



Getting started with STM32H7Rx/7Sx MCUs in STM32CubeIDE

Introduction

This application note describes how to get started with projects based on STM32H7Rx/7Sx microcontrollers in the STMicroelectronics [STM32CubeIDE](#) integrated development environment.

Table 1. Applicable products

Type	Products
Microcontrollers	STM32H7R3/7S3 line, STM32H7R7/7S7 line
Software development tool	STM32CubeIDE



1 General information

STM32CubeIDE supports STM32 32-bit products based on the Arm® Cortex® processor.

Note: Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.



1.1 Prerequisites

The following tools are prerequisites for understanding the tutorial in this document and developing an application based on the STM32H7Rx/7Sx microcontrollers:

- [STM32CubeIDE 1.15.0](#) or newer
- [STM32Cube_FW_H7RS_V1.0.0](#) or newer
- [STM32CubeMX 6.12.0](#) or newer

Users are advised to keep updated with the documentation evolution of the STM32H7Rx/7Sx microcontrollers at www.st.com/en/microcontrollers-microprocessors/stm32h7-series.html.

1.2 The use cases in this document

In the [STM32CubeIDE](#) context, users have different ways to explore and get started with the development of projects based on the STM32H7Rx/7Sx microcontrollers. From the list below, select the description that best fits the use case considered and refer to the corresponding section in this application note:

- I want to learn with and explore example projects:
Refer to [Section 2.3: Import a project from the STM32CubeH7RS MCU Package](#)
- I want to start a first STM32H7Rx/7Sx project:
 - Empty project – No STM32CubeMX support for maximum flexibility.
Refer to [Section 2.2: Create an empty project based on the template in the STM32CubeH7RS MCU Package](#)
 - STM32CubeH7RS project – STM32CubeMX-managed project.
Refer to [Section 2.1: Create a new STM32 project](#)

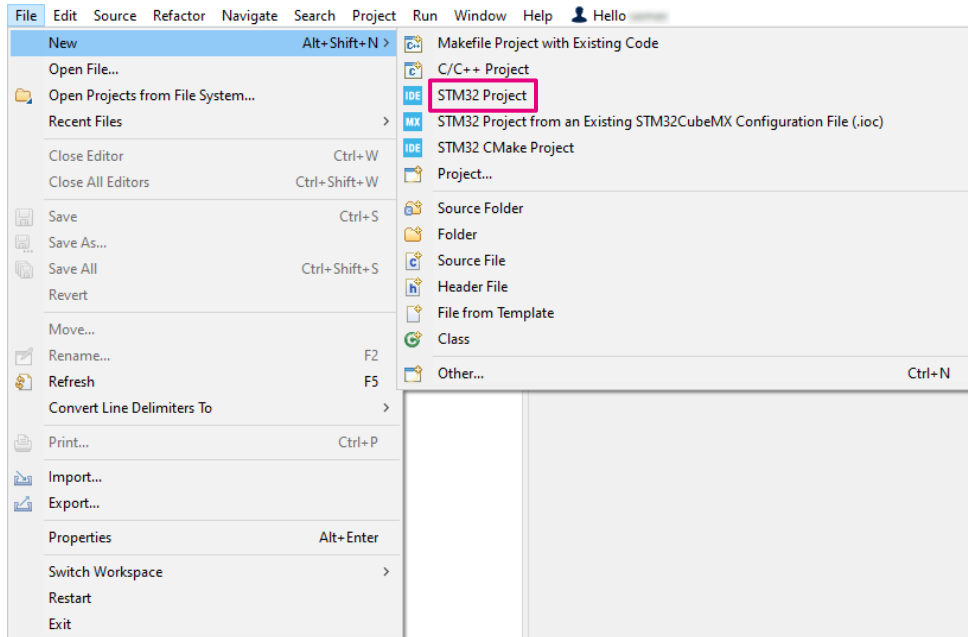
2 Create and import projects

This chapter describes how to create or import projects for STM32H7Rx/7Sx microcontrollers.

2.1 Create a new STM32 project

To start a new project, go to [File]>[New]>[STM32 Project] as shown in Figure 1.

Figure 1. New STM32 project



DT73572V1

Select the desired MCU or board. In the example illustrated in Figure 2, the selected board is the STM32H7S78-DK. Click on [Next >].

