



How to use

STM32Cube™ with Ride7

Application Note AN0063

Document version
7 November 2014

Contents

1. INTRODUCTION.....	4
1.1 Purpose of this manual.....	4
1.2 Scope of this manual.....	4
1.3 Reference documents.....	4
1.4 Additional help or information.....	4
1.5 Raisonance brand microcontroller application development tools.....	5
1.6 Conventions used in this manual.....	5
2. USING THE STM32CUBEMX TOOL.....	6
2.1 Using a demonstration board from STMicroelectronics.....	6
2.2 Using any board.....	7
3. CREATING THE RIDE7 PROJECT.....	8
3.1 Launching Ride7.....	8
3.2 Converting the Keil project to a Ride project	8
3.3 Building the project.....	8
3.4 Troubleshooting.....	8
3.5 Programming your application on your STM32 microcontroller.....	10
4. CONCLUSION.....	11
5. CONFORMITY.....	12
6. GLOSSARY.....	13

7. INDEX.....	14
8. HISTORY.....	15

1. Introduction

The Ride7 and RKit-ARM software toolsets provide everything you need to program your STM32 microcontroller and debug applications, including:

- RLink and ST-Link programming and debugging.
- Ride7 integrated development environment for code editing, device programming and application debugging.
- GNU C /C++ compiler (unlimited compiling).

1.1 Purpose of this manual

This document guides you through the construction of your STM32 application using the STMCube™ tool.

1.2 Scope of this manual

This guide should be used by anyone who is interested in building an application using the STMCube™ tool. This document assumes that Ride7 and RKit-ARM are already installed on your PC. If this is not the case, please go to the <http://www.raisonance.com> web site and download them.

1.3 Reference documents

From STMicroelectronics: UM1718 STM32CubeMX for STM32 configuration and initialization C code generation

1.4 Additional help or information

If you want additional help or information, if you find any errors or omissions, or if you have suggestions for improving this manual, go to the KEOLABS' site for Raisonance microcontroller development tools www.raisonance.com, or contact the microcontroller support team.

Microcontroller website:	www.raisonance.com
Support extranet site:	support-raisonance.com (software updates, registration, bugs database, etc.)
Support Forum:	forum.raisonance.com/index.php
Support Email:	support@raisonance.com

1.5 Raisonance brand microcontroller application development tools

January 1, 2012, Raisonance became the brand under which the company KEOLABS sells its microcontroller hardware and software application development tools.

All Raisonance branded products regardless of their date of purchase or distribution are licensed to users, supported and maintained by KEOLABS in accordance with the companies' standard licensing maintenance and support agreements for its microcontroller application development tools. For information about these standard agreements, go to:

Support and Maintenance Agreement: <http://www.raisonance.com/warranty.html>

End User License Agreement: <http://www.raisonance.com/software-license.html>

1.6 Conventions used in this manual

File New	Refers to the menu item “New” on the File menu.
While(1);	(bold, monospaced type) User input
filename	Replace the italicized text with the item it represents
[]	Items inside [and] are optional.
[...]	Represents a list of optional items that are the same as the preceding item.
while(1);	(monospaced type) Code, Directives and software generated output
0000H	A Hexadecimal value (assembly format)
0x0000	A Hexadecimal value (C language format)

The Ride installation directory is referred to as “%ridedir%” in this document.
If you installed Ride in the default “C:\Program Files\Raisonance\Ride” directory, “%ridedir%\lib” will refer to the “C:\Program Files\Raisonance\Ride\lib” directory.

2. Using the STM32CubeMX tool

The STM32CubeMX is part of STMicroelectronics' STMCube™ tool which reduces development efforts, time and costs when developing STM32 microcontrollers.

