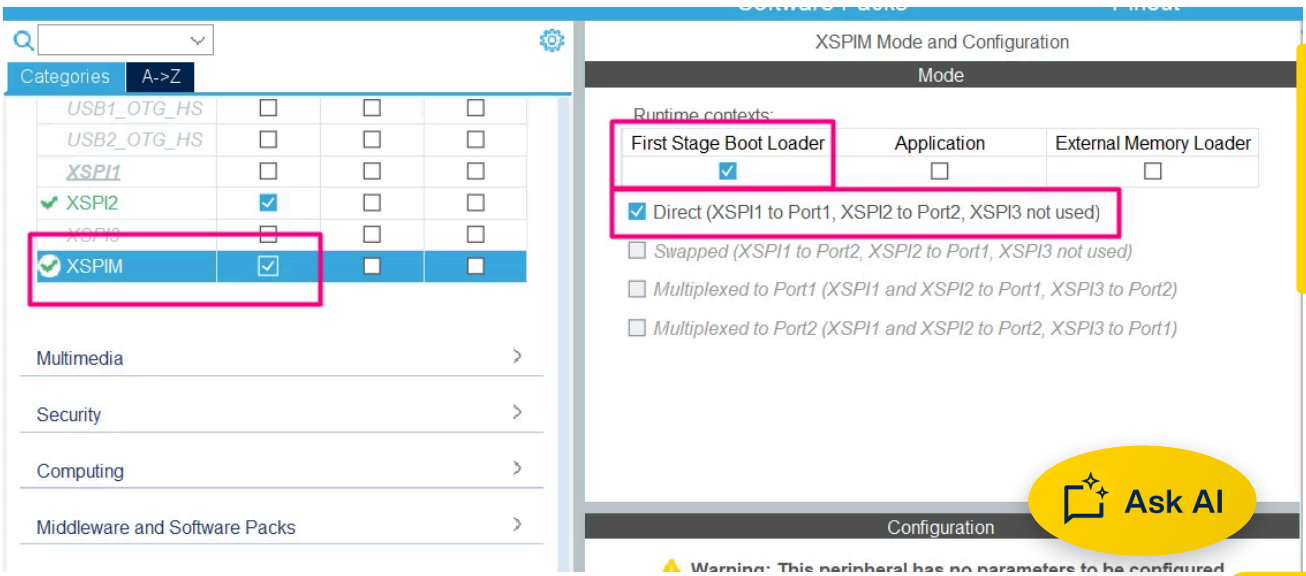
To validate the entire process worked, disconnect from the STM32CubeProgrammer. Then switch BOOT1 to 0 and reset the board. This ensures that the boot ROM uses the external memory and the green LED should blink.

2.2 Second demo with FSBL and application

Now that we concluded the process to run our simple code from the FSBL, it's time to implement the second example. This example consists of having the FSBL to copy the application content from the external FLASH to internal RAM, and executing the code from there.

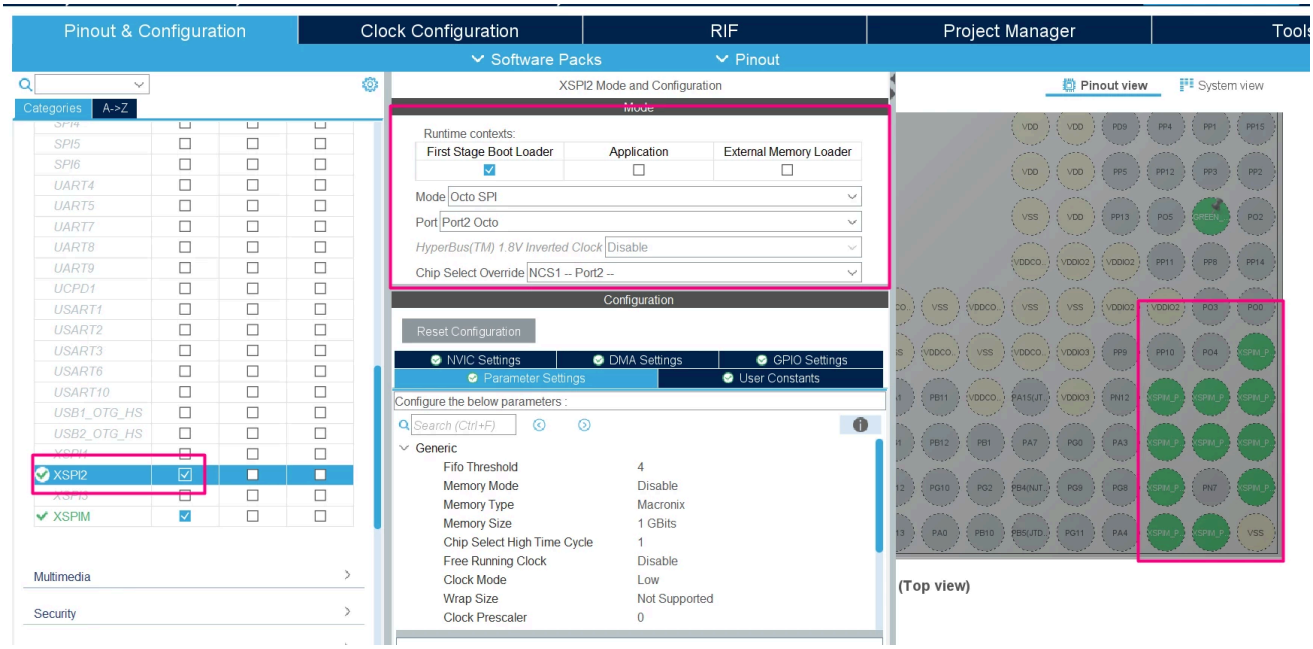
2.2.1 Configure the XSPI and XSPIM

Using the same STM32CubeMX project, locate the [XSPIM] under the [Connectivity] menu on the left and select it to run during the [FSBL] and have the [Direct] mode selected:



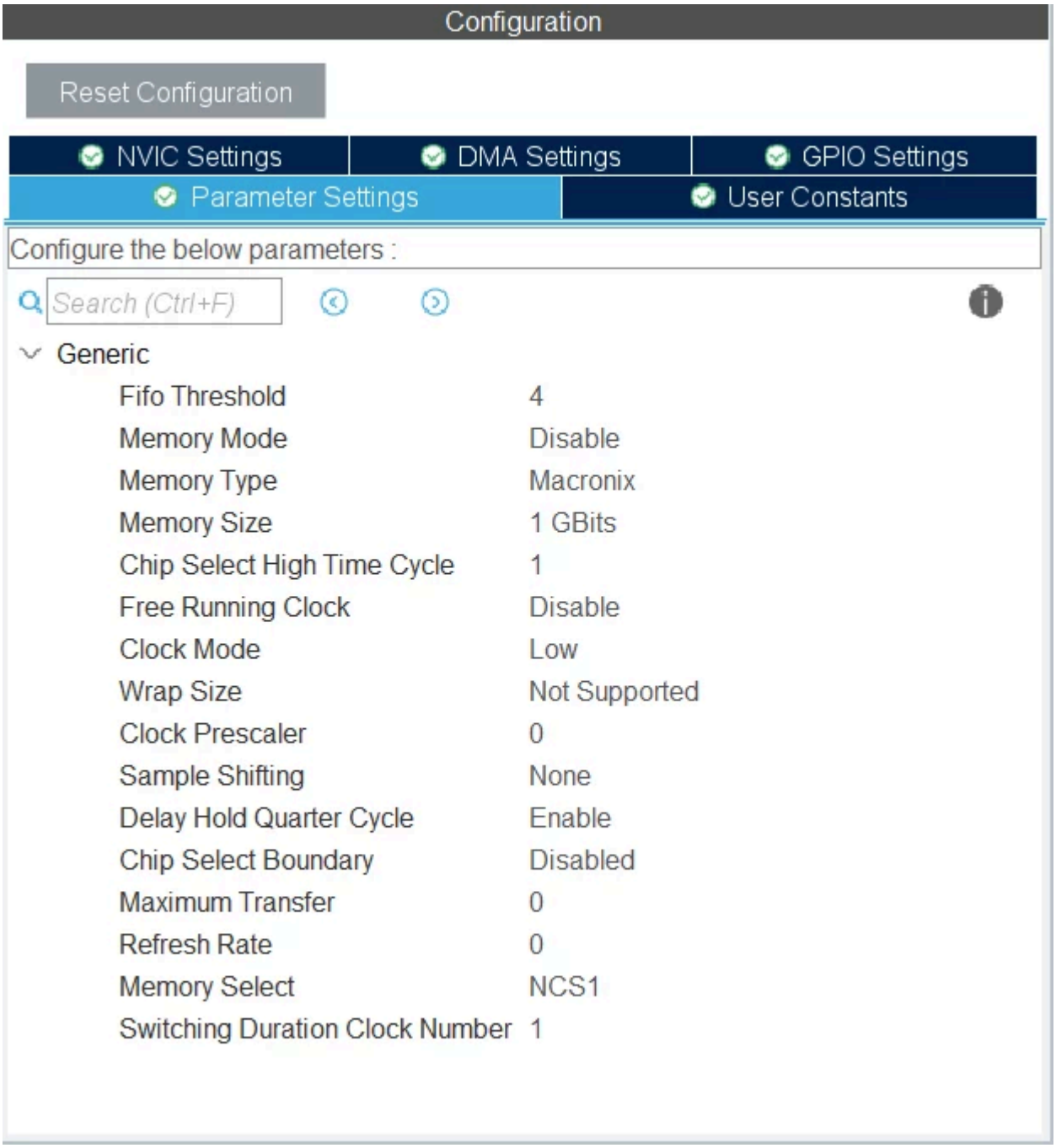
The OCTOSPI flash memory has the following characteristics: 1 Gbit, 1.8 V, 200 MHz, DTR, read w
writing. It is connected to the OCTOSPI interface of the STM32N657X0H3Q microcontroller on the