Figure 2. Target selection

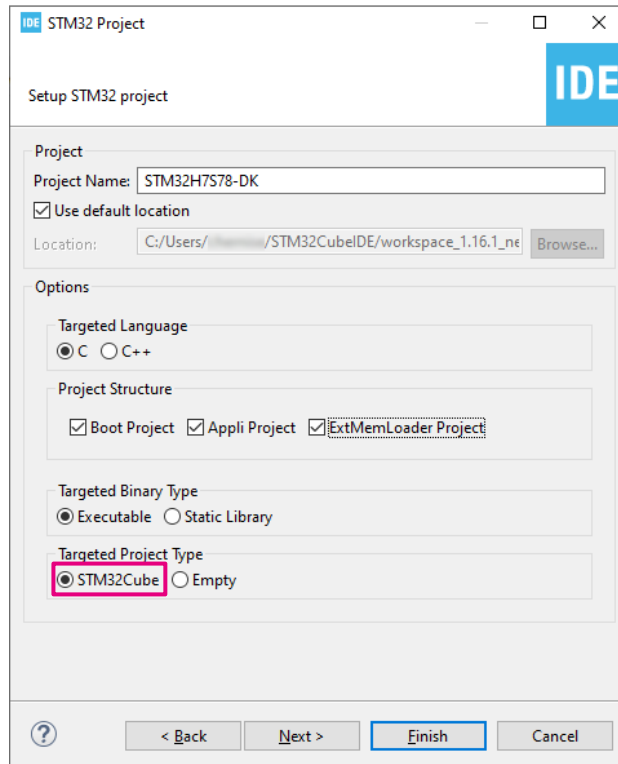
The screenshot shows the 'Target Selection' window in the IDE. The 'Board Selector' tab is active, displaying a 'Board Filters' panel on the left and a product overview on the right. The 'Boards List' table at the bottom shows one item: STM32H7S78-DK.

	Overview	Commercial Part No	Type	Marketing Status	Unit Price (US\$)	Mounted Device
☆		STM32H7S78-DK	Discovery Kit	Active	99.0	STM32H7S7L8H6H

DT73573V1

After the target selection comes the project setup step shown in Figure 3. The *Targeted Project Type* setting determines whether the project gets generated by STM32CubeMX or not. An *Empty* project is a skeleton of a project that needs building upon while *STM32Cube* indicates an STM32CubeMX-managed project.

Figure 3. Project setup (STM32CubeMX-managed project creation)



DT73574V1

The created project contains up to three subprojects:

- STM32H7S78-DK_Appli: dedicated to the main application development.
- STM32H7S78-DK_Boot: dedicated to the bootloader development.
- STM32H7S78-DK_ExtMemLoader (optional and not generated by default): to be used when the user needs an own external flash loader.

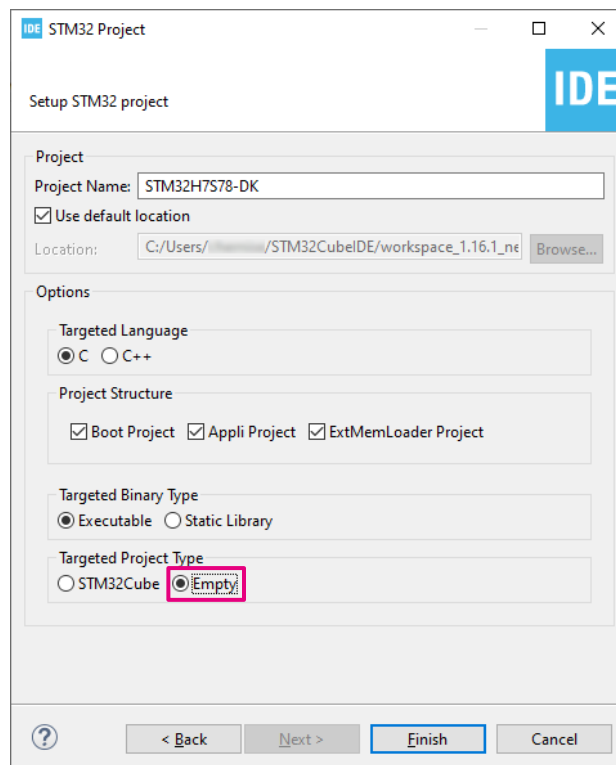
2.2 Create an empty project based on the template in the STM32CubeH7RS MCU Package

To start a new project, go to [File]>[New]>[STM32 Project] as shown in Figure 1. [New STM32 project](#).

Select the desired MCU or board. In the example illustrated in Figure 2. [Target selection](#), the selected board is the [STM32H7S78-DK](#). Click on [Next >].