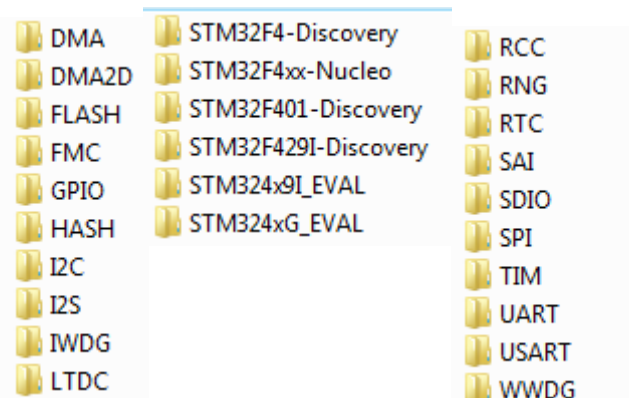
The STMCube™ tool includes:

- STM32CubeMX: a graphical configuration tool that generates C initialization code from graphics wizards.
- STM32CubeXX: a software platform customized per series (such as STM32CubeF4 for STM32F4 series), ensuring maximal portability:
 - STM32Cube HAL (an STM32 abstraction layer embedded software,)
 - Consistent middleware components (RTOS, USB, TCP/IP and graphics).
 - Full set of examples.

2.1 Using a demonstration board from STMicroelectronics

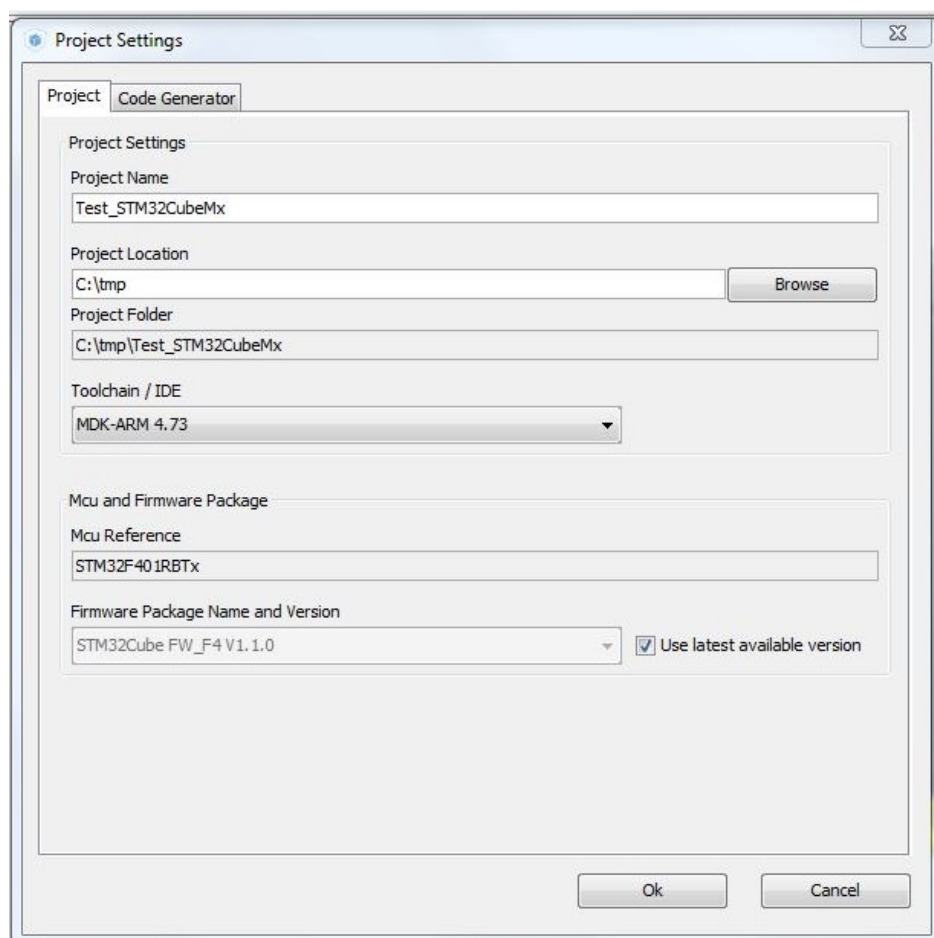
The STM32CubeXX features many examples that explain how to use HAL drivers and/or middleware components that can be run on any STMicroelectronics board.