STM32N6570-DK board on XSPI2. With that information, go to XSPI2 to configure the peripheral according to the hardware available:



As for the **[Parameter Settings]**, look carefully at the image below:

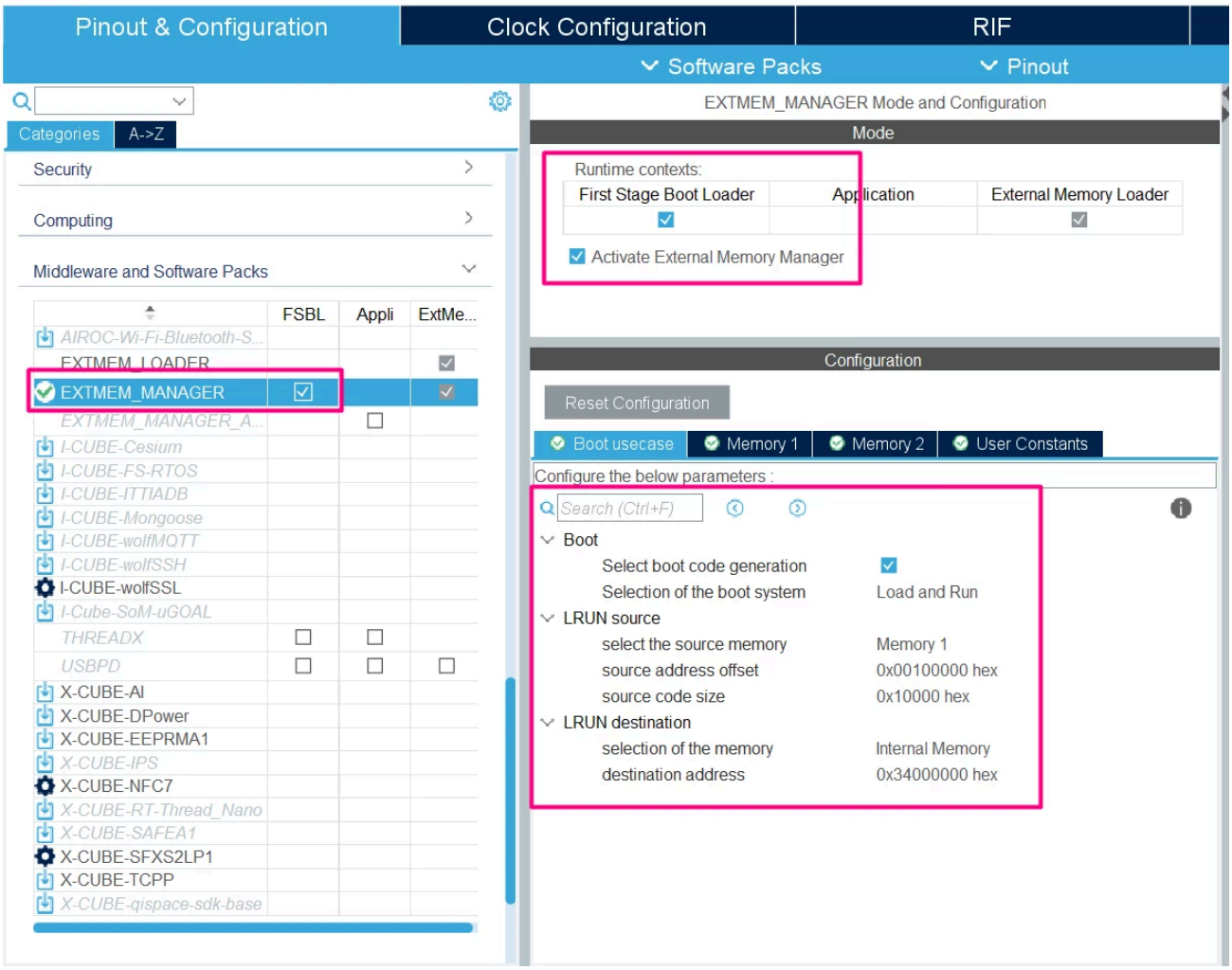
Ask AI



2.2.2 Configure the EXTMEM_MANAGER

The next step is locating the **[Middleware and Software Packs]** in the **[Categories]**. Expand the **[EXTMEM_MANAGER]**. Add the FSBL and activate it using the following settings:

Ask AI



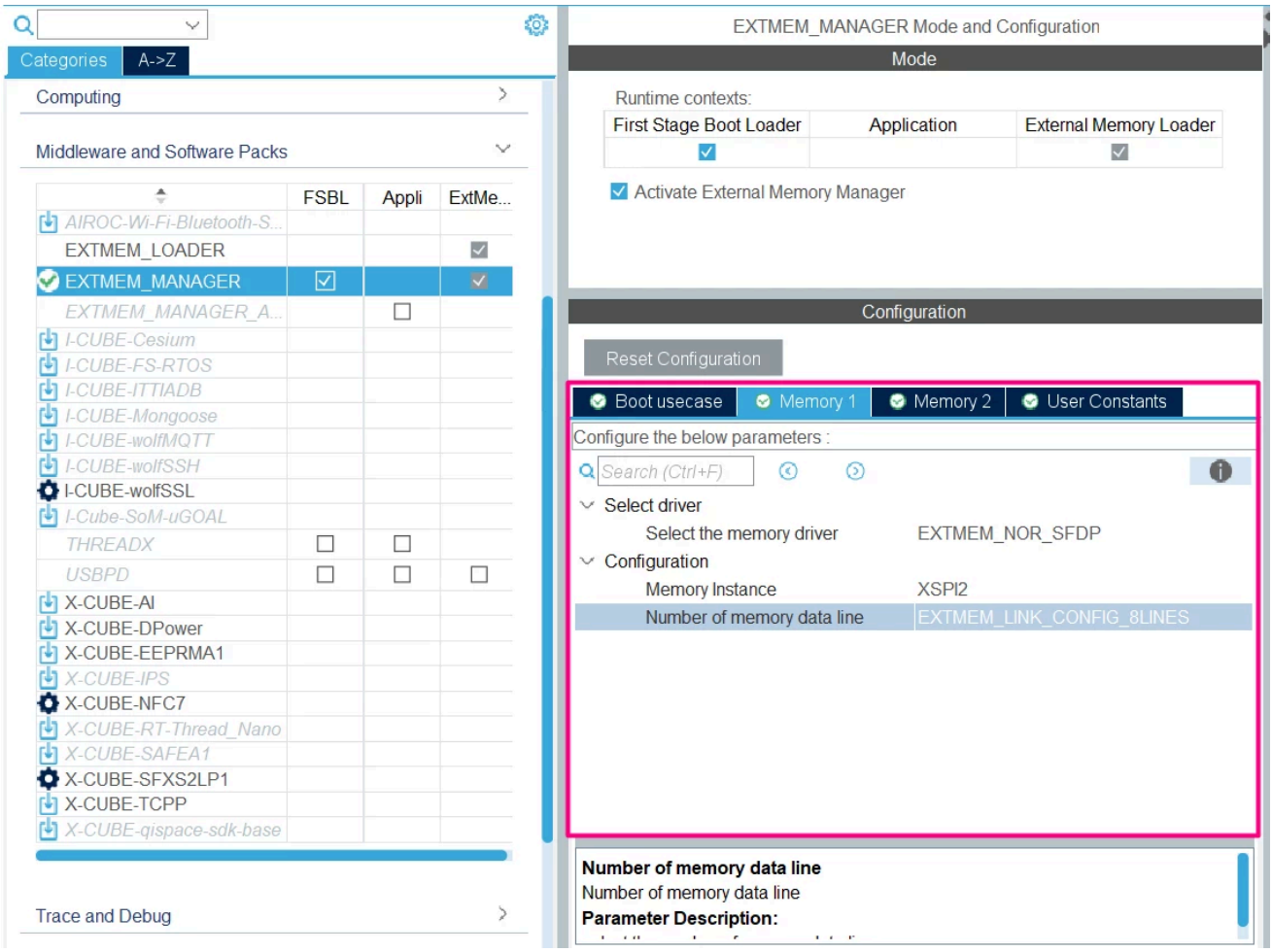
[LRUN source address offset] = 0x00100000