After the target selection comes the project setup step shown in Figure 4. The *Targeted Project Type* setting determines whether the project gets generated by STM32CubeMX or not. An *Empty* project is a skeleton of a project that needs building upon while *STM32Cube* indicates an STM32CubeMX-managed project.

Figure 4. Project setup (empty project creation)



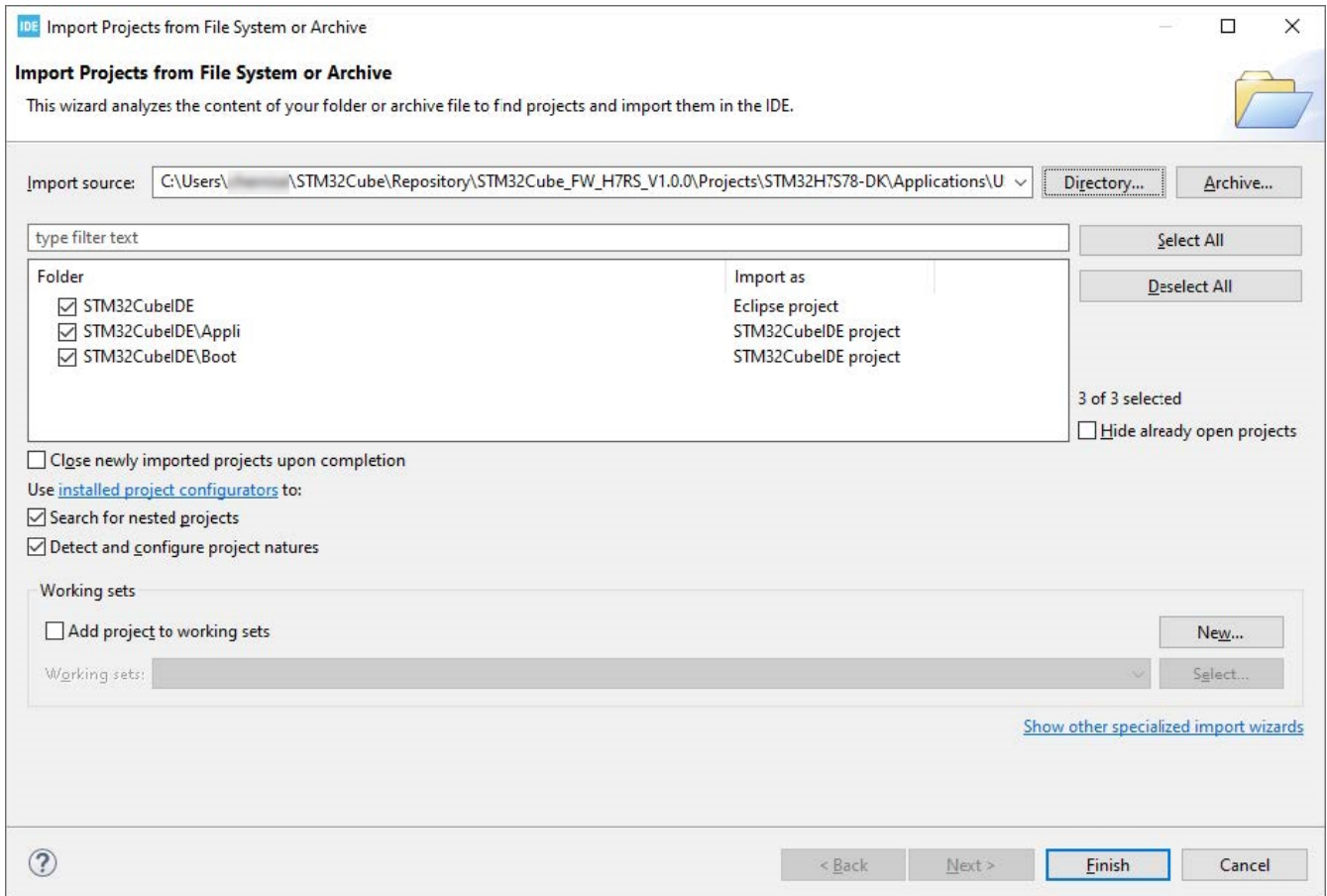
DT73575V1

2.3 Import a project from the STM32CubeH7RS MCU Package

To import the STM32Cube firmware project into STM32CubeIDE, go to **[File]>[Open Projects from File System...]**.

