ATOLLIC TRUESTUDIO® FOR STM32 QUICK START GUIDE

This document is intended for those who want a brief, "bare bones" getting started guide. This should suffice for that purpose, but a lot of detail has been left out. For further information see the ***Atollic TrueSTUDIO User Guide***.

Please note that this manual applies to users of STM32 target devices only.

INFORMATION CENTER

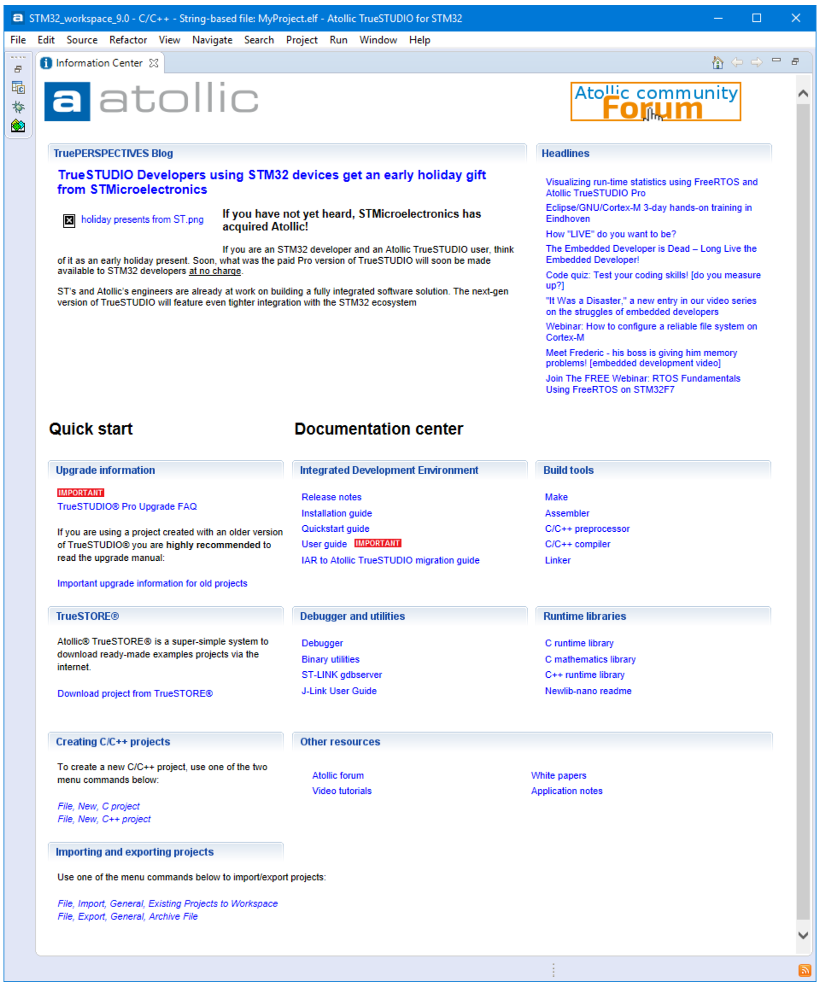
When ***Atollic TrueSTUDIO*** is started the first time the **Information Center** is opened.

The **Information Center** enables the user to quickly reach information regarding the product, and how to use it. By clicking on the corresponding hypertext links, manuals can be opened and latest information accessed at the Atollic web-site.

It is not required to read all material before using the product for the first time. Rather, it is recommended to consider the **Information Center** as a collection of reference information to return to.

The **Information Center** window may be reached at any time via the **Information Center** toolbar button or via the **Help, Information Center** menu command.

When the **Information Center** tab is closed the standard views in the **C/C++ perspective** is opened.


STM32_workspace_9.0 - C/C++ - String-based file: MyProject.elf - Atollic TrueSTUDIO for STM32
File Edit Source Refactor View Navigate Search Project Run Window Help
Information Center
atollic
Atollic community Forum

TruePERSPECTIVES Blog

TrueSTUDIO Developers using STM32 devices get an early holiday gift from STMicroelectronics

holiday presents from ST.png

If you have not yet heard, STMicroelectronics has acquired Atollic!

If you are an STM32 developer and an Atollic TrueSTUDIO user, think of it as an early holiday present. Soon, what was the paid Pro version of TrueSTUDIO will soon be made available to STM32 developers at no charge.