[LRUN source code size] = 0x10000

[LRUN destination address] = 0x34000000

In the **[Memory 1]** tab, ensure its as follows:

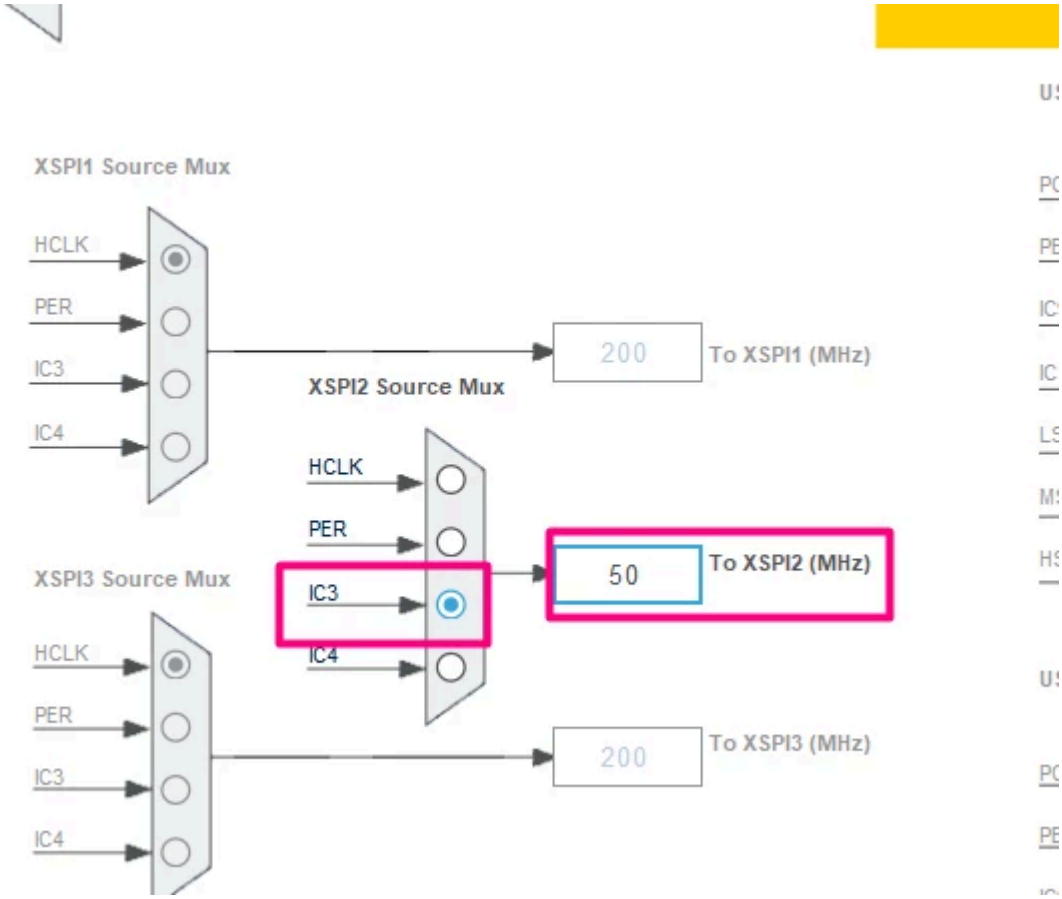
Ask AI



2.2.3 Configure the XSPI clock

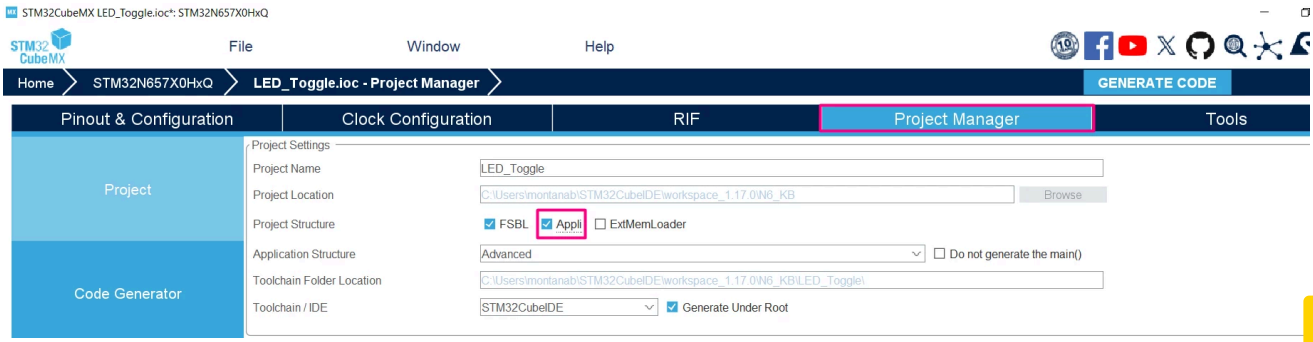
The last step for the XSPI is to configure its clock. For this example, we will set the XPI2 clock to 50 MHz. To achieve higher speeds up to the 200 MHz supported by this serial NOR FLASH, you need to make changes to OTP for I/O speed optimization, which is not covered in this article. Go to the **[Clock Configuration]** tab and have the IC3 as the source for the XSPI2 and type 50 and press enter for the clock to be automatically adjusted:

Ask AI



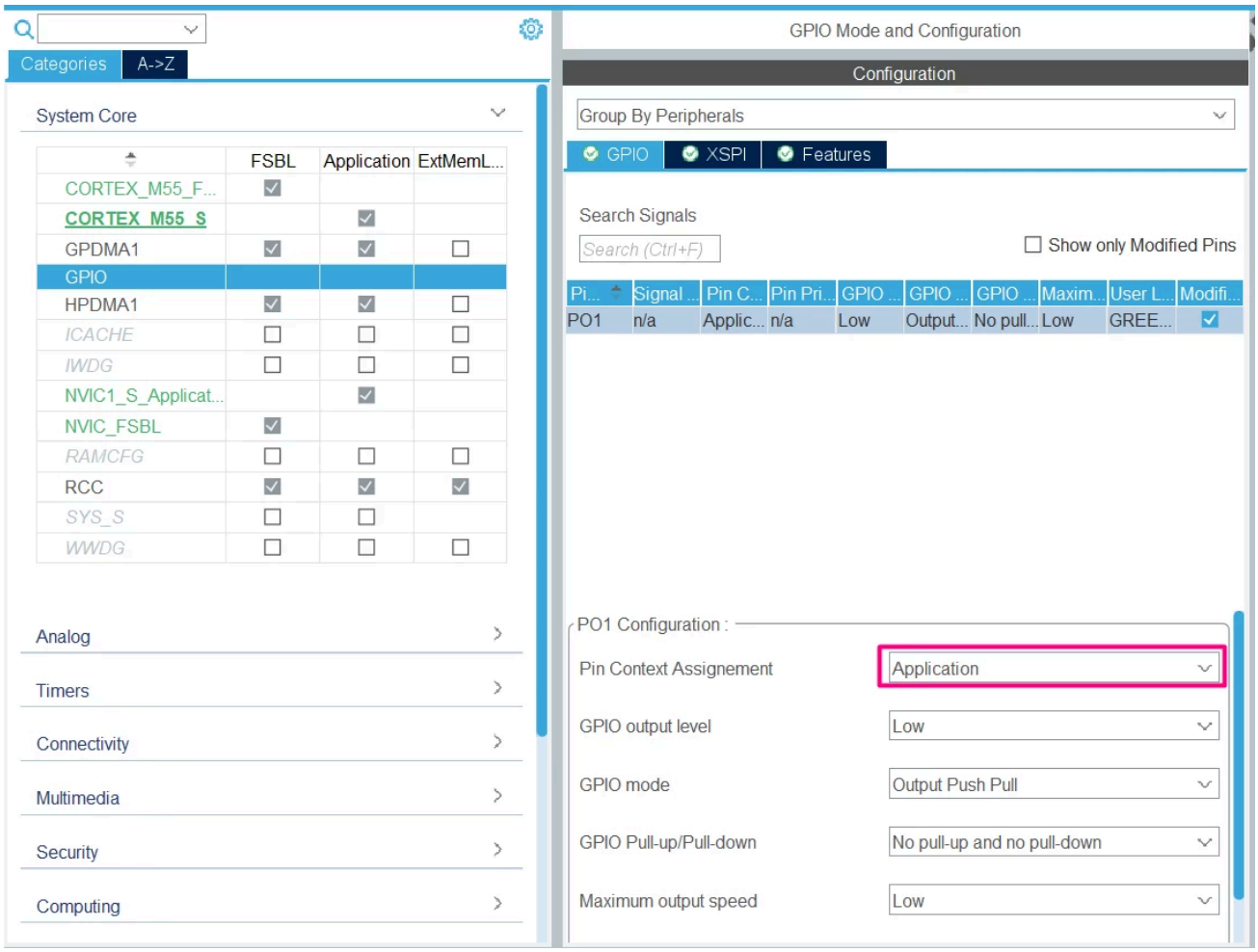
2.2.4 Code generation and build

Now that we are done with the XSPI, it is time to add the **[APPLI]** checkbox in the **[Project Manager]** tab, so the application project can be created:



