UM1671
User manual

Getting started with STM32F401 Discovery software development tools

Introduction

This document describes the software environment required to build an application around the STM32F401 Discovery board (32F401CDISCOVERY), and provides some development recommendations.

It provides guidelines to novice users on how to build and run a sample application, and allows them to create and build their own application.

This document:

- presents the toolchains supporting the STM32 families:
  - IAR Embedded Workbench® for ARM (EWARM) by IAR Systems,
  - Microcontroller Development Kit for ARM (MDK-ARM) by Keil™,
  - TrueSTUDIO® by Atollic;
- describes where to find the ST-LINK/V2 driver to be installed before starting coding on any Integrated Development Environment;
- describes step by step how to execute and debug an existing project with one of the previously presented toolchains;
- describes step by step how to create a new project with one of the toolchains from a dummy project included in the firmware package;
- provides helpful information on the advanced debugging capabilities of the Serial Wire Viewer (SWV) asynchronous trace supported by ST-LINK. This feature is especially useful when debugging certain tricky problems you may find during application development.

This user manual cannot cover all the topics relevant to software development environments, but it demonstrates the first basic steps necessary to get started with the compilers/debuggers and provides links to the required documents to fully understand every step.

Contents

List of tables

List of figures

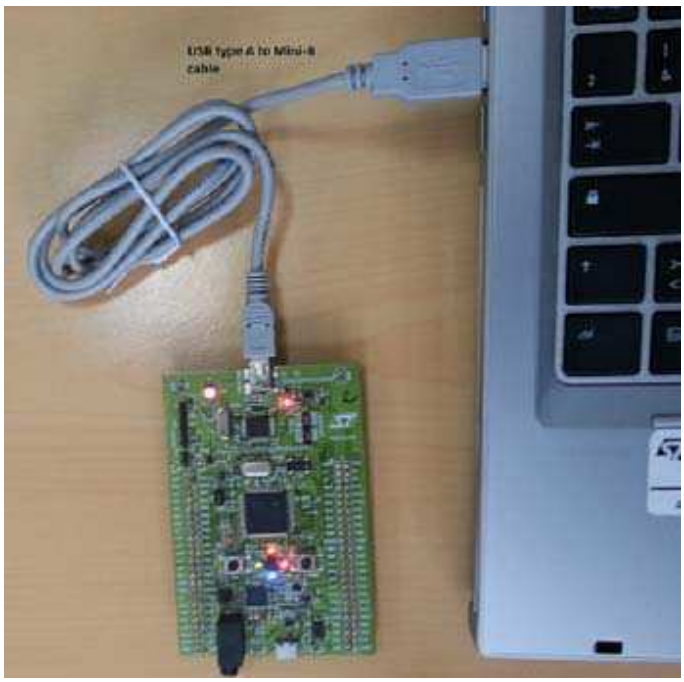
1 Getting started

1.1 System requirements

Before running your application, you should:

1. install your preferred Integrated Development Environment (IDE),
2. install the ST-LINK V2 driver from the ST Website,
3. download the STM32F401 Discovery firmware from the ST Website,
4. establish the connection with the STM32F401 Discovery board as follows.

Figure 1. Hardware environment



The above steps will be detailed in the following sections.

To run and develop any firmware application on your STM32F401 Discovery board, the minimum requirements are as follows:

- Windows PC (2000, XP, Vista, 7)
- “USB type A to Mini-B” cable, used to power the board (through USB connector CN1) from host PC and connect to the embedded ST-LINK/V2 for debugging and programming

1.2 IDEs supporting STM32 families

STMicroelectronics' 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 KB of code, depending on the partner), completed with innovative tools from STMicroelectronics.

The following table provides some general information about the most popular integrated development environments, as well as the version supporting officially the STM32F401 product.

Table 1. Toolchains supporting STM32F401 Discovery

Toolchain	Company	Compiler	Version	Download link(1)
EWARM	IAR Systems®	IAR C/C++	6.60.2 and later	www.iar.com/en/Products/IAR-Embedded-Workbench/ARM/ – 30-day evaluation edition – KickStart edition (32 Kb limitation for Cortex M3/M4) – KickStart edition (16 Kb limitation for Cortex M0)
MDK-ARM	Keil™	ARMCC	4.73 and later	www.keil.com/demo/eval/arm.htm MDK-Lite (32 Kb code size limitation)
TrueSTUDIO	©Atollic	GNUC	4.2 and later	www.atollic.com/index.php/request-eval-license – 32 Kb limitation (8 Kb on Cortex-M0 and Cortex-M1) – 30 day Professional version (trial)

1. Registration before download is required.

1.3 ST-LINK/V2 installation and development

The STM32F401 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 ST Website www.st.com ST-LINK V2 and 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 PC's local hard disk.