ST's and Atollic's engineers are already at work on building a fully integrated software solution. The next-gen version of TrueSTUDIO will feature even tighter integration with the STM32 ecosystem

Headlines

- Visualizing run-time statistics using FreeRTOS and Atollic TrueSTUDIO Pro
- Eclipse/GNU/Cortex-M 3-day hands-on training in Eindhoven
- How "LIVE" do you want to be?
- The Embedded Developer is Dead – Long Live the Embedded Developer!
- Code quiz: Test your coding skills! [do you measure up?]
- "It Was a Disaster," a new entry in our video series on the struggles of embedded developers
- Webinar: How to configure a reliable file system on Cortex-M
- Meet Frederic - his boss is giving him memory problems! [embedded development video]
- Join The FREE Webinar: RTOS Fundamentals Using FreeRTOS on STM32F7

Quick start

Upgrade information

IMPORTANT
TrueSTUDIO® Pro Upgrade FAQ

If you are using a project created with an older version of TrueSTUDIO® you are **highly recommended** to read the upgrade manual:

Important upgrade information for old projects

TrueSTORE®

Atollic® TrueSTORE® is a super-simple system to download ready-made examples projects via the internet.

Download project from TrueSTORE®

Creating C/C++ projects

To create a new C/C++ project, use one of the two menu commands below:

File, New, C project
File, New, C++ project

Importing and exporting projects

Use one of the menu commands below to import/export projects:

File, Import, General, Existing Projects to Workspace
File, Export, General, Archive File

Documentation center

Integrated Development Environment

- Release notes
- Installation guide
- Quickstart guide
- User guide IMPORTANT
- IAR to Atollic TrueSTUDIO migration guide

Build tools

- Make
- Assembler
- C/C++ preprocessor
- C/C++ compiler
- Linker

Debugger and utilities

- Debugger
- Binary utilities
- ST-LINK gdbserver
- J-Link User Guide

Runtime libraries

- C runtime library
- C mathematics library
- C++ runtime library
- Newlib-nano readme

Other resources

- Atollic forum
- Video tutorials
- White papers
- Application notes

Figure 1 – ***Atollic TrueSTUDIO*** with Information Center

Workspace and Projects

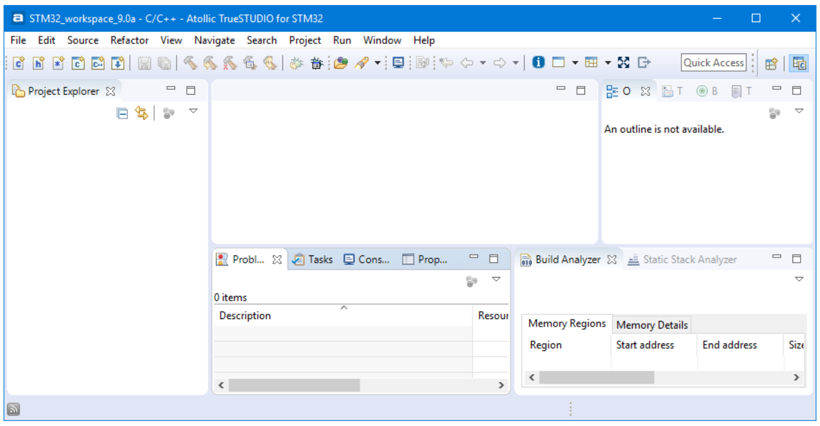

STM32_workspace_9.0a - C/C++ - Atollic TrueSTUDIO for STM32
File Edit Source Refactor View Navigate Search Project Run Window Help
Quick Access
Project Explorer
An outline is not available.
Probl... Tasks Cons... Prop...
0 items
Description
Resour
Build Analyzer
Static Stack Analyzer
Memory Regions
Memory Details
Region
Start address
End address
Size

Figure 2 – ***Atollic TrueSTUDIO*** with C/C++ perspective open, no projects defined yet

Atollic TrueSTUDIO is built using the ECLIPSE™ framework, and thus inherits some characteristics that may be unfamiliar to new users.

The ECLIPSE™ editing environment uses perspectives. A **perspective** is a set of windows dedicated to a purpose.