And roll back to the **[Pinout & Configuration]** tab to change the **[GPIO]** LED pin to be assigned to the application instead of the FSBL:

Ask AI



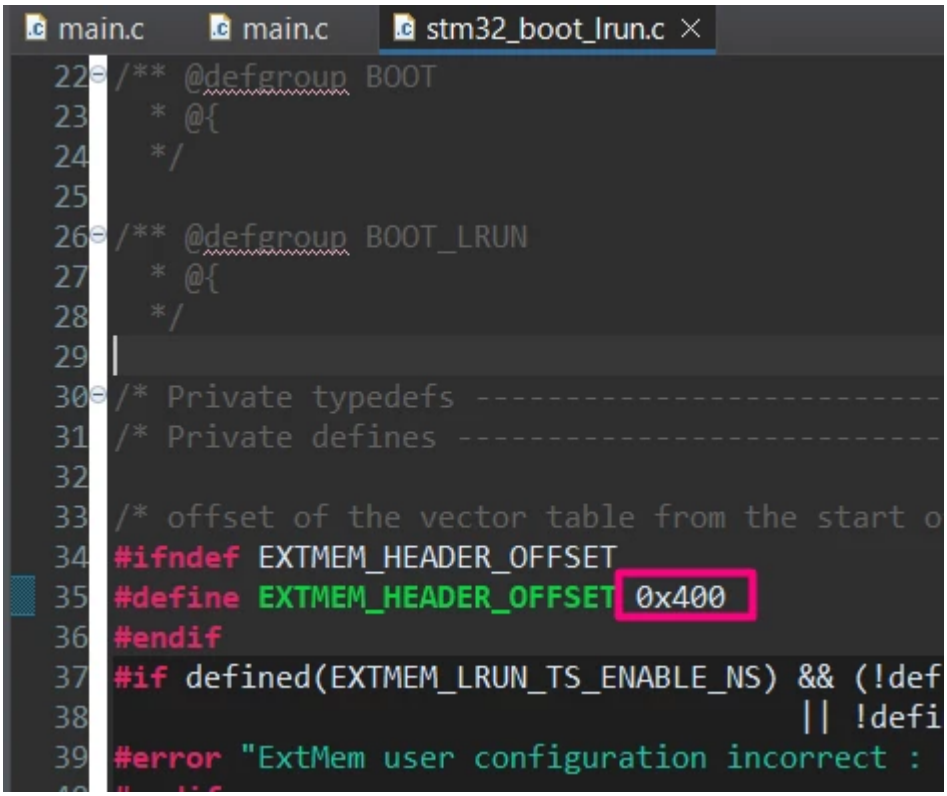
Proceed with the code generation for both projects.

In the _Appli project, locate the main.c file and add the toggle LED function call in its main loop:

```
 /* USER CODE BEGIN WHILE */
while (1)
{
/* USER CODE END WHILE */
/* USER CODE BEGIN 3 */
HAL_GPIO_TogglePin(GREEN_LED_GPIO_Port, GREEN_LED_Pin);
HAL_Delay(200);
}
/* USER CODE END 3 */
```

In the FSBL project, remove the previous user code portion in the main.c file, responsible for the LED toggling we had in the first hands-on. Make sure to either change the #define EXTMEM_HEADER_OFFSET from 0x0 to 0x400 in the stm32_boot_lrun.c or define it in the stm32_boot_lrun.h. The reason is that our application also has the 1 KB header, added by the script.

Ask AI



2.2.5 Adding the header and programming the binaries

Build both projects, making sure the *.bin is created for both the _Appli and the _FSBL projects. Once the build is done successfully, we perform the same step of calling the STM32CubeProgarmmer's CLI in each of the binaries respective folder to perform the signing. These are the commands:

FSBL:

```
"C:\Program Files\STMicroelectronics\STM32Cube\STM32CubeProgrammer\bin\STM32_SigningTool_CLI.exe" -bin LED_Toggle_FSBL.bin -nk -of 0x80000000 -t fsbl -o FSBL-trusted.bin -hv 2.3 -dump FSBL-trusted.bin -align
```

Application:

```
"C:\Program Files\STMicroelectronics\STM32Cube\STM32CubeProgrammer\bin\STM32_SigningTool_CLI.exe" -bin LED_Toggle_Appli.bin -nk -of 0x80000000 -t fsbl -o Appli-trusted.bin -hv 2.3 -dump Appli-trusted.bin -align
```

Use the same process using the STM32CubeProgrammer to program the FSBL at address 0x7000 0000 and the appli at address 0x7010 0000. Remember to have the STM32N6's BOOT1 set to HIGH and reset or power cycle the board before programming.

2.2.6 Validation

To see your application running, have the BOOT1 set to LOW, disconnect the progra cycle. Now, you have the green LED blinking with the code copied from the external fla ternal RAM by your custom FSBL!

Ask AI

Conclusion

By understanding the FSBL layout and its main features, developers can effectively utilize the FSBL to meet their specific needs. This article has provided a hands-on tutorial using the STM32N6570 discovery kit. demonstrating how to implement and run applications directly from the FSBL and how to configure the FSBL to load and run applications from external memory. By following these steps, developers can ensure a smooth and efficient boot process for their STM32N6-based projects.

Related links

- User manual 3234: How to proceed with boot ROM on STM32N6 MCUs
- User manual 3249: Getting started with STM32CubeN6 for STM32N6 series
- User manual 3300: Discovery kit with STM32N657X0 MCU

STM32CubeMX Bootloader STM32N6 series FSBL

6

Leave a reply...

23 replies

Sort by: **Oldest first**

asdfasdf
Associate III · 1 year ago

Hi @B.Montanari ! Thanks a lot for this informative tutorial! Would you mind attaching the resulting project to the article, and also the known-to-work binaries (FSBL and Appli, ideally both signed and without header)?

I followed your guide but the Appli binary seems to not boot, and having these files would help me (and probably others as well) debug the issue.

Thanks, Michael

1

B.Montanari AUTHOR
ST Technical Moderator · 1 year ago

Ask AI

Hi @asdfasdf ,

Thanks for the feedback!

I've uploaded the entire source code for this particular article in our STM32 HotSpot github page> stm32-hotspot/STM32N6_FSBL_Modes , hope it helps.

We also have the templates in the HAL driver repo, assuming the default installation folder, the template should be
here> C:\Users\%username%\STM32Cube\Repository\STM32Cube_FW_N6_V1.0.0\Projects\STM32N6570-DK\Templates\Template_FSBL_LRUN

Let me know if you have any follow up questions.

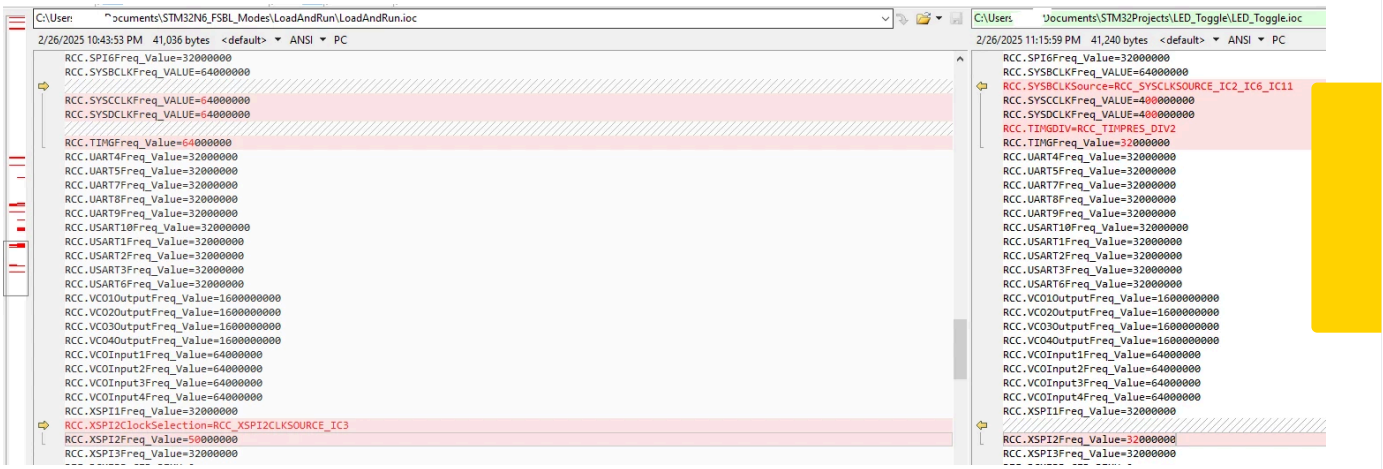
Best Regards
Bruno

B.Montanari

1

flexiglass
Associate II · 1 year ago

I had the same problem as the OP where the Appli binary wouldn't boot. Upon comparing the code shared by Bruno on Github I noticed the clock for IC3 in XSPI2 was set to 32MHz instead of 50MHz as per this guide. Once I set it to 32MHz and re-built, the Apply binary successfully booted. This change looks to have also made the clock solver to adjust other clocks as well, albeit to different values than what is in the Github repo.