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

  The toolchain is installed by default in the C:\Keil directory on the PC's local hard disk; the installer creates a µVision4 shortcut in the Start menu.

  When connecting the ST-LINK/V2 tool, the PC detects new hardware and prompts the user to install the ST-LINK_V2_USB driver. The "Found New Hardware" Wizard displays and guides you through the steps required to install the driver from the recommended location.
- Atollic TrueSTUDIO® STM32

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

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

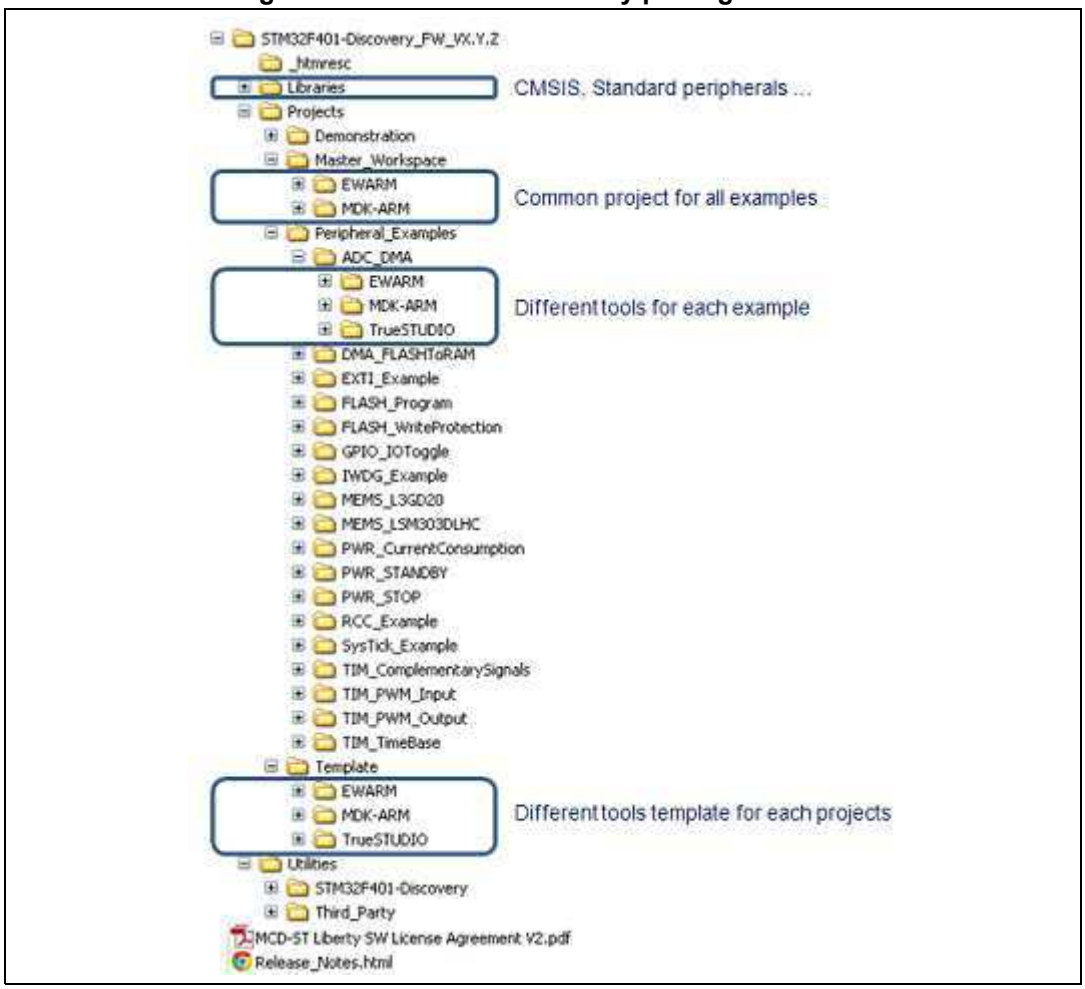
Complementary information on the firmware package content and the STM32F401 Discovery requirements is available in user manual UM1660 *Getting started with the STM32F401 Discovery kit*.