Then, select the corresponding project. A project example is by default located at `$Home\STM32Cube\Repository\STM32Cube_FW_H7RS_V1.0.0\Projects\STM32H7S78-DK\Applications\USB_Device\MSC_Standalone\STM32CubeIDE`.

Figure 5. Firmware project selection



After selecting the project, click on **[Finish]** to import and build the project.

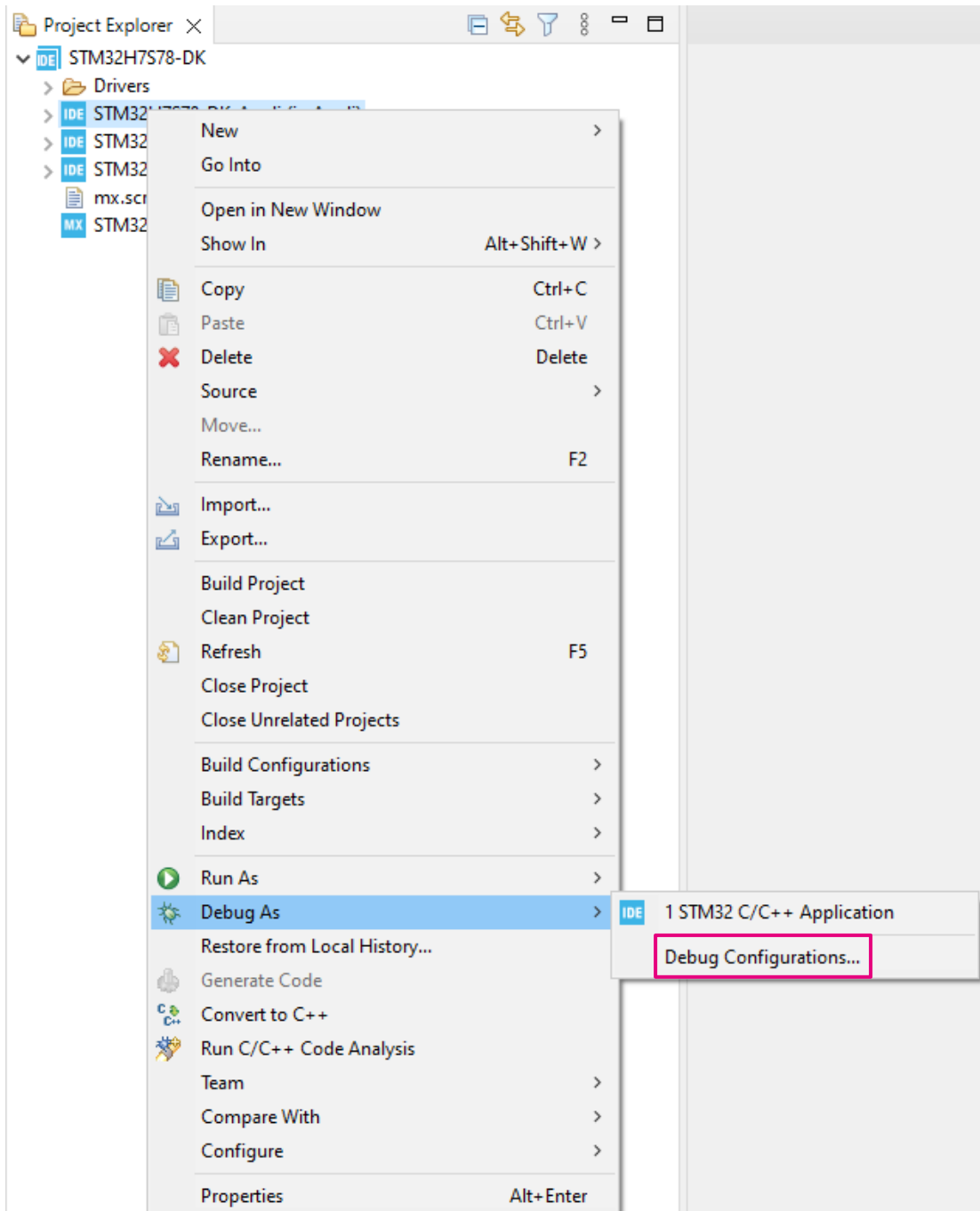
DT73576V1

3 Debugging

This chapter highlights some of the points to bear in mind while debugging an STM32H7Rx/7Sx microcontroller. In the next two sections, this application note covers the configurations needed to start debug sessions with ST-LINK GDB server and OpenOCD.