The **C/C++ perspective** is dedicated to writing and editing code and navigating through projects. You will spend a lot of time in this perspective.

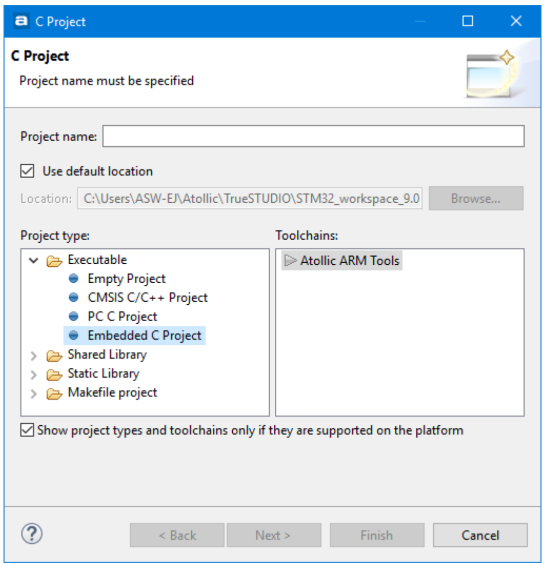
Another popular perspective is the **Debug perspective**, where you do your debugging and testing. Switching back and forth between perspectives is easy, and as we will see, automatic in some cases.

Workspace and Projects: The next thing to be done is to create a project. Before doing this we first introduce the idea of a **workspace**. A workspace is a container that includes project-folders or information about project-folders and a ***.metadata*** file, which contains information about the projects. A workspace is simply a folder on a hard drive, which can be located anywhere in the storage media. When ***Atollic TrueSTUDIO*** starts up; it asks which workspace is to be used. This may be changed at any time by selecting **File->Switch Workspace** and navigating to another folder.

Creating the First Project: The easiest way to create a new embedded project is to use the Atollic Project wizard. This can be started with the toolbar buttons or select the **File, New, C Project** menu command. Enter a project name and select **Embedded C Project**. This will bring up a series of dialogue boxes that will gather information on the kind of project you want to create, and then produce it.

Information gathered by Project Wizard includes:

1. The kind of project to be created; empty project, PC project or embedded project
2. The name of the project
3. Target information, e.g. whether a particular evaluation board is to be used, or just a particular microcontroller
4. The debug probe to be used
5. Whether or not I/O redirection is to be used, and printf() runtime library type
6. What build configurations (Debug, Release) are to be included


C Project

C Project

Project name must be specified

Project name:

☑ Use default location

Location: C:\Users\ASW-EJ\Atollic\TrueSTUDIO\STM32_workspace_9.0

Project type:
- Executable
  - Empty Project
  - CMSIS C/C++ Project
  - PC C Project
  - Embedded C Project
- Shared Library
- Static Library
- Makefile project