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

1.4 Firmware package

The STM32F401 Discovery firmware applications, demonstration and IP examples are provided in one single package and supplied in one single zip file. The extraction of the zip file generates a folder named "STM32F401-Discovery_FW_VX.Y.Z" which contains the following subfolders:

Figure 2. STM32F401 Discovery package content



Template project: pre-configured project with empty main function to be customized by the user. This is helpful to start creating your own application based on the peripheral drivers.

Master workspace: collection of all projects available within this firmware package.

Peripheral examples: includes a set of ready-to-run examples for each peripheral.

2 Executing and debugging firmware using a software toolchain

This section describes how to compile/link and execute an existing project using the toolchains.

The steps below can be applied to an already existing example, demonstration or template project available at STM32F401-Discovery_FW_VX.Y.Z firmware available at www.st.com/stm32f4-discovery.

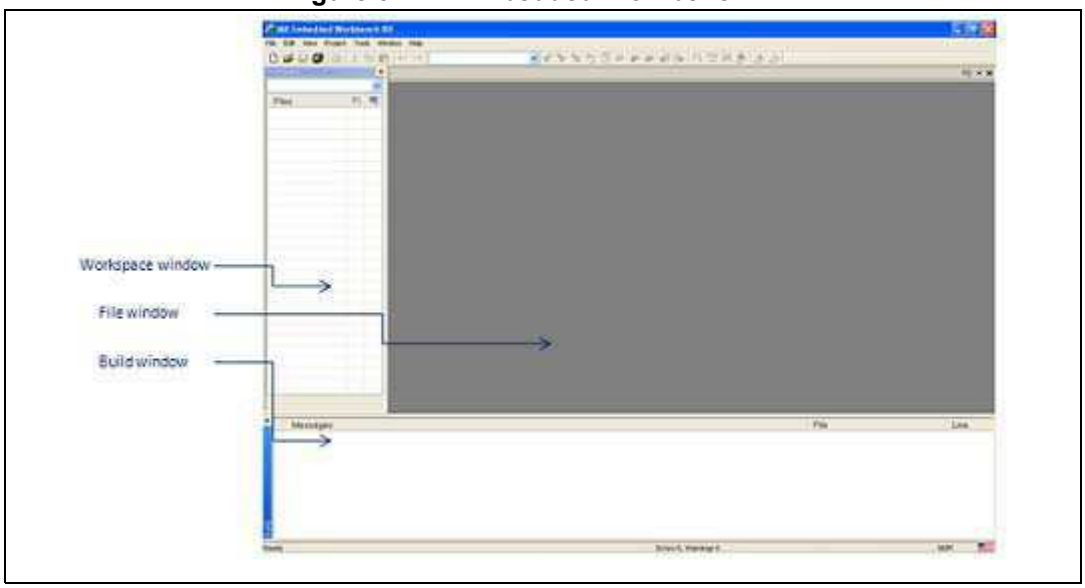
First of all, you need to go through the firmware readme.txt file which contains a description of the firmware and hardware/software requirements.

2.1 EWARM toolchain

1. Open the IAR Embedded Workbench® for ARM (EWARM).

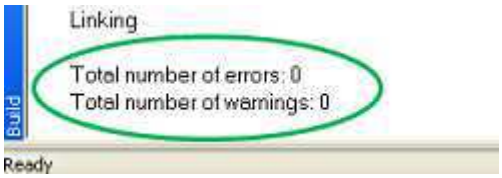
Figure 3 shows the basic names of the windows referred to in this document.

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 your project.
4. If your project is successfully compiled, the window shown in *Figure 4* is displayed.

Figure 4. EWARM project successfully compiled



If you need to change your project settings (Include and preprocessor defines), simply go through the project options:

