# UM1962
# User manual

## Getting started with STM32F469I discovery software development tools

## Introduction

This document describes some software environment recommendations, required to build an application using the STM32F469I Discovery kit (32F469IDISCOVERY).

The document provides users with guidelines about how to build and run a simple example and how to create and build an application. It has the following structure:

- The first chapter presents the software and hardware requirements (some toolchains supporting the STM32 families, ST-LINK/V2-1 installation and firmware package presentation).
- The second chapter provides step by step guidelines on how to execute and debug an application example using the following toolchains:
  - IAR Embedded Workbench® for ARM® (EWARM) by IAR Systems®
  - Microcontroller development kit for ARM® (MDK-ARM) by Keil®
  - System Workbench for STM32 (SW4STM32) by AC6

Although this user manual does not cover all the topics relevant to software development environment, it demonstrates the first basic steps necessary to get started with the compilers/debuggers and it includes references for complementary information.

# Contents

# List of tables

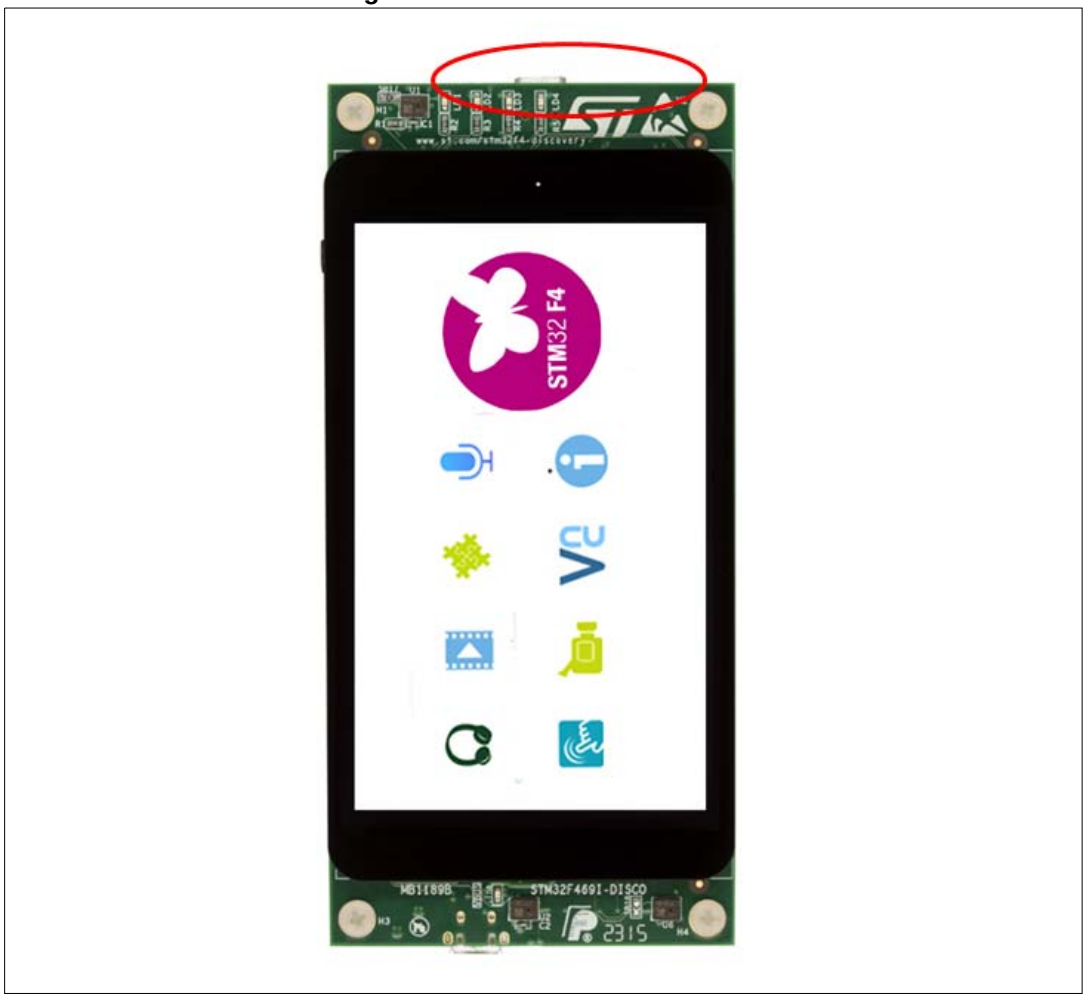
# List of figures

# 1 System requirements

Before running an application, the user should:

1. Install his preferred Integrated Development Environment (IDE).
2. Install the ST-LINK V2-1 driver from the *www.st.com* web page.
3. Download the STM32F469I Discovery firmware from the *www.st.com* web page.
4. Establish the USB connection with the STM32F469 Discovery board by connecting the CN1 connector of the STM32F469 Discovery board to the USB port of the PC, using a type A to Mini-B USB cable (the *Figure 1* shows the connector CN1 inside the red circle).