To start configuring the launch of the debug session, right-click the subproject (in this example, the STM32H7S78-DK_Appli) and select **[Debug As]>[Debug Configurations...]** as shown in Figure 6.

Figure 6. ST-LINK GDB server debug configuration (1 of 4) and OpenOCD debug configuration (1 of 2)



DT73577V1

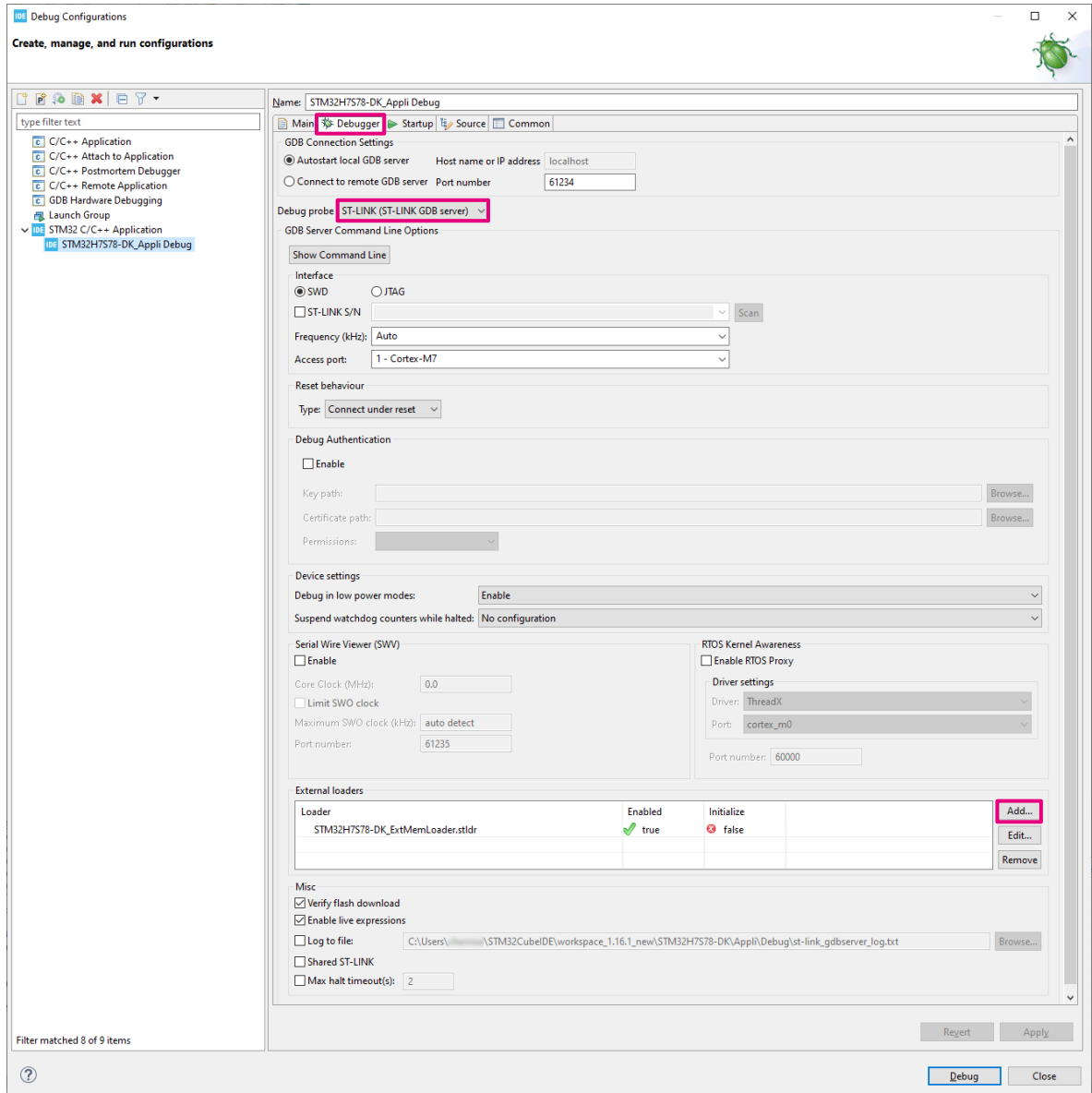
Note: For the rest of this chapter, examples are based on the STM32H7S78-DK board.

3.1 Setting up with ST-LINK GDB server

By default, the internal flash memory is used to load the program binary. However, due to the small size of the internal flash memory (64 Kbytes), the external memory can be used instead. Multiple linker files are generated for use according to the memory type desired. If the external memory is used, an external flash loader must be set in the debug configuration as shown in Figure 7.