1. Find your board in STM32Cube_FW_XXXX\Projects (F4 in this example)
2. Choose an example for your project.
3. Create the Ride7 project from the Keil project as explained the next section.



2.2 Using any board

1. Go to www.st.com/stm32cube, click on **Download** at the bottom of the page.
2. Configure your device using UM1718: STM32CubeMX for STM32 configuration and initialization C code generation (available from www.st.com).
3. Export your finished configuration to your IDE.
4. Configure the project: project name, project location, choice of Integrated Development Environment tools, and C code generation options.
5. Chose the IDE **MDK-ARM** (Keil) to create your project.
6. Create the Ride7 project from the Keil project, as explained in the next section.



3. Creating the Ride7 project

As the STM32cube tool does not save projects in Ride format *.rprj, you must open the Keil *.uvproj project and Ride converts it, as explained in this chapter.

3.1 Launching Ride7

On your computer select **Start | Programs | Raisonance Tools | Ride 7**, and click on the **Ride** icon.

3.2 Converting the Keil project to a Ride project

1. From the top menu in Ride7, click on the **Project | Open**.
2. Click on **Type of files** and choose the Keil uVision project.
3. Select the directory of your project.
4. Open the project.

Ride7 prepares the project for your application.

3.3 Building the project

Simply click on the command **Project | Build Project** from the top menu in Ride7. The Ride7 **Build log** window displays the compiler/linker results during the build operation, and then the **Build succeeded** message should appear.

3.4 Troubleshooting

Your converted Ride7 project may contain some incorrectly translated options. Here are the common problem areas:

- source files list
- target device name
- preprocessor defines
- include directories
- optimization level

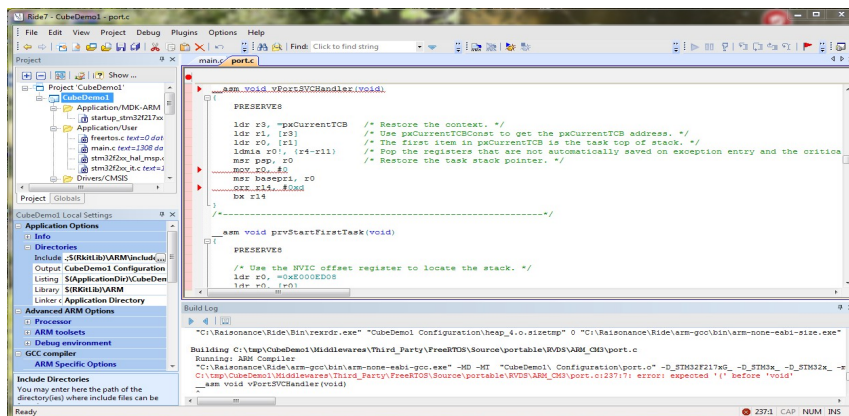
3.4.1 Middleware

Some middleware source files are not well recognized during compilation because they were written for Keil compilers, not for GCC compilers.

All assembler code for Keil projects must be replaced by assembler code for GCC compilers.

To better understand the issue have a look at this example:

1. As explained above, convert the following Keil project: STM32Cube_FW_F2_V1.1.0\Projects\STM322xG_EVAL\Applications\FreeRTOS\FreeRTOS_Queue.
2. Build the project (Ctrl + F9).
3. A compilation error will appear in the **port.c** source file (assembler code).

