TN0072
Technical note

Software toolchains and STM32 features

Introduction

The documentation provides an overview of the STM32 devices and various toolchains. It provides information on the STM32 characteristics and how they are supported. Many features of the STM32 devices such as the STM32 peripheral description, macro and script files, and the STM32 option bytes are described in comparative tables.

This description concerns all the STM32 families and their derivatives.

STM32 overview

The STM32 family of 32-bit Flash microcontrollers incorporates the high-performance ARM® Cortex™-M3 32-bit RISC core, high-speed embedded memories (Flash memory up to 512 Kbytes and RAM up to 64 Kbytes), and an extensive range of enhanced I/Os and peripherals.

The STM32 family benefits from the Cortex-M3 architectural enhancements including the Thumb-2 instruction set to deliver improved performance with better code density, significantly faster response to interrupts, all combined with industry leading minimal power consumption.IAR™, Hitex, Raisonance and Keil™ toolchains

IAR, Hitex, Raisonance and Keil toolchains

The following table provides a general information of the considered toolchains.

Table 1.   IAR™, Hitex, Raisonance and Keil™ toolchains

Toolchain	Company	Version	Release date	Hardware emulator	Used Compiler
EWARM (IAR™ embedded workbench® for ARM)	IAR Systems®	5.20	July 2008	J-Link	IAR C/C++
HiTOP	Hitex Development Tools	5.30.0014	October 2008	Tantino	TASKING
Ride (Raisonance integrated development environment)	Raisonance	7.14.0001	October2008	RLink	ARM-GCC
RVMDK (RealView® microcontroller development kit)	Keil™, an ARM® Company	3.24	October 2008	ULINK	ARMCC

Contents

1 STM32 peripheral description

Every toolchain provides specific windows and files that contain information about the device and allow to display and/or to modify the STM32 peripheral registers.

Table 2.   STM32 peripheral description

Toolchain	Command to display register window	File path	Possibility to add new peripheral description
EWARM	**View > Register** command from the toolbar.	*[install_directory]\IAR Systems\ Embedded Workbench 5.0\ ARM\config\Debugger\ST*	Yes
HiTOP	**View > SFR Window** command from the toolbar.	*[install_directory]\HiTOP53- CTX\Tantino-Cortex\derivatives\ ST Microelectronics*	Yes
Ride	**View >View Debug Explorer > peripheral** command from the toolbar.	*[install_directory]\Raisonance\Ride\ Sim\ARM*	Yes
RVMDK	**Peripheral** command from the toolbar menu.	*Files will be available in next releases*	No

2 Macro and script files

The script files are used to perform any settings/initialization which must be done before or after loading the application.

Table 3.   Macro and script files

Toolchain	Availability
EWARM	Available and executed before/after debug.
HiTOP	Available and executed before/after debug. The script file may be assigned to HiTOP toolbar button.
Ride	Not available.
RVMDK	Available and executed before/after debug.

3   STM32 option bytes

The option bytes are programmed differently than normal user addresses.

The STM32 has the following option bytes:

●   4 for write protection,
●   1 for read-out protection,
●   1 for configuration,
●   2 for user data storage.

They are programmed by the end user depending on the application requirements.

Table 4.   STM32 option bytes

Toolchain	Availability	Command to display the option bytes
EWARM	Not available. Can be programmed via a script.	
HiTOP	All option bytes are available.	**View > SFR Window > FLASH**
Ride	The following option bytes are available: – write protection, – read-out protection, – configuration.	**Project > Properties > RLink Configuration > Advanced Options**
RVMDK	All option bytes are available.	Add the file which describes the option bytes (provided by Keil) to the project.

4 STM32 Flash operations

In addition to the Flash programming, toolchains offer many other Flash operations.

4.1 Flash erase

The Flash memory erase operation can be performed either at sector level or on the whole Flash area (full chip erase). The erase time varies from one toolchain to another and it depends on many factors, such as the JTAG frequency.

Table 5. Flash erase

Toolchain	Full chip erase	Sectors erase	Command to display the erase menu
EWARM	Not available[1]	Not available[1]	
HiTOP	Available via menu	Available via menu	**Project settings > STM32flash > erase range**
Ride[2]			**Project > properties > RLink configuration > Advanced options > erase target now**
RVMDK	Available via menu	Available via menu	**Flash > erase**

1. Erase is performed implicitly at each flash downlaod.

2. Ride provides also a command line application "cortex_pgm.exe" and a window interface "Rflasher" to insure Flash operations such as program, erase, verify, mass product and blank-check.

4.2 STM32 Flash algorithm source files

An algorithm source file defines and adds Flash programming algorithms. It contains the function code for the necessary operations such as `Erase Chip`, `Erase Block` and `Program Block` and defines the device parameters.

Table 6. STM32 Flash algorithm source files

Toolchain	Availability	Path
EWARM	Available	*[install_directory]\IAR Systems\Embedded Workbench 5.20\ARM\src\flashloader\ST*
HiTOP	Not available [1]	*[install_directory]\Hitex\HiTOP53-CTX\Tantino-Cortex\Flash*
Ride	Not available [1]	*[install_directory]\Raisonance\Ride\Sim\ARM*
RVMDK	Available	*[install_directory]\Keil\ARM\flash*

1. Only some device parameters such as flash size, sectors number… are accessible. the flash algorithm file can be prvided on demand.

4.3   Flash breakpoints

A breakpoint identifies a location in the program area of the target system. When it is reached, the program stops running and the emulation is halted.

The STM32 has improved debugging and EWARM, HiTOP, Ride and RVMDK all allow up to six hardware flash breakpoints. The breakpoint can be set when the run mode is enabled or disabled.