To be able to debug and run the application, the boot must be programmed first in the device memory. In the debug configuration, switch to the *Startup* tab and select [Add...].

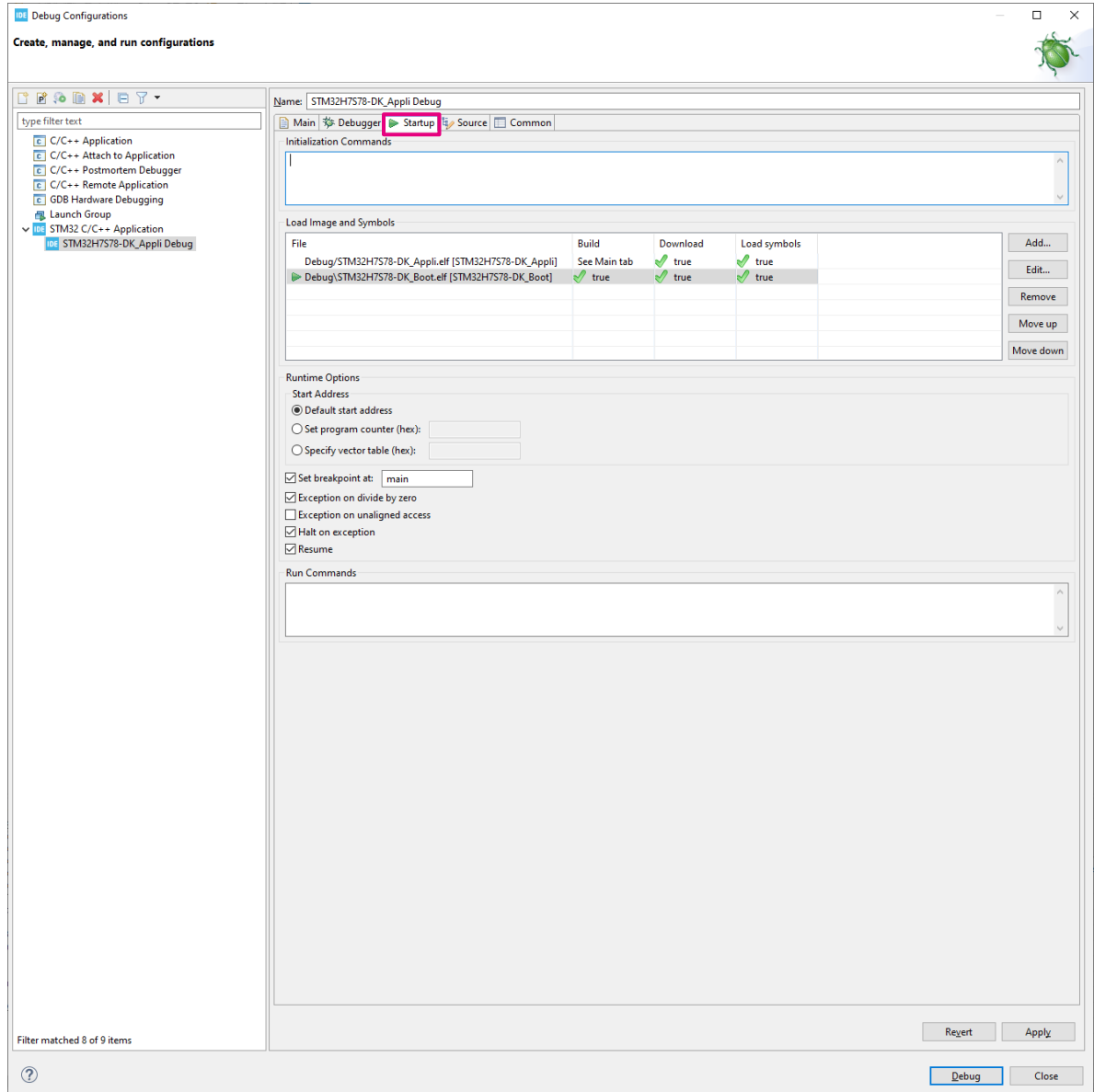
Figure 7. ST-LINK GDB server debug configuration (2 of 4)