**Figure 1. Hardware environment**




The above steps will be detailed in the following sections.

# 2 IDEs supporting STM32 families

STM32 families of 32-bit ARM® Cortex® -M core-based microcontrollers are supported by a complete range of software tools. It encompasses traditional integrated development environments IDEs with C/C++ compilers and debuggers from major 3rd-parties (free versions up to 64 Kbytes of code, depending on 3rd-parties), completed with innovative tools from STMicroelectronics.

The following table regroups general information about the three integrated development environments, as well as the version supporting the STM32F469I product.

**Table 1. Toolchains supporting STM32F469I Discovery kit**

| Toolchain | Company | Compiler | Version | Download link(1) |
|---|---|---|---|---|
| EWARM | IAR Systems® | IAR C/C++ | 7.40.4 and later | http://www.iar.com<br>30-day evaluation edition<br>KickStart edition (32-KByte code size limitation) |
| MDK-ARM | Keil® | ARMCC | 5.xx and later with *STM32F4xx_DFP2.6.0.pack* | http://www.keil.com<br>MDK-Lite (32-KByte code size limitation) |
| SW4STM32 | AC6 | GNU C | 1.4 and later | http://www.openstm32.org<br>Free version: no limitation |

1. Registration before downloading is required.

# 3 ST-LINK/V2-1 installation

The STM32F469I Discovery board includes an embedded ST-LINK/V2 debug tool interface. The interface needs an ST-LINK/V2 dedicated USB driver to be installed. This driver is available from the *www.st.com* web page and it is supported by the software toolchains:

- IAR Embedded Workbench for ARM (EWARM).

  The toolchain is installed by default in the C:\Program Files\IAR Systems\Embedded Workbench x.x directory on the hard disk of the local PC.

  After having installed EWARM, install the ST-LINK/V2 driver by running the ST-Link_V2_USB.exe from IAR_INSTALL_DIRECTORY]\Embedded Workbench x.x\arm\drivers\ST-LINK \ST-Link_V2_USBdriver.exe.
- Keil Microcontroller Development Kit (MDK-ARM) toolchain.

  The toolchain is installed by default in the C:\Keil directory on the hard disk of the local PC; the installer creates a ARM® Keil® µVision®4 shortcut in the start menu.

  When connecting the ST-LINK/V2 tool, the PC detects new hardware and prompts user to install the ST-Link_V2_USB driver. The “Found New Hardware” wizard displays and guides the user through the steps, required to install the driver from the recommended location.
- AC6 System Workbench for STM32 (SW4STM32).

  The toolchain is installed by default in the C:\Program Files\AC6 directory on the hard disk of the local PC.

  The ST-Link_V2_USB.exe is automatically executed, when installing the software toolchain.

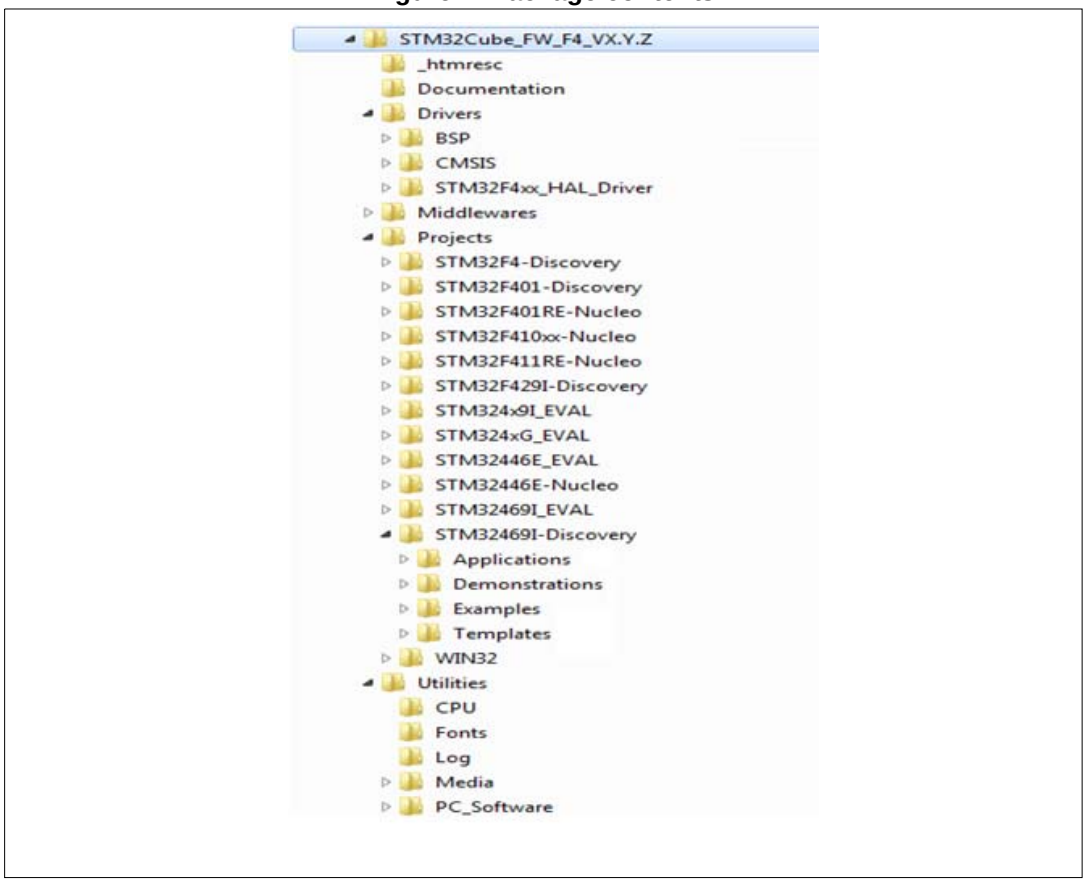
For complementary information on the firmware package content and the STM32F469I Discovery requirements, refer to: *STM32CubeF4 demonstration platform* (UM1743 user manual).