Ask AI

asdfasdf

Associate III · 1 year ago

@flexiglass In my case the problem wasn't the XSPI2 clock (that works for me at 50 MHz as in the tutorial), but the SYSB clock (which feeds the AXI). I had this running at 400 MHz whereas @B.Montanari uses just 64 MHz. Lowering SYSB to 64 MHz made it work for me, and it also continues to work even if I increase the CPU clock (SYSA) to 600 MHz.

I have no idea why it is so sensitive to the SYSB clock, as that should not really have to do anything with the external flash I guess? I have the exact same problem when doing XIP instead of LRUN and asked about it here, sadly without an answer so far. This is pretty disappointing, as the MCU is dead-slow with RAM running at just 64 MHz...






B.Montanari AUTHOR

ST Technical Moderator · 1 year ago

Hi @flexiglass and @asdfasdf ,

I've just made some changes in the *.ioc file to run the CPU at 600MHz, AXISRAM at 400MHz and kept the IC3 (XSPI2) at 50MHz. Please see the attached *.ioc. In this version I've also added the ICACHE and DCACHE plus the MPU settings, so we should have the fastest performance results without going into overdrive mode, which allows the core to go up to 800MHz. The zip for the changed project is here> STM32N6_FSBL_Modes/LoadAndRun/LoadAndRun_600MHz.7z at main · stm32-hotspot/STM32N6_FSBL_Modes

I'm planning on writing a quick article on the overdrive and another one on the OTP programming, so I could detail more on how to change the settings to get a higher clock on the XSPI2 in a reliable manner as well. I'll have a code example for that as well once I'm done.

Hope this helps and don't forget to change the #define EXTMEM_HEADER_OFFSET to 0x400 after code generation.

The final comment, if you want to avoid always using the CLI, you can have it as a post build command:. Go to the project properties, C/C++ Build, Settings then type this command in the Build Steps tab under the Post-build steps:

cd "${ProjDirPath}/Debug" && echo y | "C:\Program Files\STMicroelectronics\STM32Cube\STM32CubeProgrammer\bin\STM32_SigningTool_CLI.exe" -bin "${ProjName}.bin" -nk -of 0x80000000 -t fsbl -o "${ProjName}-Trusted.bin" -hv 2.3 -du ...e}-Trusted.bin"

Ask AI

You can use the same command for the Appli and FSBL projects.

Best Regards

B.Montanari

asdfasdf
Associate III · 1 year ago

Hi @B.Montanari , thank you very much for your help in resolving this issue! Sadly, even flashing the compiled binaries that you uploaded the board does not start (that is, no blinking LED). I'm starting to suspect something might be wrong with my Discovery board, or that this is due to differing OTP fuse configurations. I have not changed anything in this regard but am using the Discovery board exactly in the state I bought it.

I have attached an Export of the OTP settings generated by CubeProg - would you mind doing the same for your board (where this is apparently working) and diffing with mine to see whether there are any differences?

Also, I attached three photos showing the exact hardware revision of the Discovery board and MCU - it would be great if you could check this against your version as well, just to make sure I'm not having a broken hardware revision or something like that.

Thank you very much!

Best regards, Michael

```
Name | Word | Value | Status | Lock
-------------------------------------------------+-----------------+---------------
--+-----------------+----------------
OTP_HW_WORD0 | OTP0 | 0xAAAA5555 | 0x40000000 | Yes
OTP_HW_WORD1 | OTP1 | 0x0000000F | 0x00000000 | No
OTP_HW_WORD2 | OTP2 | 0x00000000 | 0x00000000 | No
OTP_HW_WORD3 | OTP3 | 0x00000000 | 0x00000000 | No
OTP_HW_WORD4 | OTP4 | 0x0000F0FF | 0x00000000 | No
ID0 | OTP5 | 0x0044002E | 0x40000000 | Yes
ID1 | OTP6 | 0x4236500E | 0x40000000 | Yes
ID2 | OTP7 | 0x0036324E | 0x40000000 | Yes
OTP_RPN_OPTION | OTP8 | 0xFFFFFFFF | 0x40000000 | Yes
OTP_RPN_CODING | OTP9 | 0x00002000 | 0x40000000 | Yes
BOOTROM_CONFIG_1 | OTP10 | 0x00030103 | 0x00000000 | No
BOOTROM_CONFIG_2 | OTP11 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_3 | OTP12 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_4 | OTP13 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_5 | OTP14 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_6 | OTP15 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_7 | OTP16 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_8 | OTP17 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_9 | OTP18 | 0x00000000 | 0x00000000 | No
```

Ask AI

```
BOOTROM_CONFIG_10 | OTP19 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_11 | OTP20 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_12 | OTP21 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_13 | OTP22 | 0x00000000 | 0x00000000 | No
BOOTROM_CONFIG_14 | OTP23 | 0x36FC814F | 0x40000000 | Yes
BOOT_TZ_EPOCH0 | OTP24 | 0x00000000 | 0x00000000 | No
BOOT_TZ_EPOCH1 | OTP25 | 0x00000000 | 0x00000000 | No
BOOT_TZ_EPOCH2 | OTP26 | 0x00000000 | 0x00000000 | No
BOOT_TZ_EPOCH3 | OTP27 | 0x00000000 | 0x00000000 | No
BOOT_TZ_EPOCH4 | OTP28 | 0x00000000 | 0x00000000 | No
BOOT_TZ_EPOCH5 | OTP29 | 0x00000000 | 0x00000000 | No
BOOT_TZ_EPOCH6 | OTP30 | 0x00000000 | 0x00000000 | No
BOOT_TZ_EPOCH7 | OTP31 | 0x00000000 | 0x00000000 | No
BOOT_NS_EPOCH0 | OTP32 | 0x00000000 | 0x00000000 | No
BOOT_NS_EPOCH1 | OTP33 | 0x00000000 | 0x00000000 | No
BOOT_NS_EPOCH2 | OTP34 | 0x00000000 | 0x00000000 | No
BOOT_NS_EPOCH3 | OTP35 | 0x00000000 | 0x00000000 | No
BOOT_NS_EPOCH4 | OTP36 | 0x00000000 | 0x00000000 | No
BOOT_NS_EPOCH5 | OTP37 | 0x00000000 | 0x00000000 | No
BOOT_NS_EPOCH6 | OTP38 | 0x00000000 | 0x00000000 | No
BOOT_NS_EPOCH7 | OTP39 | 0x00000000 | 0x00000000 | No
 available to customer | OTP40 | 0x00000000 | 0x00000000 | No
available to customer | OTP41 | 0x00000000 | 0x00000000 | No
available to customer | OTP42 | 0x00000000 | 0x00000000 | No
available to customer | OTP43 | 0x00000000 | 0x00000000 | No
available to customer | OTP44 | 0x00000000 | 0x00000000 | No
available to customer | OTP45 | 0x00000000 | 0x00000000 | No
available to customer | OTP46 | 0x00000000 | 0x00000000 | No
available to customer | OTP47 | 0x00000000 | 0x00000000 | No
available to customer | OTP48 | 0x00000000 | 0x00000000 | No
available to customer | OTP49 | 0x00000000 | 0x00000000 | No
available to customer | OTP50 | 0x00000000 | 0x00000000 | No
available to customer | OTP51 | 0x00000000 | 0x00000000 | No
available to customer | OTP52 | 0x00000000 | 0x00000000 | No
available to customer | OTP53 | 0x00000000 | 0x00000000 | No
available to customer | OTP54 | 0x00000000 | 0x00000000 | No
available to customer | OTP55 | 0x00000000 | 0x00000000 | No
TAMP_EN | OTP56 | 0x00000000 | 0x00000000 | No
TAMP_CFM | OTP57 | 0x00000000 | 0x00000000 | No
TAMP_CFG | OTP58 | 0x00000000 | 0x00000000 | No
available to customer | OTP59 | 0x00000000 | 0x00000000 | No
available to customer | OTP60 | 0x00000000 | 0x00000000 | No
available to customer | OTP61 | 0x00000000 | 0x00000000 | No
available to customer | OTP62 | 0x00000000 | 0x00000000 | No
available to customer | OTP63 | 0x00000000 | 0x00000000 | No
available to customer | OTP64 | 0x00000000 | 0x00000000 | No
available to customer | OTP65 | 0x00000000 | 0x00000000 | No
available to customer | OTP66 | 0x00000000 | 0x00000000 | No
available to customer | OTP67 | 0x00000000 | 0x00000000 | No
```