- For Include directories

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

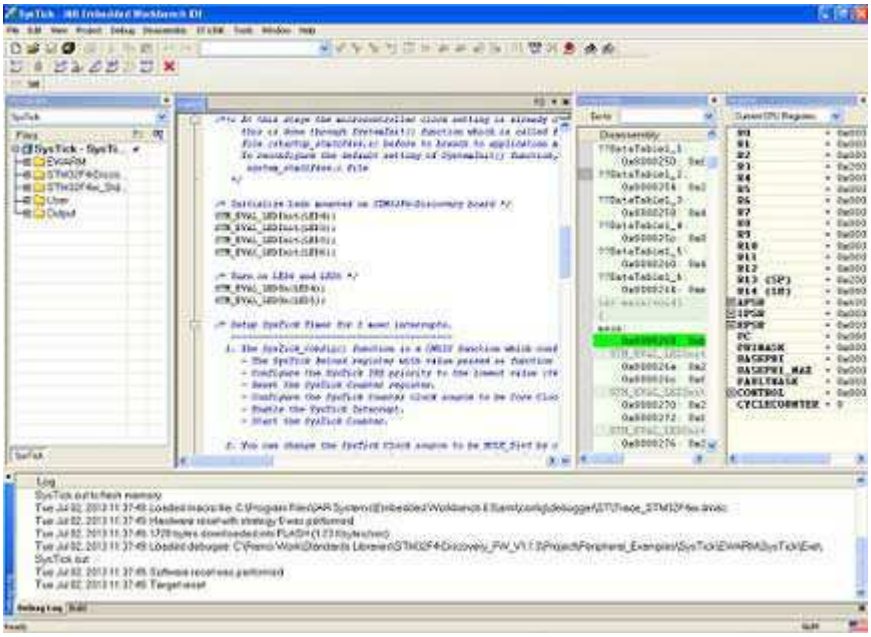
  Project>Options…C/C++ compiler>pre-processor>

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.

Figure 5. IAR Download and Debug button

6. The debugger in the IAR Embedded Workbench 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.

Figure 6. IAR Embedded Workbench debugger screen



To run your application, from the Debug menu, select Go. Alternatively, click the Go button in the toolbar to run your application.

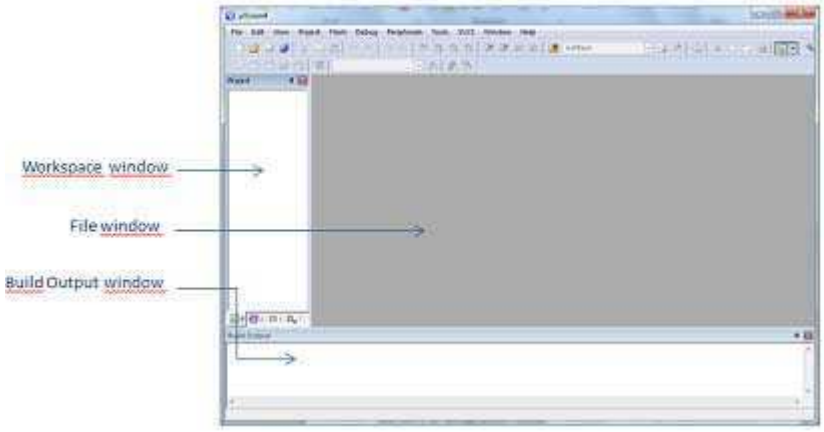
Figure 7. IAR Go button

2.2 MDK-ARM toolchain

1. Open the Keil MDK-ARM Microcontroller Kit.

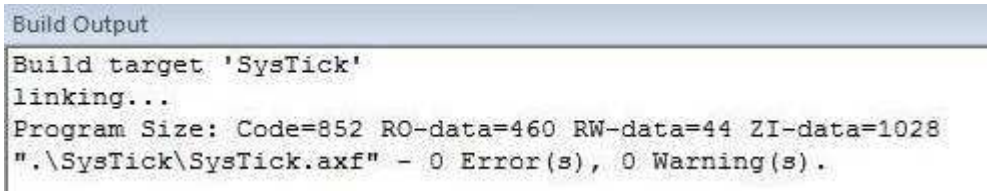
Figure 8 shows the basic names of the Keil uVision4 windows referred to in this document.

Figure 8. uVision4 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 your project.
4. If your project is successfully compiled, the window shown in *Figure 9* is displayed.