*Note: The embedded ST-LINK/V2-1 supports only the SWD interface for STM32 devices.*

# 4 Firmware package

The STM32F469I Discovery firmware applications, demonstration and examples are provided in one single.zip file. The extraction of the zip file generates a folder named “STM32Cube_FW_F4_VX.Y.Z”, which contains the subfolders shown in *Figure 2*.

**Figure 2. Package contents**




## Subfolders description

**Template**: pre-configured project with empty main function to be customized. This is helpful to start creating your own application.

**Peripheral examples**: including set of examples for each STM32F469 peripheral ready to be run.

**Applications**: including set of applications for each STM32F469 peripheral ready to be run.

**Demonstrations**: including demonstration firmware for STM32F469 boards ready to be run.

# 5 Compiling, linking and executing firmware using software toolchains

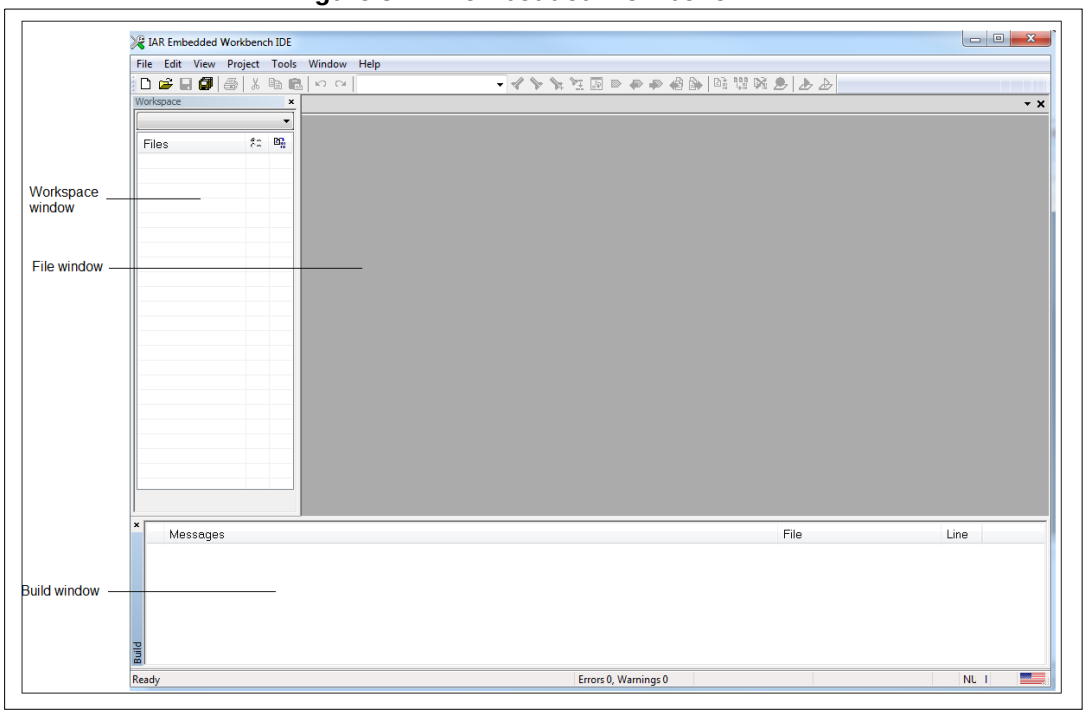
The below steps can be applied to an already existing example, demonstration or template project available inside the STM32F469I Discovery package firmware from the *www.st.com* web site.