Ask AI

available to customer	OTP68	0x00000000	0x00000000	No
available to customer	OTP69	0x00000000	0x00000000	No
available to customer	OTP70	0x00000000	0x00000000	No
available to customer	OTP71	0x00000000	0x00000000	No
available to customer	OTP72	0x00000000	0x00000000	No
available to customer	OTP73	0x00000000	0x00000000	No
available to customer	OTP74	0x00000000	0x00000000	No
available to customer	OTP75	0x00000000	0x00000000	No
available to customer	OTP76	0x00000000	0x00000000	No
available to customer	OTP77	0x00000000	0x00000000	No
available to customer	OTP78	0x00000000	0x00000000	No
available to customer	OTP79	0x00000000	0x00000000	No
available to customer	OTP80	0x00000000	0x00000000	No
available to customer	OTP81	0x00000000	0x00000000	No
available to customer	OTP82	0x00000000	0x00000000	No
available to customer	OTP83	0x00000000	0x00000000	No
available to customer	OTP84	0x00000000	0x00000000	No
available to customer	OTP85	0x00000000	0x00000000	No
available to customer	OTP86	0x00000000	0x00000000	No
available to customer	OTP87	0x00000000	0x00000000	No
available to customer	OTP88	0x00000000	0x00000000	No
available to customer	OTP89	0x00000000	0x00000000	No
available to customer	OTP90	0x00000000	0x00000000	No
available to customer	OTP91	0x00000000	0x00000000	No
available to customer	OTP92	0x00000000	0x00000000	No
available to customer	OTP93	0x00000000	0x00000000	No
available to customer	OTP94	0x00000000	0x00000000	No
available to customer	OTP95	0x00000000	0x00000000	No
Reserved	OTP96	0x1F38978A	0x40000000	Yes
Reserved	OTP97	0x00251132	0x40000000	Yes
Reserved	OTP98	0x00000130	0x40000000	Yes
Reserved	OTP99	0x00007889	0x40000000	Yes
Reserved	OTP100	0x00000000	0x00000000	No
Reserved	OTP101	0x00000000	0x00000000	No
Reserved	OTP102	0x00000000	0x00000000	No
Reserved	OTP103	0xC57E7E55	0x40000000	Yes
Reserved	OTP104	0x00000000	0x40000000	Yes
Reserved	OTP105	0x00000000	0x40000000	Yes
Reserved	OTP106	0x46523E53	0x40000000	Yes
Reserved	OTP107	0x00000000	0x00000000	No
Reserved	OTP108	0x00000000	0x00000000	No
Reserved	OTP109	0x00000000	0x00000000	No
Reserved	OTP110	0x00000718	0x40000000	Yes
Reserved	OTP111	0x00000000	0x40000000	Yes
Reserved	OTP112	0x00000000	0x00000000	No
Reserved	OTP113	0x00763331	0x40000000	Yes
Reserved	OTP114	0x80000249	0x00000000	No
Reserved	OTP115	0x00000000	0x40000000	Yes
Reserved	OTP116	0x00000000	0x00000000	No

Ask AI

```
Reserved | OTP117 | 0x00000000 | 0x00000000 | No
Reserved | OTP118 | 0x00000000 | 0x00000000 | No
Reserved | OTP119 | 0x00000000 | 0x00000000 | No
Reserved | OTP120 | 0x3047C60E | 0x40000000 | Yes
Reserved | OTP121 | 0x013B8079 | 0x40000000 | Yes
Reserved | OTP122 | 0x93C0078C | 0x40000000 | Yes
Reserved | OTP123 | 0x9E087777 | 0x40000000 | Yes
HCONF1 | OTP124 | 0x00018000 | 0x00000000 | No
Reserved | OTP125 | 0x00000000 | 0x40000000 | Yes
Reserved | OTP126 | 0x00000000 | 0x40000000 | Yes
Reserved | OTP127 | 0x00000000 | 0x40000000 | Yes
STM32_CERTIF0 | OTP128 | 0xC0EF8743 | 0x40000000 | Yes
STM32_CERTIF1 | OTP129 | 0x56A778B1 | 0x40000000 | Yes
STM32_CERTIF2 | OTP130 | 0xF080A19E | 0x40000000 | Yes
STM32_CERTIF3 | OTP131 | 0xF8FA0E42 | 0x40000000 | Yes
STM32_CERTIF4 | OTP132 | 0xC7DD2450 | 0x40000000 | Yes
STM32_CERTIF5 | OTP133 | 0x6E0B01B1 | 0x40000000 | Yes
STM32_CERTIF6 | OTP134 | 0x0B1FC481 | 0x40000000 | Yes
STM32_CERTIF7 | OTP135 | 0x3D40305F | 0x40000000 | Yes
STM32_CERTIF8 | OTP136 | 0x9A7662EE | 0x40000000 | Yes
STM32_CERTIF9 | OTP137 | 0x7EA6323A | 0x40000000 | Yes
STM32_CERTIF10 | OTP138 | 0x05BF7294 | 0x40000000 | Yes
STM32_CERTIF11 | OTP139 | 0xF85E01B4 | 0x40000000 | Yes
STM32_CERTIF12 | OTP140 | 0x105B9B8A | 0x40000000 | Yes
STM32_CERTIF13 | OTP141 | 0x30769158 | 0x40000000 | Yes
STM32_CERTIF14 | OTP142 | 0xF4EEBD72 | 0x40000000 | Yes
STM32_CERTIF15 | OTP143 | 0x71DE1D05 | 0x40000000 | Yes
STM32PUBKEY0 | OTP144 | 0x151AA162 | 0x40000000 | Yes
STM32PUBKEY1 | OTP145 | 0x1FC421F5 | 0x40000000 | Yes
STM32PUBKEY2 | OTP146 | 0xC099CF18 | 0x40000000 | Yes
STM32PUBKEY3 | OTP147 | 0x5017C2A9 | 0x40000000 | Yes
STM32PUBKEY4 | OTP148 | 0xEE5462AE | 0x40000000 | Yes
STM32PUBKEY5 | OTP149 | 0x7E691775 | 0x40000000 | Yes
STM32PUBKEY6 | OTP150 | 0xCC771020 | 0x40000000 | Yes
STM32PUBKEY7 | OTP151 | 0x275B9E70 | 0x40000000 | Yes
STM32PUBKEY8 | OTP152 | 0x4CE648A4 | 0x40000000 | Yes
STM32PUBKEY9 | OTP153 | 0x8BF2D84C | 0x40000000 | Yes
STM32PUBKEY10 | OTP154 | 0xCEF0E429 | 0x40000000 | Yes
STM32PUBKEY11 | OTP155 | 0x855984EB | 0x40000000 | Yes
STM32PUBKEY12 | OTP156 | 0x1A63E8DC | 0x40000000 | Yes
STM32PUBKEY13 | OTP157 | 0x47B71FAE | 0x40000000 | Yes
STM32PUBKEY14 | OTP158 | 0x4644432D | 0x40000000 | Yes
STM32PUBKEY15 | OTP159 | 0x5B0B721A | 0x40000000 | Yes
 OTP_ROT_HASH0 | OTP160 | 0x00000000 | 0x00000000 | No
OTP_ROT_HASH1 | OTP161 | 0x00000000 | 0x00000000 | No
OTP_ROT_HASH2 | OTP162 | 0x00000000 | 0x00000000 | No
OTP_ROT_HASH3 | OTP163 | 0x00000000 | 0x00000000 | No
OTP_ROT_HASH4 | OTP164 | 0x00000000 | 0x00000000 | No
OTP_ROT_HASH5 | OTP165 | 0x00000000 | 0x00000000 | No
```