Figure 9. MDK-ARM project successfully compiled



If you need to change your 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

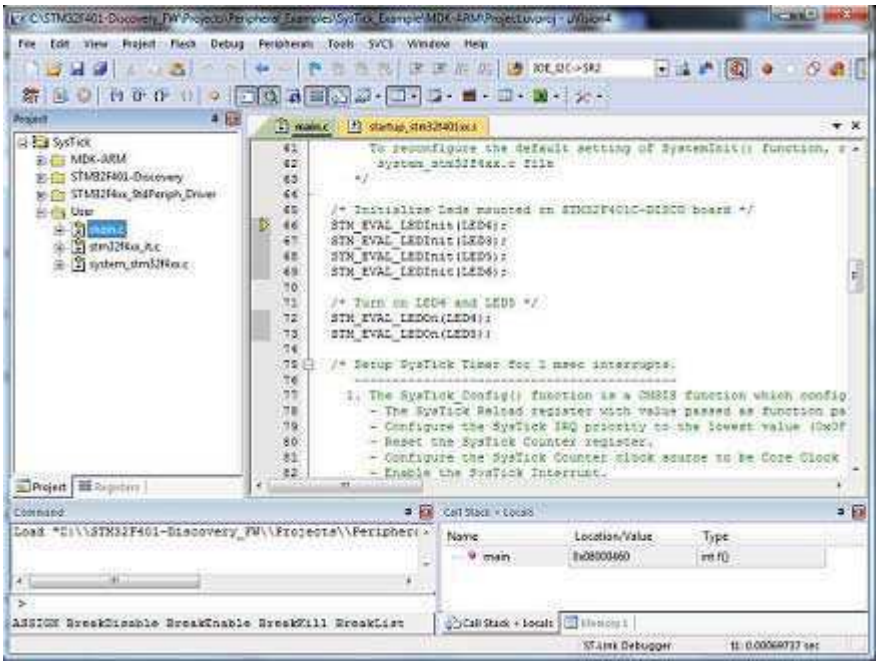
  Project>Options for Target > C/C++ > Preprocessor symbols > Define

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

Figure 10. MDK-ARM Start/Stop Debug Session button

6. The MDK-ARM debugger 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.

Figure 11. MDK-ARM debugger screen



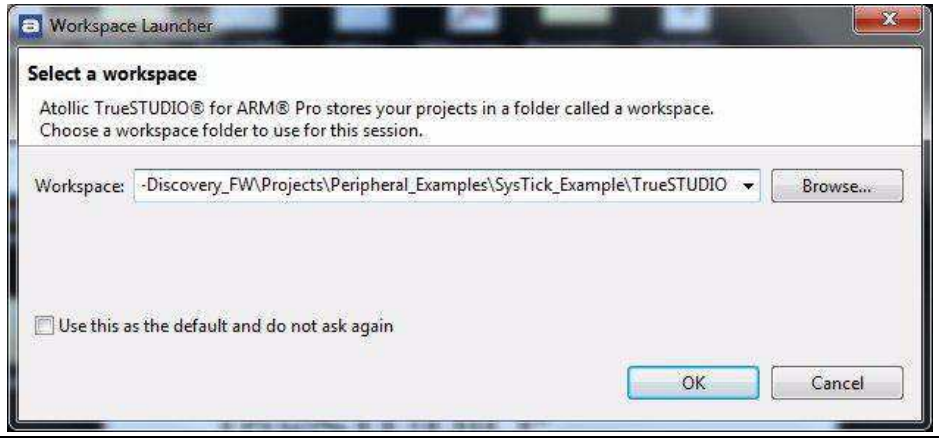
To run your application, from the Debug menu, select Run. Alternatively, click the Run button in the toolbar to run your application.

Figure 12. MDK-ARM Run button

2.3 TrueSTUDIO toolchain

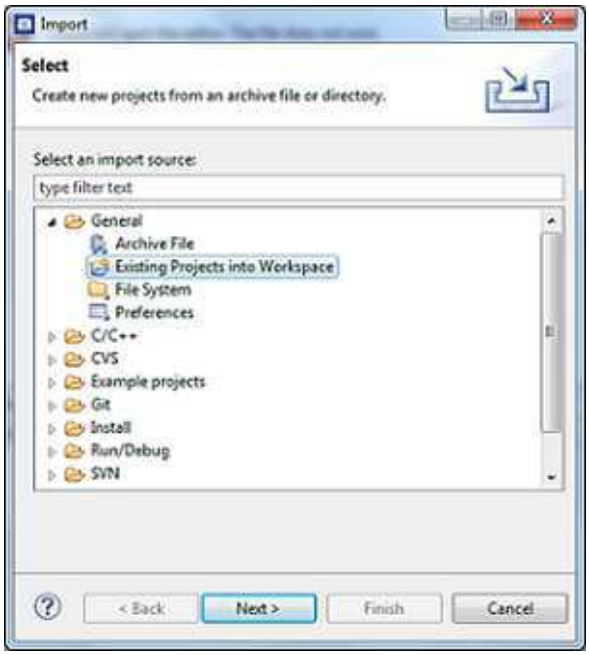
1. Open the Atollic TrueSTUDIO® for ARM product. The program launches and prompts for the Workspace location.