First of all, the user must read the readme.txt file, which contains a description of the firmware and hardware/software requirements.

## 5.1 EWARM toolchain

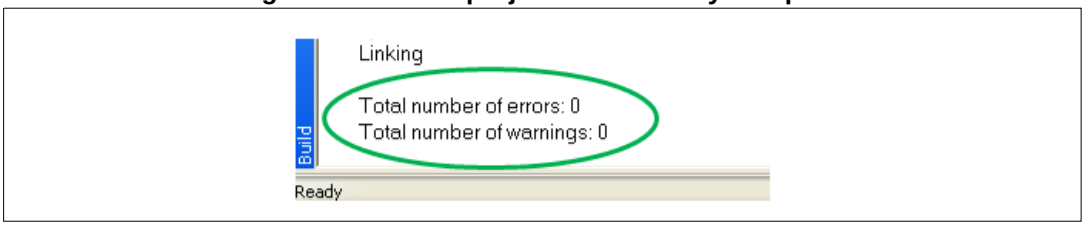
1. Open the IAR Embedded Workbench for ARM (EWARM). *Figure 3* shows the names of the windows, to which this document refers.

**Figure 3. IAR embedded workbench IDE**




2. In the **File** menu, select **Open** and click **Workspace,** to display the Open Workspace dialog box. Browse to select either an example or demonstration or template workspace file, and click Open to launch it in the Project window.
3. In the Project menu, select Rebuild All to compile the project.
4. If the **Project** is successfully compiled, the window, shown in *Figure 4: EWARM project successfully compiled*, is displayed.

**Figure 4. EWARM project successfully compiled**




To change the project settings (Include and preprocessor defines), simply go through the project options:

- For Include directories

  Project>Options…>C/C++ compiler>
- For pre-processor defines

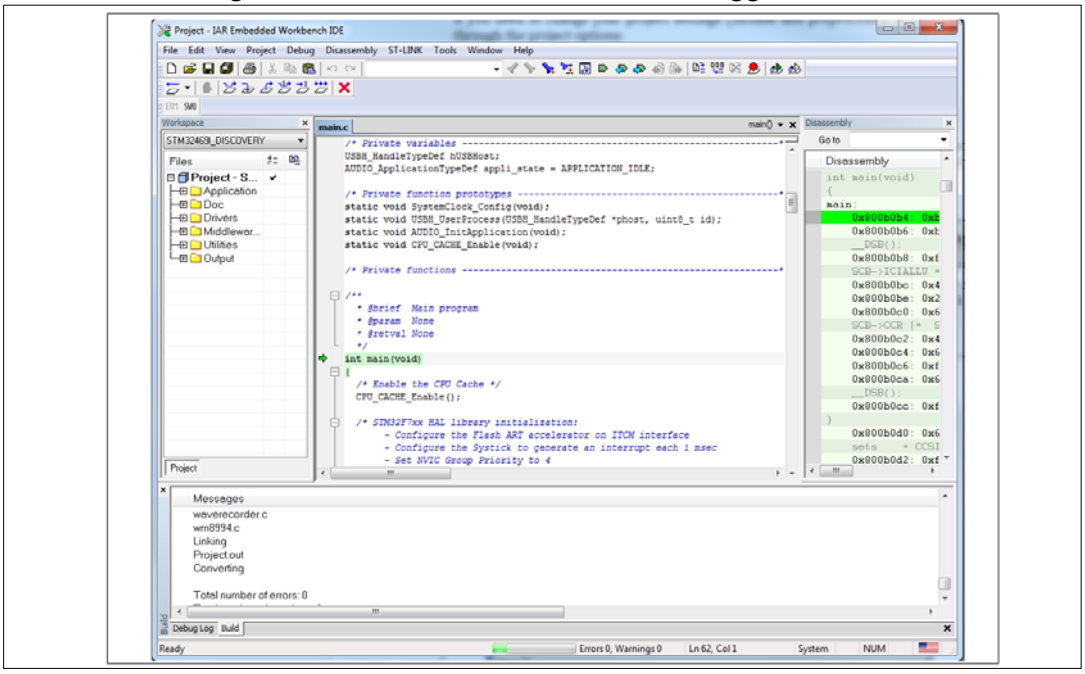
  Project>Options…C/C++ compiler>pre-processor>

5. In the IAR Embedded Workbench IDE, from the Project menu, select Download and Debug or, alternatively, click the Download and Debug button in the toolbar, to program the Flash memory and begin debugging (see *Figure 5*).

**Figure 5. Download and debug button**


6. The debugger in the IAR Embedded Workbench can be used to debug source code at C and assembly levels, to set breakpoints, to monitor individual variables and watch events during the code execution (see *Figure 6*).