Ask AI

OTP_ROT_HASH6	OTP166	0x00000000	0x00000000	No
OTP_ROT_HASH7	OTP167	0x00000000	0x00000000	No
ST_RSSE_EDMK_DERIV_CSTE_FUSE	OTP168	0x4FB6E1A0	0x40000000	Yes
OTP_MAC1_ADDR_LOW	OTP169	0x00000000	0x00000000	No
OTP_MAC1_ADDR_HIGH	OTP170	0x00000000	0x00000000	No
OTP_MAC2_ADDR_LOW	OTP171	0x00000000	0x00000000	No
OTP_MAC2_ADDR_HIGH	OTP172	0x00000000	0x00000000	No
available to customer	OTP173	0x00000000	0x00000000	No
available to customer	OTP174	0x00000000	0x00000000	No
available to customer	OTP175	0x00000000	0x00000000	No
available to customer	OTP176	0x00000000	0x00000000	No
available to customer	OTP177	0x00000000	0x00000000	No
available to customer	OTP178	0x00000000	0x00000000	No
available to customer	OTP179	0x00000000	0x00000000	No
available to customer	OTP180	0x00000000	0x00000000	No
available to customer	OTP181	0x00000000	0x00000000	No
available to customer	OTP182	0x00000000	0x00000000	No
available to customer	OTP183	0x00000000	0x00000000	No
available to customer	OTP184	0x00000000	0x00000000	No
available to customer	OTP185	0x00000000	0x00000000	No
available to customer	OTP186	0x00000000	0x00000000	No
available to customer	OTP187	0x00000000	0x00000000	No
available to customer	OTP188	0x00000000	0x00000000	No
available to customer	OTP189	0x00000000	0x00000000	No
available to customer	OTP190	0x00000000	0x00000000	No
available to customer	OTP191	0x00000000	0x00000000	No
available to customer	OTP192	0x00000000	0x00000000	No
available to customer	OTP193	0x00000000	0x00000000	No
available to customer	OTP194	0x00000000	0x00000000	No
available to customer	OTP195	0x00000000	0x00000000	No
available to customer	OTP196	0x00000000	0x00000000	No
available to customer	OTP197	0x00000000	0x00000000	No
available to customer	OTP198	0x00000000	0x00000000	No
available to customer	OTP199	0x00000000	0x00000000	No
available to customer	OTP200	0x00000000	0x00000000	No
available to customer	OTP201	0x00000000	0x00000000	No
available to customer	OTP202	0x00000000	0x00000000	No
available to customer	OTP203	0x00000000	0x00000000	No
available to customer	OTP204	0x00000000	0x00000000	No
available to customer	OTP205	0x00000000	0x00000000	No
available to customer	OTP206	0x00000000	0x00000000	No
available to customer	OTP207	0x00000000	0x00000000	No
available to customer	OTP208	0x00000000	0x00000000	No
available to customer	OTP209	0x00000000	0x00000000	No
available to customer	OTP210	0x00000000	0x00000000	No
available to customer	OTP211	0x00000000	0x00000000	No
available to customer	OTP212	0x00000000	0x00000000	No
available to customer	OTP213	0x00000000	0x00000000	No
available to customer	OTP214	0x00000000	0x00000000	No

Ask AI

```
available to customer | OTP215 | 0x00000000 | 0x00000000 | No
available to customer | OTP216 | 0x00000000 | 0x00000000 | No
available to customer | OTP217 | 0x00000000 | 0x00000000 | No
available to customer | OTP218 | 0x00000000 | 0x00000000 | No
available to customer | OTP219 | 0x00000000 | 0x00000000 | No
available to customer | OTP220 | 0x00000000 | 0x00000000 | No
available to customer | OTP221 | 0x00000000 | 0x00000000 | No
available to customer | OTP222 | 0x00000000 | 0x00000000 | No
available to customer | OTP223 | 0x00000000 | 0x00000000 | No
available to customer | OTP224 | 0x00000000 | 0x00000000 | No
available to customer | OTP225 | 0x00000000 | 0x00000000 | No
available to customer | OTP226 | 0x00000000 | 0x00000000 | No
available to customer | OTP227 | 0x00000000 | 0x00000000 | No
available to customer | OTP228 | 0x00000000 | 0x00000000 | No
available to customer | OTP229 | 0x00000000 | 0x00000000 | No
available to customer | OTP230 | 0x00000000 | 0x00000000 | No
available to customer | OTP231 | 0x00000000 | 0x00000000 | No
available to customer | OTP232 | 0x00000000 | 0x00000000 | No
available to customer | OTP233 | 0x00000000 | 0x00000000 | No
available to customer | OTP234 | 0x00000000 | 0x00000000 | No
available to customer | OTP235 | 0x00000000 | 0x00000000 | No
available to customer | OTP236 | 0x00000000 | 0x00000000 | No
available to customer | OTP237 | 0x00000000 | 0x00000000 | No
available to customer | OTP238 | 0x00000000 | 0x00000000 | No
available to customer | OTP239 | 0x00000000 | 0x00000000 | No
available to customer | OTP240 | 0x00000000 | 0x00000000 | No
available to customer | OTP241 | 0x00000000 | 0x00000000 | No
available to customer | OTP242 | 0x00000000 | 0x00000000 | No
available to customer | OTP243 | 0x00000000 | 0x00000000 | No
available to customer | OTP244 | 0x00000000 | 0x00000000 | No
available to customer | OTP245 | 0x00000000 | 0x00000000 | No
available to customer | OTP246 | 0x00000000 | 0x00000000 | No
available to customer | OTP247 | 0x00000000 | 0x00000000 | No
available to customer | OTP248 | 0x00000000 | 0x00000000 | No
available to customer | OTP249 | 0x00000000 | 0x00000000 | No
available to customer | OTP250 | 0x00000000 | 0x00000000 | No
available to customer | OTP251 | 0x00000000 | 0x00000000 | No
available to customer | OTP252 | 0x00000000 | 0x00000000 | No
available to customer | OTP253 | 0x00000000 | 0x00000000 | No
available to customer | OTP254 | 0x00000000 | 0x00000000 | No
available to customer | OTP255 | 0x00000000 | 0x00000000 | No
OTP_RMA_LOCK_PSWD0 | OTP256 | 0x00000000 | 0x00000000 | No
OTP_RMA_LOCK_PSWD1 | OTP257 | 0x00000000 | 0x00000000 | No
OTP_RMA_LOCK_PSWD2 | OTP258 | 0x00000000 | 0x00000000 | No
OTP_RMA_LOCK_PSWD3 | OTP259 | 0x00000000 | 0x00000000 | No
OEM Secrets available to customer | OTP260 | 0xC48FFC64 | 0x40000000 |
OEM Secrets available to customer | OTP261 | 0x8B94A062 | 0x40000000 | Yes
OEM Secrets available to customer | OTP262 | 0x0B9085FE | 0x40000000 | Yes
OEM Secrets available to customer | OTP263 | 0x68B30CC5 | 0x40000000 | Yes
```

Ask AI

OEM Secrets available to customer	OTP264	0x33E0B6E5	0x40000000	Yes
OEM Secrets available to customer	OTP265	0xE30D255E	0x40000000	Yes
OEM Secrets available to customer	OTP266	0xD49FDEC8	0x40000000	Yes
OEM Secrets available to customer	OTP267	0xBEE6EC4C	0x40000000	Yes
OEM Secrets available to customer	OTP268	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP269	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP270	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP271	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP272	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP273	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP274	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP275	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP276	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP277	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP278	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP279	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP280	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP281	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP282	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP283	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP284	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP285	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP286	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP287	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP288	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP289	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP290	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP291	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP292	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP293	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP294	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP295	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP296	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP297	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP298	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP299	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP300	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP301	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP302	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP303	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP304	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP305	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP306	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP307	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP308	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP309	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP310	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP311	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP312	0x00000000	0x00000000	No

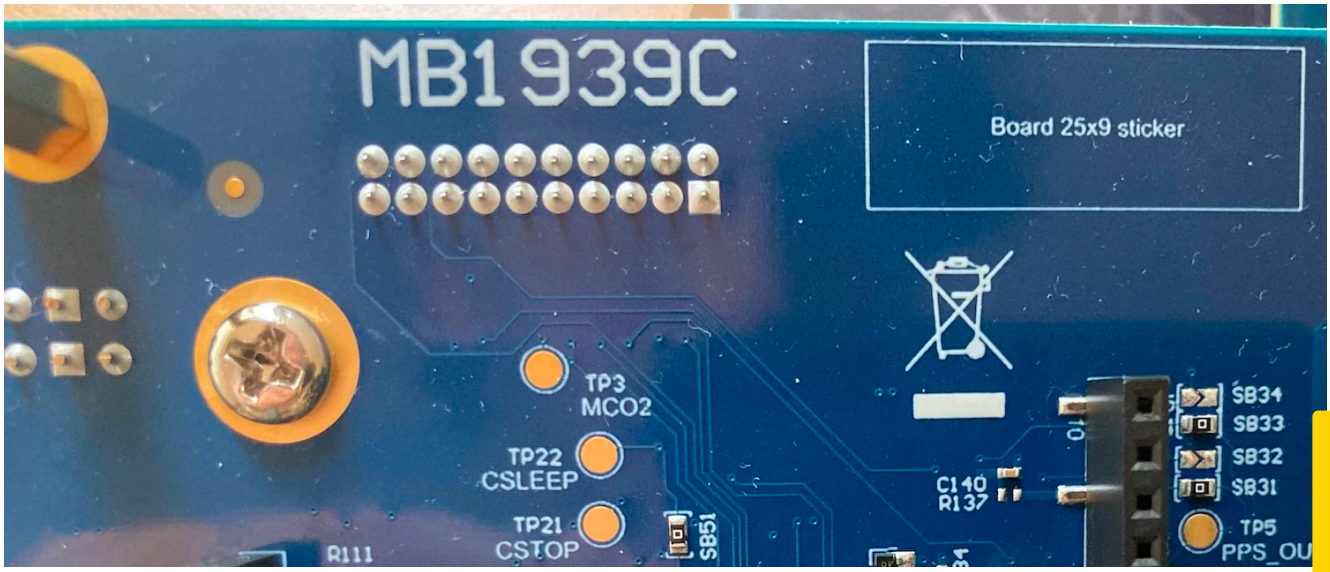
Ask AI

OEM Secrets available to customer	OTP313	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP314	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP315	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP316	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP317	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP318	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP319	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP320	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP321	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP322	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP323	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP324	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP325	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP326	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP327	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP328	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP329	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP330	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP331	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP332	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP333	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP334	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP335	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP336	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP337	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP338	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP339	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP340	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP341	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP342	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP343	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP344	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP345	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP346	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP347	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP348	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP349	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP350	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP351	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP352	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP353	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP354	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP355	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP356	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP357	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP358	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP359	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP360	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP361	0x00000000	0x00000000	No