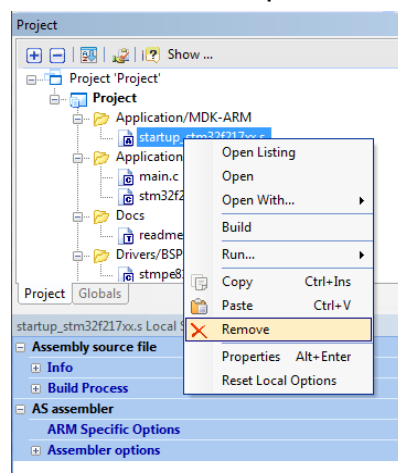


4. Replace this file by the one you find at STM32Cube_FW_F2_V1.1.0\Middlewares\Third_Party\FreeRTOS\Source\portable\GCC\ARM_CM3\port.c
5. Build with success.

3.4.2 Startup

Before compiling the converted project with Ride7 you must delete the file startup_stm32fxxx.s and use the default GCC startup file provided in Ride7.

This unwanted file is produced by the conversion and contains Keil assembler code.



3.5 Programming your application on your STM32 microcontroller

To program your application on your STM32:

1. Connect RLink/ST-Link debugger to an available USB port on your PC.
2. Power-on your board.
3. Launch the debug session of your project by running the **Debug | Start** command in the main menu. This action erases the Flash and reloads it with the necessary information to debug your current application.
4. Run the program through the **Debug | Run** command.

See the GettingStartedARM documentation for more information about programming and debugging using Ride and RKit-ARM.

4. Conclusion

This tutorial was designed to introduce you to using STM32Cube tool with Ride7.

Further information and detailed documentation can be found at
<http://www.raisonance.com>.

5. Conformity



ROHS Compliance (Restriction of Hazardous Substances)

KEOLABS products are certified to comply with the European Union RoHS Directive (2002/95/EC) which restricts the use of six hazardous chemicals in its products for the protection of human health and the environment.

The restricted substances are as follows: lead, mercury, cadmium, hexavalent chromium, polybrominated biphenyls (PBB), and polybrominated diphenyl ethers (PBDE).



CE Compliance (Conformité Européenne)

KEOLABS products are certified to comply with the European Union CE Directive.

In a domestic environment, the user is responsible for taking protective measures from possible radio interference the products may cause.



FCC Compliance (Federal Communications Commission)

KEOLABS products are certified as Class A products in compliance with the American FCC requirements. In a domestic environment, the user is responsible for taking protective measures from possible radio interference the products may cause.



WEEE Compliance (The Waste Electrical & Electronic Equipment Directive)

KEOLABS disposes of its electrical equipment according to the WEEE Directive (2002/96/EC). Upon request, KEOLABS can recycle customer's redundant products.

For more information on conformity and recycling, please visit the KEOLABS website www.keolabs.com

6. Glossary

Term	Description
RFlasher	Raisonance Flasher: Programming interface for user-friendly flash programming
Ride7	Raisonance Integrated Development Environment

7. Index

Build Project.....	9	Lead.....	12
CE.....	12	Purpose of this manual.....	4
Compliance.....	12	ROHS.....	12
Conformity.....	12	Scope of this manual.....	4
Directive.....	12	WEEE.....	12
FCC.....	12		

8. History

Date	Description
07 Nov 14	Creation of the tutorial



Disclaimer

Information in this document is subject to change without notice and does not represent a commitment on the part of the manufacturer. The software described in this document is provided under license and may only be used or copied in accordance with the terms of the agreement. It is illegal to copy the software onto any medium, except as specifically allowed in the license or nondisclosure agreement.

No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information storage and retrieval systems, for any purpose other than the purchaser's personal use, without prior written permission.

Every effort has been made to ensure the accuracy of this manual and to give appropriate credit to persons, companies and trademarks referenced herein.

This manual exists both in paper and electronic form (pdf).

Please check the printed version against the .pdf installed on the computer in the installation directory of the most recent version of the software, for the most up-to-date version.

The examples of code used in this document are for illustration purposes only and accuracy is not guaranteed. Please check the code before use.

Copyright © KEOLABS 2014 All rights reserved