Toolchains:
- Atollic ARM Tools

☑ Show project types and toolchains only if they are supported on the platform

Figure 3 – Embedded C project wizard

PROJECT INFORMATION

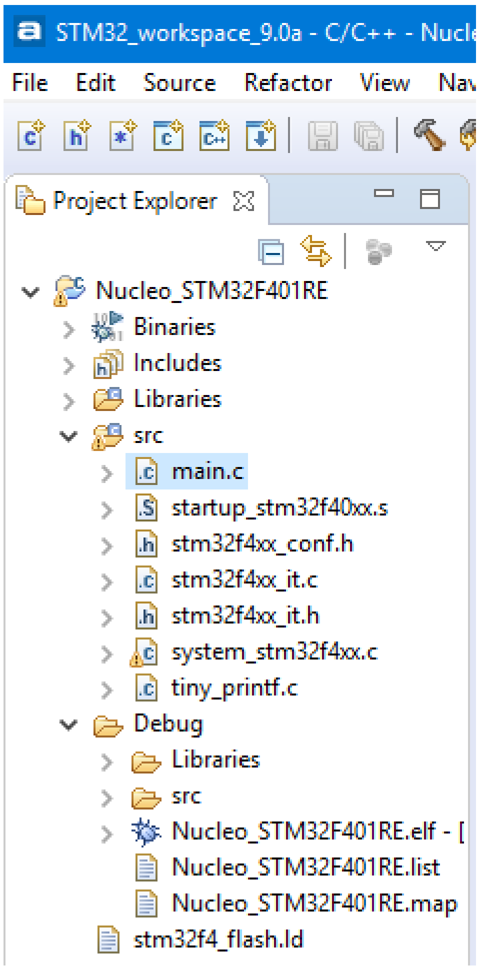


Figure 4 –Project Explorer window populated with project

Once the information is entered into the wizard, things happen: the **Project Explorer** docking view is populated with the chosen project, and the project is built. In the **Project Explorer** window, the subfolders can be expanded to any desired sublevel. What kind of project that Project Wizard creates depends on the family of chip. Most chipmakers have created a full library of drivers written in C for the peripherals of their parts. If available, the **Project Wizard** pulls these into the project for future use. In any case, the project contains all necessary elements of a complete embedded application, ready to program into the memory of the part and run. This normally includes:

- Startup code to initialize variables, set up stack and heap and place exception handler table in default memory location
- Linker script that reflects the resources of the chosen part
- Code to initialize core functions and system clocks
- A main() function that toggles at least one I/O pin
- CMSIS functions and header files
- If an evaluation board was selected, some board initialization functions
- Optionally, depending on selection in Project Wizard, a `syscalls.c` module for I/O redirection
- Function stubs for exception handlers
- Some examples, but not all, have support for SysTick timer
- Programming utility for programming code into Flash or RAM as specified in the Project Wizard

When many projects exists in a workspace it is possible to close some projects. Actions controlled by the icons such as run, launch debug session etc. will then apply to the open projects.

Projects can be Closed / Opened by right-clicking on the project name in the **Project Explorer** docking view. Then select **Open Project** or **Close Project.**

Controls: All control icons have tool tips activated by hovering the mouse over the icon. The icons are shortcuts for the same functions that can be activated from the main menu bar. The most important icons are shown in Figure 4. These icons control specific functions that pertain to code editing, building and project management, so they are unique to the C/C++ perspective.

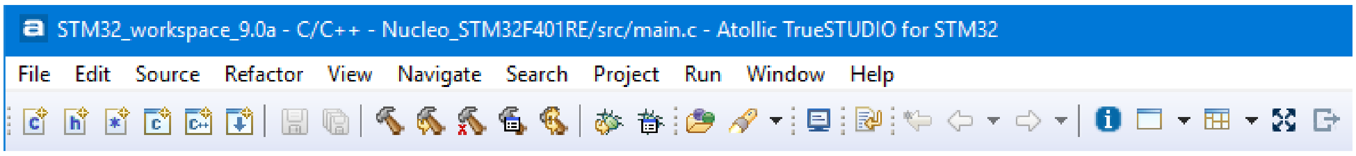

STM32_workspace_9.0a - C/C++ - Nucleo_STM32F401RE/src/main.c - Atollic TrueSTUDIO for STM32
File
Edit
Source
Refactor
View
Navigate
Search
Project
Run
Window
Help

Figure 5 - Main control icons for editing, building and project management

The main control icons for editing, building and project management:

Use these icons to create new C source code module, header files or new object such as project, library or repository [**File->New** from Main Menu options]

Use these icons to create new C or C++ projects. [**File->New** from Main Menu options]

This icon opens ***Atollic TrueSTORE***. With a single mouse-click projects can be downloaded and installed into to the current workspace. Internet access is required. [**File->New** from Main Menu options]

Use these icons (from left to right) to do incremental builds, complete rebuild, clean (delete object and executables), access the build settings menu, and manage build configurations [Functions can be activated in **Project** option in Main Menu]

Use these icons (from left to right) to launch a particular debug configuration, or configure a debug configuration [Functions can be activated in **Run** option in Main Menu]

The flashlight icon launches various search utilities, the arrows navigate among recently visited places in the project [**Search** and **Navigate** Main Menu options]

This icon bring up the information window where documentation, examples and other help resources are accessed [**Help->Welcome** from Main Menu options]