DT73578V1

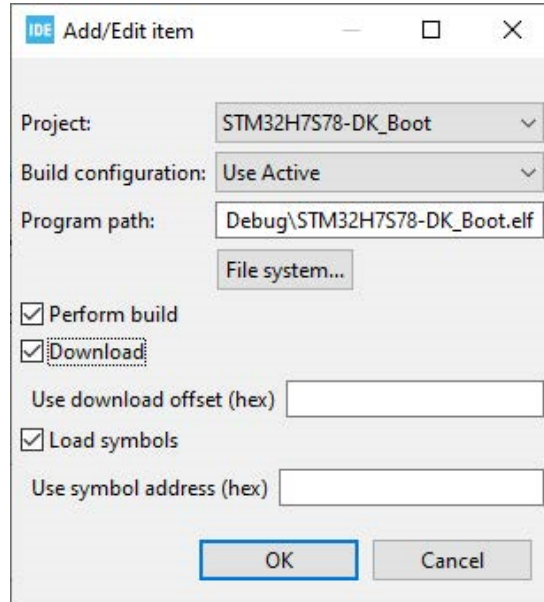
The STM32H7S78-DK_Appli configuration is responsible for loading both the STM32H7S78-DK_Appli and STM32H7S78-DK_Boot images. Go to the *Startup* tab to do this as shown in Figure 8:

Figure 8. ST-LINK GDB server debug configuration (3 of 4)



To download also the STM32H7S78-DK_Boot image, click on **[Add...]**, browse the correct project, and build the configuration. The result is shown in [Figure 9](#).

Figure 9. ST-LINK GDB server debug configuration (4 of 4)



DTT73580V1

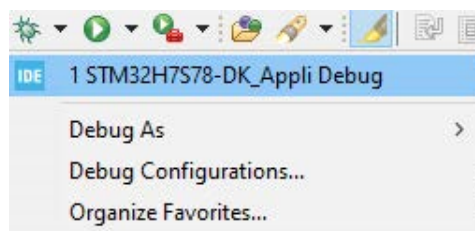
The order in the load list is very important. The last entry in the list, marked by a green arrow (refer to [Figure 8](#)), is the image debugged with this debug configuration. Consequently, the debugger fetches the program counter value (PC) from this image.

The configuration is complete.

3.1.1 Launching the configurations (ST-LINK GDB server)

To debug the project, launch the STM32H7S78-DK_Appli debug session. The execution starts at the beginning of the main function from the boot project. Then, it is possible to jump to the application.