5 STM32 serial wire view and STM32 serial wire debug

The serial wire view (SWV) in combination with serial wire debug (SWD) provide advanced debugging features like execution trace, as well as the same execution control and debugging features that are possible using JTAG.

The SWV feature is only supported when debugging in SWD mode.

Table 7.   STM32 serial wire viewer availability

Toolchain	Availability
EWARM	Available since J-Link 6.0
HiTOP	Available since Tantino ARM
Ride	only SWD is available
RVMDK	Available since ULINK 2

6 Debug support for timers, watchdog, CAN and I^{2}C

- Timers

  During a breakpoint, it is necessary to choose how timer counters and watchdog behave:
  - Timer counters can continue counting inside the breakpoint. This is usually required when a motor is controlled by a PWM output
  - Timer counters can stop counting inside a breakpoint. This is required for watchdog purposes.
- CAN peripheral

  The user can choose to block the update of the receive register (RDR) during a breakpoint.
- I^{2}C interface

  For the I^{2}C, the user can choose to block the SMBUS timeout during a breakpoint.

Table 8.   Debug support for timers, watchdog, CAN and I^{2}C

Toolchains	Timers debug support	Watchdog debug support	CAN debug support	I^{2}C debug support
EWARM	Available through a macro provided on demand	Available through a macro provided on demand	Available through a macro provided on demand	Available through a macro provided on demand
HiTOP	Available Project settings > processor settings > Debug MCU configuration	Available Project settings > processor settings > Debug MCU configuration	Available Project settings > processor settings > Debug MCU configuration	Not available. Can be supported by macro.

Table 8.   Debug support for timers, watchdog, CAN and I^{2}C (continued)

Toolchains	Timers debug support	Watchdog debug support	CAN debug support	I^{2}C debug support
Ride	Not available	Not available	Not available	Not available
RVMDK	Available through the *STM32DBG.ini* macro located in *[install_directory]\ARM\ Boards\Keil\MCBSTM32 \STLib_Blinky*	Available through the *STM32DBG.ini* macro lacated in *[install_directory]\ARM\ Boards\Keil\MCBSTM32 \STLib_Blinky*	Available through the *STM32DBG.ini* macro located in *[install_directory]\ARM\ Boards\Keil\MCBSTM32 \STLib_Blinky*	Not available. Can be added by user in the *STM32DBG.ini* macro

7   Connections

In order to satisfy the STM32 features (such as the capability of the debugger host to connect under reset), several options are provided when establishing the connection between the target and the toolchains.

Table 9.   Connection options

Toolchain	Connect + system reset [1]	Connect after delay[2]	Connect at different JTAG frequencies	Connect under reset [3]
EWARM	Available	Available	Available	Available
HiTOP	Available	Not available	Available	Not available
Ride	Available	Not available	Available	Available
RVMDK	Available	Not available	Available	Available

1. Connect + system reset: connect to the target with resetting the core and setting all the registers to their reset values

2. Connect after delay: the delay is between the end of the software reset and the halt of CPU. This feature is used, for example,to run an initialization code before halting the CPU.

3. Connect under reset: it is possible for the debugger to connect under System Reset, programming the Core Debug Registers to halt the core when fetching the reset vector. Then the host can release the system reset and the core will halt immediately without having executed any instructions. In addition, it is possible to program any debug features under System Reset.

8 Output formats

In addition to the default output format (.out for EWARM, .abs for HiTOP, .elf for Ride and .axf for RVMDK), other (optional) output formats are supported:

● **HEX**: A .hex file (Intel HEX) is a text file, with each line containing hexadecimal values encoding a sequence of data and its starting offset or absolute address.
There are three types of Intel HEX: 8-bit, 16-bit, and 32-bit.
They are distinguished by their byte order.

● **S19:** An ASCII encoding format for binary data developed by Motorola.
It consists of a series of ASCII records. Every record begins with the letter "S," followed by a "1" if the record contains data or a "9" if it is the last record in the file.

● **ELF**: (executable and linkable file) is a standard for object modules, libraries, executables, and core files. Each ELF file is made up of one ELF header, followed by file data.
The data file is composed from segment and sections. The segments contain information that is necessary for runtime execution of the file, while sections contain important data for linking and relocation.

Table 10.   Output formats

Toolchain	HEX	S19	ELF
EWARM	Yes	Yes	Yes
HiTOP	Yes	Yes	Yes
Ride	Yes	No	Yes
RVMDK	Yes	No	Yes

9 RTOS plugins

This option allows the use of a real-time operating system (RTOS) in the application. The debugger RTOS awarness plugin modules give the user a high level of control and visibility over an application built on top of a real time operating sysstem.

It displays RTOS-specific items like task lists, queues, semaphores, mailboxes and various RTOS system variables

The following table describes how and which RTOS is supported by every toolchain.

Table 11.   RTOS plugins

Toolchain	Availability	Available RTOSs
EWARM	Available	RTXC, IAR PowerPac, uC/OS-II, Segger embOS, CMX...
HiTOP	Available [1]	AMX, Nucleus Plus, RCX, PxROS, ThreadX...
Ride	Not available	
RVMDK	Available	RTX, RTX166 Tiny ThreadX…

1. An additional license (which can be purchased from Hitex) is needed.

10   Revision history

Table 12.   Document revision history

Date	Revision	Changes
18-Nov-2008	1	Initial release.
05-Dec-2008	2	Added *Section 6: Debug support for timers, watchdog, CAN and I^{2}C*.

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.

UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED ST REPRESENTATIVE, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER'S OWN RISK.

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 - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com