Figure 13. TrueSTUDIO workspace launcher dialog box



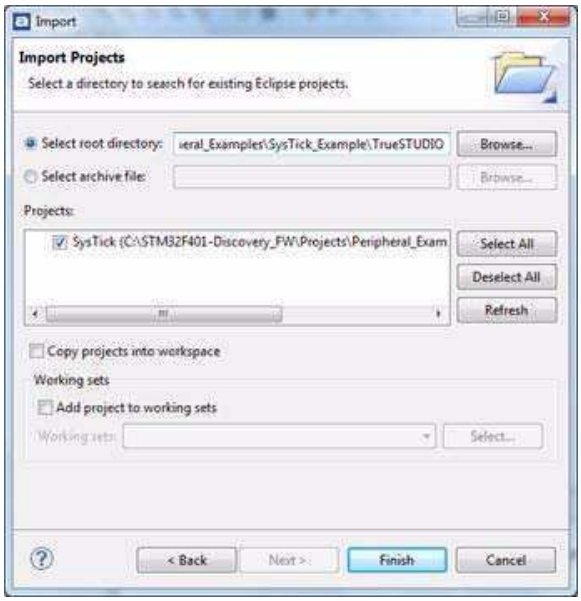
2. Browse to select a TrueSTUDIO workspace of either an example or demonstration or template workspace file, and click OK to load it.
3. To load an existing project in the selected workspace, select Import from the File menu to display the Import dialog box.
4. In the Import window, open General, select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

Figure 14. TrueSTUDIO import source select dialog box



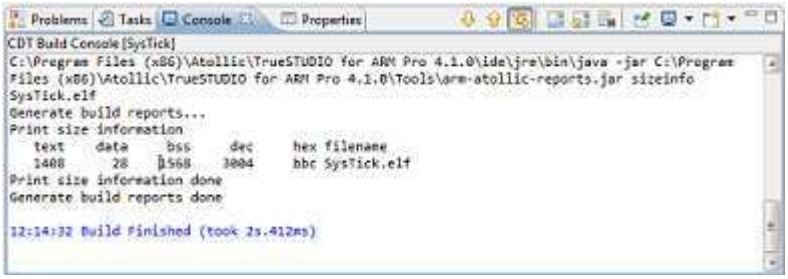
5. Click Select root directory; then browse to the TrueSTUDIO workspace folder.

Figure 15. TrueSTUDIO 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 your project is successfully compiled, the following messages display on the Console window.

Figure 16. TrueSTUDIO project successfully compiled



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

- For Include directories

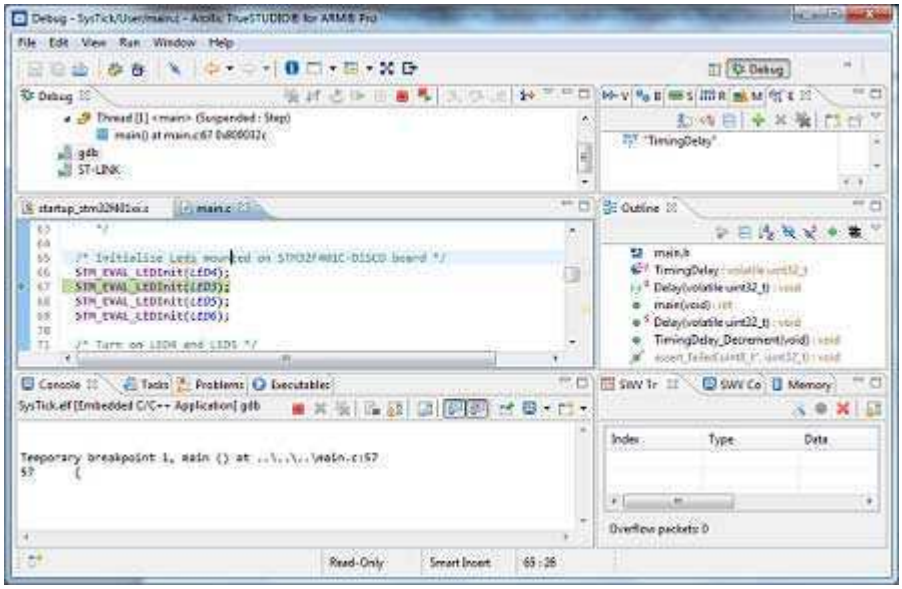
  C Compiler>Directories>Include path
- For pre-processor defines

  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. TrueSTUDIO debug window



The debugger in the Atollic TrueSTUDIO 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 your application, from the Run menu, select Resume or, alternatively, click the Resume button in the toolbar.