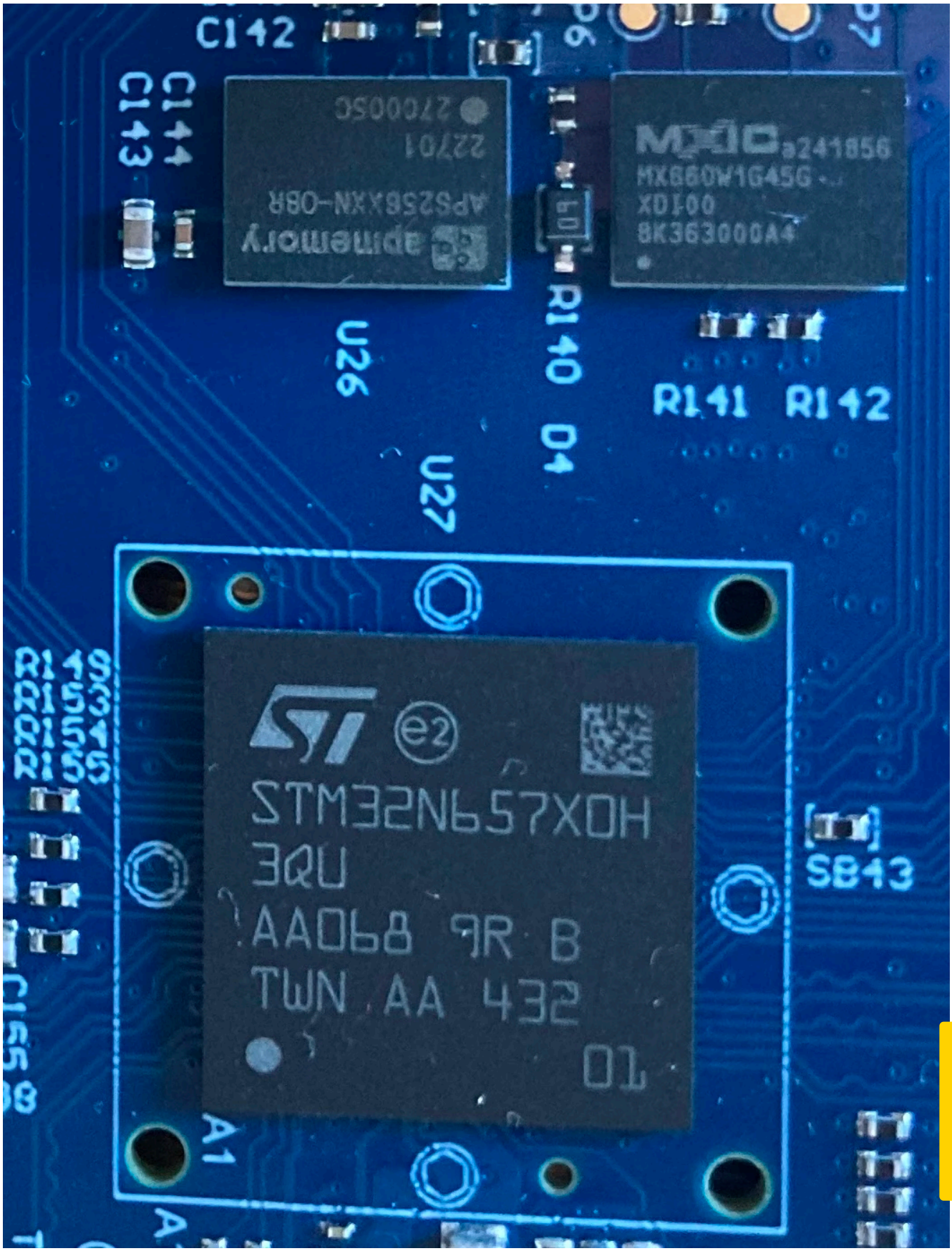
Ask AI

OEM Secrets available to customer	OTP362	0x00000000	0x00000000	No
OEM Secrets available to customer	OTP363	0x00000000	0x00000000	No
OEM_SECRET_FOR_CRYPTED_BOOT0	OTP364	0x00000000	0x00000000	No
OEM_SECRET_FOR_CRYPTED_BOOT1	OTP365	0x00000000	0x00000000	No
OEM_SECRET_FOR_CRYPTED_BOOT2	OTP366	0x00000000	0x00000000	No
OEM_SECRET_FOR_CRYPTED_BOOT3	OTP367	0x00000000	0x00000000	No





Ask AI



1

Ask AI

asdfasdf
Associate III · 1 year ago

Ok, so I got it working, the problem was that even when running the XSPI flash at 50 MHz the "high speed IO optimizations" should be enabled - see also this thread.

Thanks again @B.Montanari !

1

exarian
Associate III · 1 year ago

Thank you @B.Montanari and ST team for the great article!

Do you have an example that checks the integrity of the Appli signature on the FSBL?

For example what happens if an error occurred during the copy?

The FSBL should be able to recover this and try again. As a first step it could check that the hash of the file in RAM matches that which was expected in the Header.

Looking through the LoadAndRun example it appears the Appli signature is ignored?

It appears the Application is copied from external flash to RAM, and then immediately executed, without checking its integrity?

Thank you again for the write up!

Kind regards,

B.Montanari AUTHOR
ST Technical Moderator · 1 year ago

Hi @exarian , thanks for the feedback!

Please see my comments below:

Q1: Do you have an example that checks the integrity of the Appli signature on the FSBL?

Ask AI

C1: Currently we don't have any examples to cover that portion without using the Secure framework. Given the higher complexity of the topic, it will be addressed in our wiki page instead. The content is under creation, but will be here once available> Category:STM32N6 - stm32mcu

Q2: For example what happens if an error occurred during the copy?

C2: If an error happens during the copy, since there is no control or verification, it will execute the jump to the copied region and eventually get a hardfault.

Q3: Looking through the LoadAndRun example it appears the Appli signature is ignored?

C3: Yes, the Appli signature is ignored in this example, the firmware is bypassing/ignoring the header.

Q4: It appears the Application is copied from external flash to RAM, and then immediately executed, without checking its integrity?

C4: Correct, we can work on a code example that implements a small verification to ensure the copy is properly made.

Thanks!

Best Regards

B.Montanari

1

exarian
Associate III · 1 year ago

Thank you @B.Montanari ,

I really appreciate that feedback and your answers.

I will keep an eye on the Wiki's. I am excited to see what comes next, looking forward to future updates!

Kind Regards,

Show more replies

ST Community highlights – April to June 2026

Ask AI

Related topics

How to connect the STM32N6 using STLINK and serial interface

STM32 MCUs

STM32N657L0 NOR flash XIP/LRUN problem

STM32 MCUs Products

STM32N657I0HxQ - Custom Board - FSBL Load & Run issue

STM32CubeProgrammer (MCUs)

Linker garbage problem when deploying AI models on the STM32N6 development board

Edge AI

Problem with STM32N6 Nucle USB Device CDC ACM

STM32 MCUs Embedded software

Popular tags

STM32H7 series | STM32F4 series | STM32CubeMX | STM32CubeIDE | TouchGFX | STM32L4 series | Legacy products | STM32F7 series | USB | UART-USART

About STMicroelectronics

Who we are

Investor relations

Sustainability

Innovation & technology

Connect with us

Contact ST offices

Find sales offices & distributors

Community

Newsroom

Ask AI

Careers

Events & trainings

Suppliers to ST

AI at ST

Blog

General terms and conditions

Browse

Shortcuts

Sitemap

Compliance, Ethics, Security & Privacy

Ethics and compliance

ST Ethics Hotline

ST's Code of Conduct

Security & privacy portal

Follow us

All rights reserved © 2026 STMicroelectronics | Terms of use | Sales terms & conditions | Trademarks | Privacy portal | Manage cookies

Ask AI