Figure 10. ST-LINK GDB server debug configuration launch

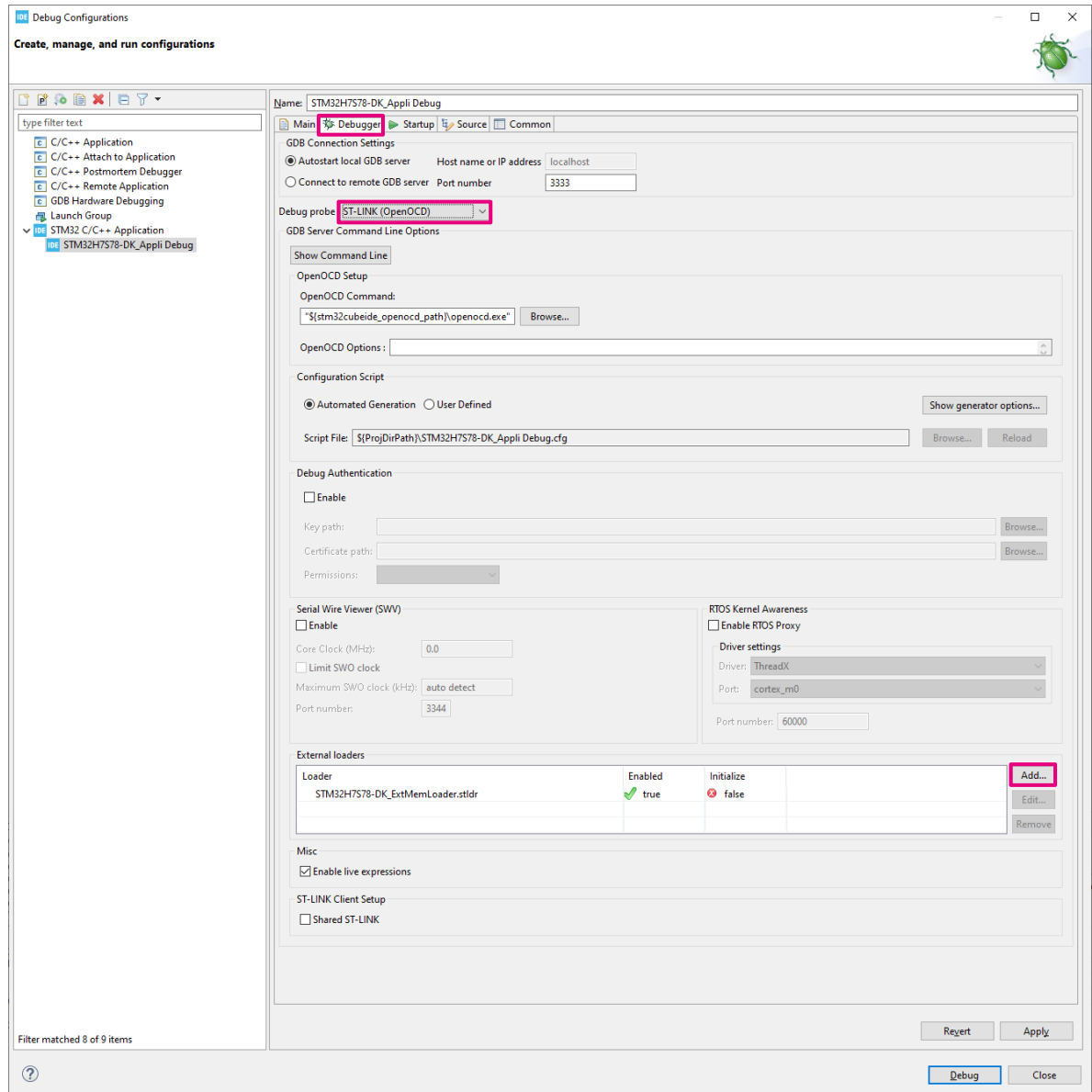


DTT73581V1

3.2 Setting up with OpenOCD

Select *ST-LINK (OpenOCD)* as the [Debug probe] in the debug configuration of the STM32H7S78-DK_Appli.

Figure 11. OpenOCD debug configuration (2 of 2)



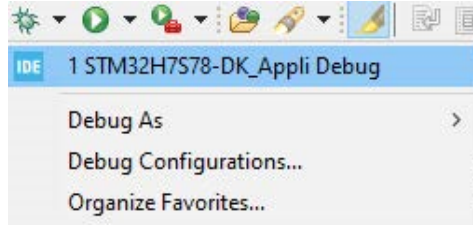
The rest of the configuration is the same as for debugging with ST-LINK GDB server (refer to Figure 7).

DTT3562V1

3.2.1 Launching the configurations (OpenOCD)

The same approach is used as with ST-LINK GDB server (refer to [Section 3.1.1](#)).

Figure 12. OpenOCD debug configuration launch



DT73583V1

Revision history

Table 2. Document revision history

Date	Revision	Changes
19-Feb-2025	1	Initial release.

Contents

1	General information	2
1.1	Prerequisites	2
1.2	The use cases in this document	2
2	Create and import projects	3
2.1	Create a new STM32 project	3
2.2	Create an empty project based on the template in the STM32CubeH7RS MCU Package	6
2.3	Import a project from the STM32CubeH7RS MCU Package	7
3	Debugging	8
3.1	Setting up with ST-LINK GDB server	9
3.1.1	Launching the configurations (ST-LINK GDB server)	11
3.2	Setting up with OpenOCD	12
3.2.1	Launching the configurations (OpenOCD)	13
	Revision history	14
	List of tables	16
	List of figures	17

List of tables

Table 1.	Applicable products	1
Table 2.	Document revision history	14

List of figures

Figure 1.	New STM32 project.	3
Figure 2.	Target selection	4
Figure 3.	Project setup (STM32CubeMX-managed project creation)	5
Figure 4.	Project setup (empty project creation)	6
Figure 5.	Firmware project selection	7
Figure 6.	ST-LINK GDB server debug configuration (1 of 4) and OpenOCD debug configuration (1 of 2)	8
Figure 7.	ST-LINK GDB server debug configuration (2 of 4)	9
Figure 8.	ST-LINK GDB server debug configuration (3 of 4)	10
Figure 9.	ST-LINK GDB server debug configuration (4 of 4)	11
Figure 10.	ST-LINK GDB server debug configuration launch	11
Figure 11.	OpenOCD debug configuration (2 of 2)	12
Figure 12.	OpenOCD debug configuration launch.	13

IMPORTANT NOTICE – READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgment.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, refer to www.st.com/trademarks. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2025 STMicroelectronics – All rights reserved