3 STM32F401 advanced debugging

The STM32 family using the Cortex-M4 processor includes many interrupts, and it can be difficult to determine when they are being activated and how often.

The Serial Wire Viewer (SWV) tool of on the STM32F401 family makes this task easy. SWV displays PC samples, exceptions (including interrupts), data reads and writes, ITM (printf), CPU counters and a timestamp. This information comes from the ARM CoreSight™ debug module integrated into the STM32F401 CPU.

SWV does not steal any CPU cycles and is non-intrusive (except for the ITM Debug printf Viewer).

You have already configured SWV on the template project. This allows:

1. Retargeting printf to the ITM stimulus port(0). This facilitates the display of the debug messages.

   How to use it:

   – EWARM: View > Terminal IO
   – MDK-ARM: View > Serial Windows Debug (printf) Viewer
   – TrueSTUDIO: View > SWV Console

2. Exception trace:
   - Provides information on the exception:
     – Entry: when the exception enters.
     – Exit: when the exception exits or returns.
     – Return: when all the exceptions have returned to the main XXX
   - How to use it:
     – EWARM: ST-LINK > Interrupt log
     – MDK-ARM: View > Trace > Exceptions
     – TrueSTUDIO: View > SWV Exception Trace Log

3. Function profiler: shows timing information for the functions in an application
   – EWARM: ST-LINK > Function Profiler
   – MDK-ARM: View > Analysis Window > Code Coverage
   – TrueSTUDIO: View > SWV Statistical Profiling

4. Data Trace Timeline: shows a graphical representation of the data
   – EWARM: ST-LINK > Timeline (Data log)
   – MDK-ARM: View > Analysis Window > Logic Analyzer
   – TrueSTUDIO: View > SWV Data Trace Timeline

4 Software toolchains helpful references and links

The following table provides useful references about the integrated development environments described in this document.

Table 2. Links to software toolchains

Toolchain	Download link
EWARM	www.iar.com/en/Products/IAR-Embedded-Workbench/ARM/ – EWARM_UserGuide
MDK-ARM	www.keil.com/demo/eval/arm.htm www.keil.com/arm/mdk.asp
TrueSTUDIO	www.atollic.com/index.php/request-eval-license

5 Revision history

Table 3. Document revision history

Date	Revision	Changes
02-Oct-2013	1	Initial release.

Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST’s terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST’S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

ST PRODUCTS ARE NOT DESIGNED OR AUTHORIZED FOR USE IN: (A) SAFETY CRITICAL APPLICATIONS SUCH AS LIFE SUPPORTING, ACTIVE IMPLANTED DEVICES OR SYSTEMS WITH PRODUCT FUNCTIONAL SAFETY REQUIREMENTS; (B) AERONAUTIC APPLICATIONS; (C) AUTOMOTIVE APPLICATIONS OR ENVIRONMENTS, AND/OR (D) AEROSPACE APPLICATIONS OR ENVIRONMENTS. WHERE ST PRODUCTS ARE NOT DESIGNED FOR SUCH USE, THE PURCHASER SHALL USE PRODUCTS AT PURCHASER’S SOLE RISK, EVEN IF ST HAS BEEN INFORMED IN WRITING OF SUCH USAGE, UNLESS A PRODUCT IS EXPRESSLY DESIGNATED BY ST AS BEING INTENDED FOR “AUTOMOTIVE, AUTOMOTIVE SAFETY OR MEDICAL” INDUSTRY DOMAINS ACCORDING TO ST PRODUCT DESIGN SPECIFICATIONS. PRODUCTS FORMALLY ESCC, QML OR JAN QUALIFIED ARE DEEMED SUITABLE FOR USE IN AEROSPACE BY THE CORRESPONDING GOVERNMENTAL AGENCY.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.
Information in this document supersedes and replaces all information previously supplied.
The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com