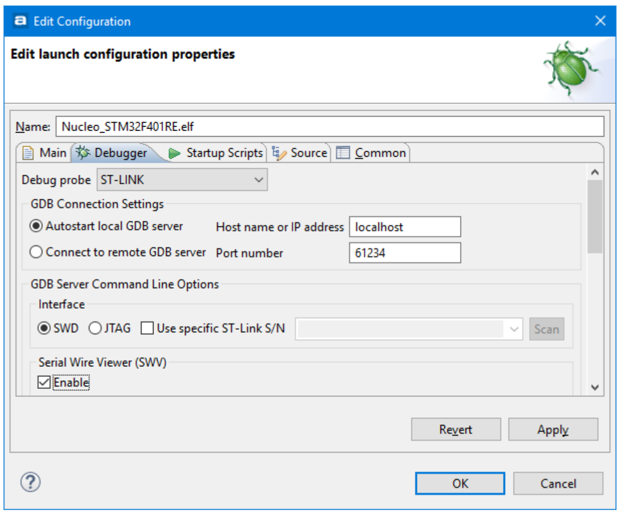
Click on the down arrow to show a list of perspectives that can be opened [**Window->Open Perspective** from Main Menu options]

Click on the down arrow to show a list of views that are relevant to this perspective that can be opened [**Window->Show View** from Main Menu options]

This icon toggles to full screen mode (ESC to exit)

This icon is a shortcut back to the C/C++ perspective (disabled in C/C++ perspective)

DEBUGGING


Edit Configuration

Edit launch configuration properties

Name: Nucleo_STM32F401RE.elf

Main | Debugger | Startup Scripts | Source | Common

Debug probe: ST-LINK

GDB Connection Settings
- (●) Autostart local GDB server — Host name or IP address: localhost
- ( ) Connect to remote GDB server — Port number: 61234

GDB Server Command Line Options
- Interface: (●) SWD ( ) JTAG [ ] Use specific ST-Link S/N — Scan

Serial Wire Viewer (SWV)
- [✓] Enable

Revert | Apply

OK | Cancel

Figure 6 – Debug Configuration Properties Dialogue box

Once you have built a project without errors, use the icon to launch the program code into the part and launch the debugger. The first time a debug session is launched, ***Atollic TrueSTUDIO*** shows the debug launch configuration menu. This gives the user the opportunity to verify settings and make changes if desired. Some of the settings are derived from Project Wizard choices. For example, the choice of the debug probe, e.g. Segger J-Link or ST-Link is made in the Project Wizard. Further, based on information in the chip database, the choice between Serial Wire Debug (SWD) and JTAG is made, as some parts do not have enough pins for the full JTAG implementation.

Important Note: If the Serial Wire Viewer (SWV) is to be used during debugging, this must be explicitly enabled in the dialogue box. Click on the **Debugger** tab to reach the SWV settings and enable SWV. This is because SWV is not enabled by default. If changes are made to the default settings in the launch configuration properties dialogue box, they must be saved by clicking on the “Apply” button. When ready to launch the debug session, click on OK. Several things happen “under the hood” including the launch of the debugger driver, and in the case of J-Link, the gdbserver, as well as the programming of the application into the part, and any other actions called for in the startup script. At this point, the application is normally halted at the first line of main().

The main control icons of the debugger are as follows:

Resume full speed execution of the application on the target (greyed out when running)

Halt execution (greyed out when halted)

Step into a function, over a function, out of a function

Toggle between C and assembly language stepping

Basic Debugging Techniques:

- To set a breakpoint, click on the blue horizontal bar next to the statement line number.
- To view memory locations, you may use the Variables window (variables in focus or global variables), the Memory window or the Expressions window.
- To terminate the debug session, click on the red square . This automatically brings you back to the C/C++ perspective for further editing of the code. Edits may be made and saved in the Debug perspective. To rebuild and re-launch the debug session, left click on the project name in the Debug window, then right click and select **Terminate and Relaunch**, or click the icon.

Configuring and using your debugging tools, especially with SWV is frequently more complicated than using the IDE for C/C++ code development. The User Manual has a great deal more detail on this topic, and should be consulted for anything beyond the most basic operations.

COPYRIGHT

TRADEMARKS

DISCLAIMER

DOCUMENT IDENTIFICATION

TS-QSG	October 2012

REVISION HISTORY

2nd	December 2017 – Updated for Atollic TrueSTUDIO for STM32 v9.0.0

STMicroelectronics Software AB
Science Park
Gjuterigatan 7
SE- 553 18 Jönköping
Sweden

Email: sales@atollic.com
Web: www.atollic.com

STMicroelectronics

Web: www.st.com