**Figure 6. IAR embedded workbench debugger screen**



To run the application, from the Debug menu, select Go. Alternatively, click the Go button in the toolbar to run the application (see *Figure 7*).

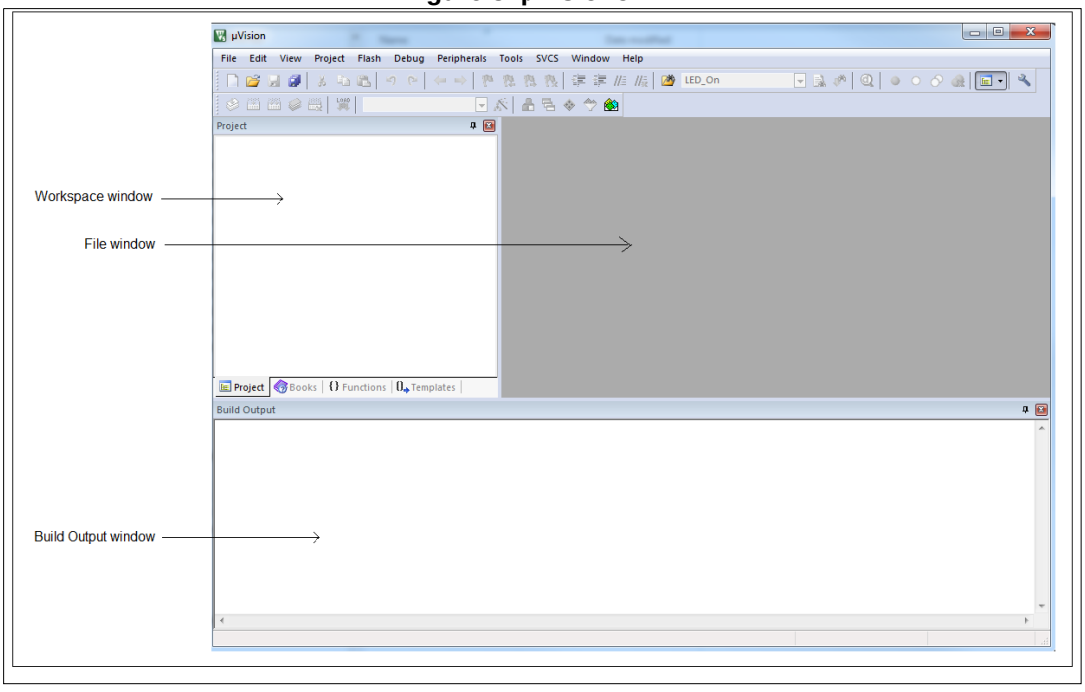
**Figure 7. IAR Go button**


## 5.2 MDK-ARM toolchain

1. Open the Keil MDK-ARM Microcontroller Kit.

*Figure 8* shows the basic names of the “Keil µVision5” windows to which this document refers.

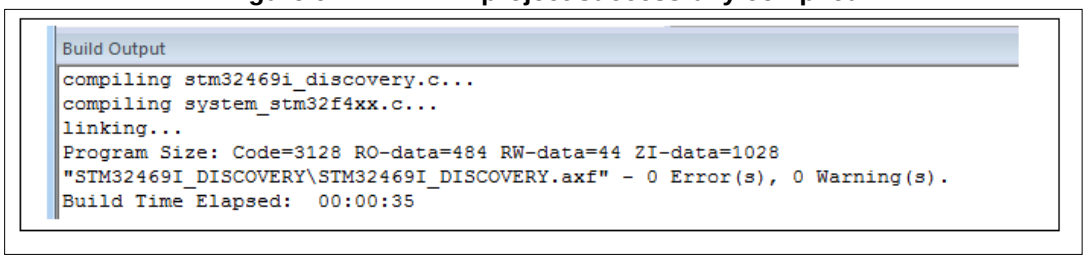
**Figure 8. µVision5 IDE**




2. In the **Project** menu, select **Open Project...** Browse to select either an example or demonstration or template project file, and click **Open** to launch it in the Project window.
3. In the **Project** menu, select **Rebuild All target files** to compile the project.

4. If the project is successfully compiled, the following window in *Figure 9* is displayed.

**Figure 9. MDK-ARM project successfully compiled**




To change the project settings (Include and preprocessor defines), simply go through the project options:

- For Include directories

  Project>Options for Target > C/C++ > Include Paths

- For pre-processor defines

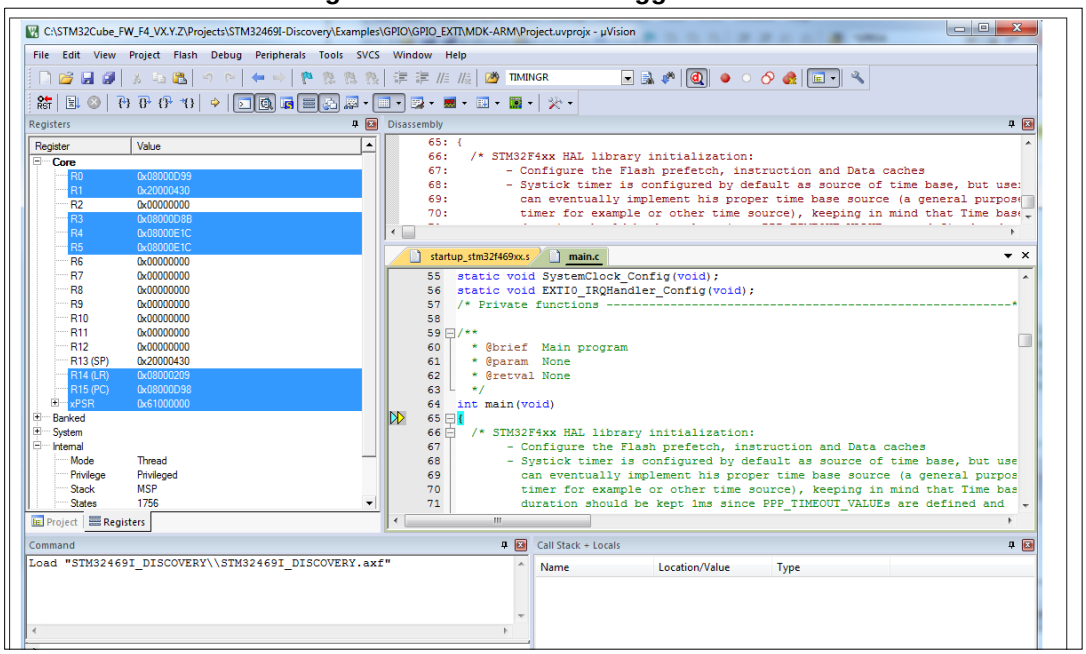
  Project>Options for Target > C/C++ > Preprocessor symbols > Define

5. In the MDK-ARM IDE, from the Debug menu, select Start/Stop Debug Session or, alternatively, click the Start/Stop Debug Session button in the toolbar, to program the Flash memory and begin debugging (see *Figure 10*).

**Figure 10. Start/Stop debug session button**


6. The MDK-ARM debugger can be used to debug source code at C and assembly levels, to set breakpoints, to monitor individual variables and to watch events during the code execution (see *Figure 11*).

**Figure 11. MDK-ARM debugger screen**



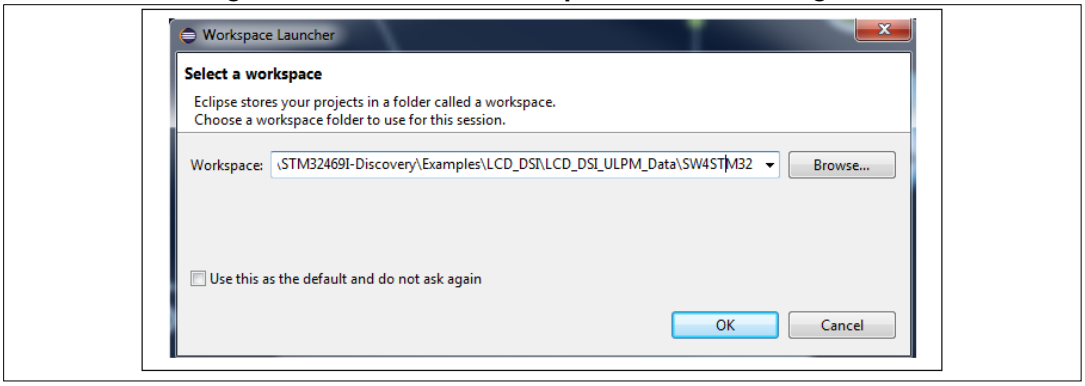

To run the application, from the Debug menu, select Run. Alternatively, click the Run button in the toolbar to run the application (see *Figure 12*).

**Figure 12. Run button**

## 5.3 SW4STM32 toolchain

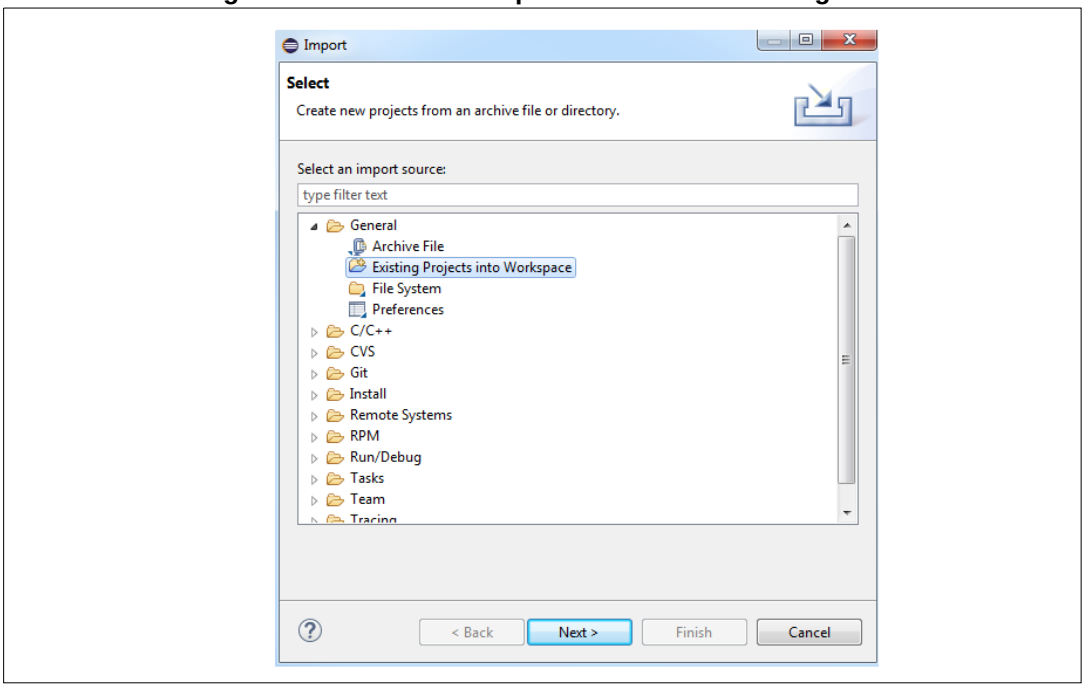
1. Open the AC6 SW4STM32 for STM32 products. The program launches and prompts for the workspace location (see *Figure 13*).

**Figure 13. SW4STM32 workspace launcher dialog box**



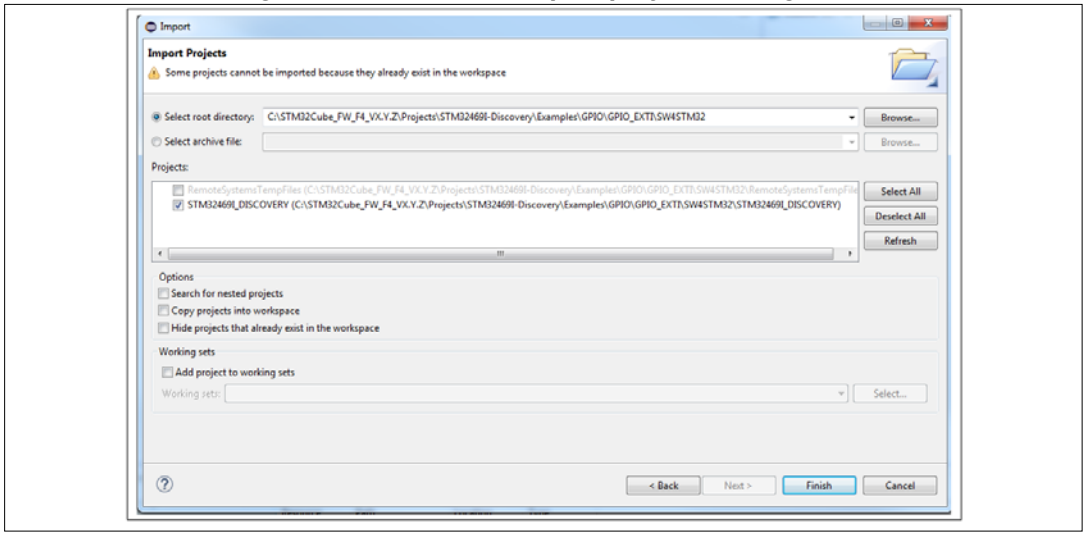

2. Browse to select a SW4STM32 workspace of either an example or demonstration or template workspace file and click OK to load it.
3. To load an existing project in the selected workspace, select Import from the File menu to display the Import dialog box.
4. In the Import window, open General, select Existing Projects into Workspace and click Next (see *Figure 14*).

**Figure 14. SW4STM32 import source select dialog box**



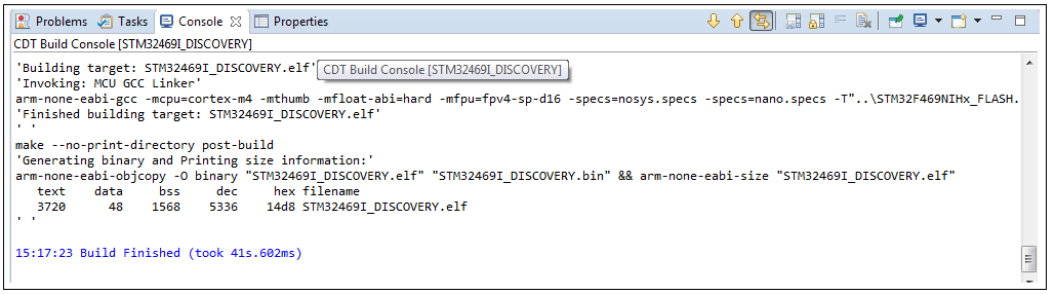
5. Click Select root directory, browse to the SW4STM32 workspace folder (see *Figure 15*).

**Figure 15. SW4STM32 import projects dialog box**




6. In the Projects panel, select the project and click Finish.
7. In the Project Explorer, select the project, open the Project menu, and click Build Project.
8. If the project is successfully compiled, the following messages is displayed on the Console window (see *Figure 16*).

**Figure 16. SW4STM32 project successfully compiled**




To change the project settings (Include directories and preprocessor defines), simply go through Project>Properties, select C/C++ Build>Settings from the left panel:

- For Include directories'

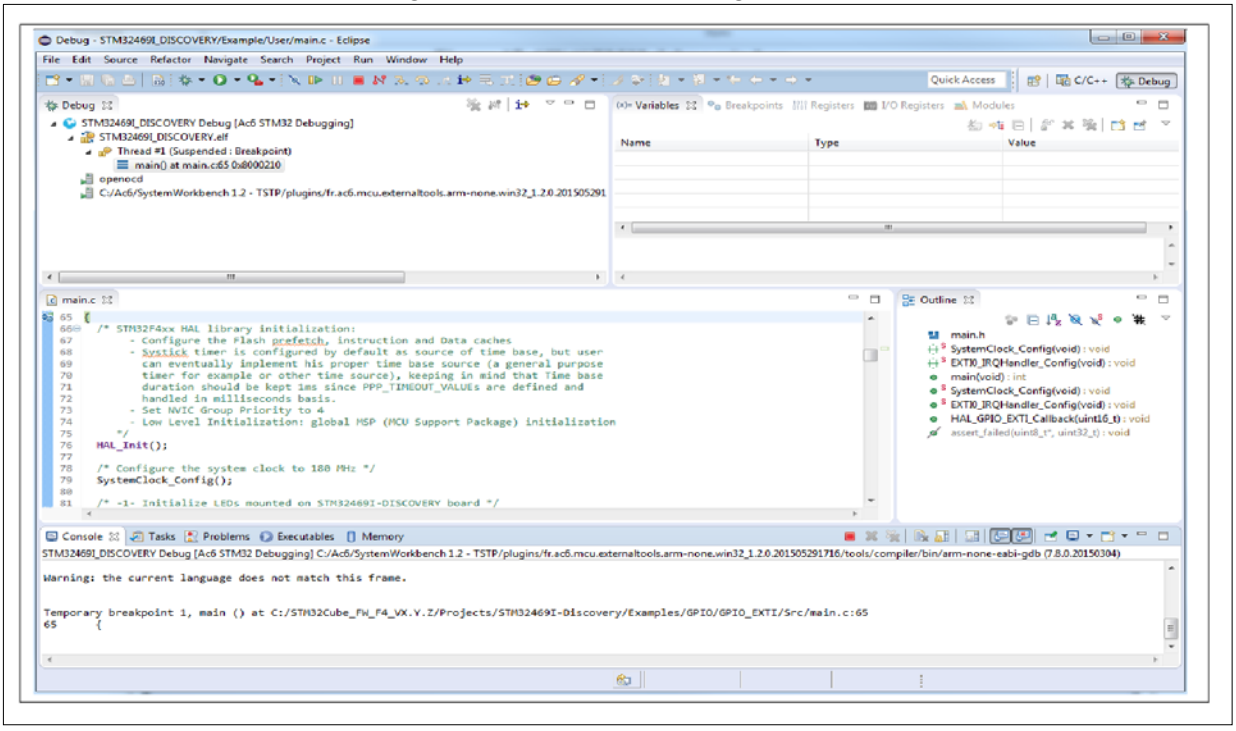
  C Compiler>Directories>Include path
- For pre-processor defines

  C Compiler>Symbols> Defined symbols

9. To debug and run the application, select the project In the Project Explorer and press F11 to start a debug session.

In the Project Explorer, select the project and press F11 to start a debug session (see *Figure 17*).

**Figure 17. SW4STM32 debug window**




The debugger in the AC6 SW4STM32 can be used to debug source code at the C and assembly levels, to set breakpoints, to monitor individual variables and to watch events during the code execution.

To run the application, from the Run menu, select Resume, or alternatively click the Resume button in the toolbar.

# 6 Revision history

Table 2. Document revision history

| Date | Revision | Changes |
|---|---|---|
| 30-Oct-2015 | 1 | Initial release. |