

---

**STM32 embedded target for MATLAB<sup>®</sup> and Simulink<sup>®</sup> release 5.1.0**

---

## Introduction

This release note is updated periodically to keep abreast of the evolution, problems and limitations of the STM32 embedded target for MATLAB<sup>®</sup> and Simulink<sup>®</sup> toolbox (STM32-MAT/TARGET). Check STMicroelectronics support website at [www.st.com](http://www.st.com) for its latest version. Refer to [Table 1](#) for the latest release summary.

**Table 1. STM32-MAT/TARGET 5.1.0 release summary**

Type	Summary
Major release	<ul style="list-style-type: none"><li>– Updated all peripherals as per versions STM32CubeMX 5.1.0 and MATLAB<sup>®</sup> R2018b</li><li>– Solved issues detected in version 4.4.2</li></ul>

## Customer support

For more information or help concerning STM32-MAT/TARGET, contact the nearest STMicroelectronics sales office. For a complete list of STMicroelectronics offices and distributors, refer to the [www.st.com](http://www.st.com) webpage.

## Software updates

Software updates and all the latest documentation can be downloaded from STMicroelectronics [www.st.com/en/development-tools/stm32-mat-target.html](http://www.st.com/en/development-tools/stm32-mat-target.html) webpage.

# Contents

- 1      General information ..... 3**
  - 1.1    Overview ..... 3
  - 1.2    Host PC system requirements ..... 3
  - 1.3    License ..... 3
  
- 2      Release 5.1.0 information ..... 4**
  - 2.1    New features ..... 4
  - 2.2    Fixed issues ..... 4
  - 2.3    Enhancements ..... 5
  - 2.4    Limitations ..... 5
  
- 3      Information on previous releases ..... 6**
  - 3.1    Release 4.4.2 information ..... 6
  - 3.2    Release 4.4.1 information ..... 7
  - 3.3    Release 4.4 information ..... 7
  - 3.4    Release 4.3 information ..... 7
  - 3.5    Release 4.2 information ..... 8
  - 3.6    Release 4.1 information ..... 10
  - 3.7    Release 4.0 information ..... 10
  - 3.8    Release 3.1 information ..... 11
  - 3.9    Release 3.0 information ..... 12
  
- 4      Revision history ..... 13**

# 1 General information

STM32-MAT/TARGET supports STM32 microcontrollers based on the Arm<sup>®(a)</sup> Cortex<sup>®</sup>-M processor.



## 1.1 Overview

The STM32-MAT/TARGET software development tool is used to deploy quickly application models in MATLAB<sup>®</sup> and Simulink<sup>®</sup> to STM32 MCUs.

The STM32-MAT/TARGET verifies and profiles STM32 execution results versus Simulink<sup>®</sup> simulation behavior, using processor-in-the-loop (PIL) testing.

The STM32-MAT/TARGET provides a Simulink<sup>®</sup> blockset library for STM32 peripherals and automatically generates C code based on HAL libraries.

The STM32-MAT/TARGET allows the monitoring of applications running on STM32, using *External Mode*. The user controls the parameters of the application running on the STM32 and visualizes the results from Simulink<sup>®</sup>.

## 1.2 Host PC system requirements

- Windows<sup>®</sup> 7 64-bit operating system
- MATLAB<sup>®</sup>, Simulink<sup>®</sup> and Embedded Coder<sup>®</sup> R2018b version or more
- STM32CubeMX V5.1.0

## 1.3 License

STM32-MAT/TARGET is delivered under the *Mix Ultimate Liberty+OSS+3rd-party V1* software license agreement (SLA0048).

---

a. Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

## 2 Release 5.1.0 information

### 2.1 New features

- Updated all peripherals according to STM32CubeMX and MATLAB® versions:
  - STM32CubeMX V5.1.0
  - MATLAB® R2018b

### 2.2 Fixed issues

- General:
  - `getCubeVer()`: some improvements to get the correct version of STM32CubeMX
  - Updated documents
  - Changed the file organization of the demonstration examples to enable straight forward build and run
- ADC:
  - Updated Simulink® models ADC Regular conversion in the DMA and IT modes
  - Added examples for the 32F429IDISCOVERY Discovery board
- CAN:
  - Updated CAN blocks and demonstration
  - Support of the new CAN API introduced by STM32CubeMX 5.1.0
- USART:
  - Added examples for the 32F429IDISCOVERY Discovery board
- GPIO:
  - Added examples for the 32F429IDISCOVERY Discovery board
- TIMER
  - Output can be ARR or duty cycle for input capture mode
  - `_ETR` channel now is managed
  - Start Timer even when no channel is selected
  - Input capture and PWM input processed differently
- HRTIMER:
  - Updated *t/c* files for compilation/link procedure to be compatible with STM32CubeF3 firmware version 1.10.0
- PIL mode:
  - Proper initialization of `toolChainName` and `toolChainVersion`
  - Launcher class: set default for `toolChainName` and `toolChainVersion`
  - *File Version* for *ioc* set to 6 in accordance with STM32CubeMX V5.1.0
  - Moved ADC demonstration STM32F429I-DISC1 (STM32F429ZITx, USART1)

- External mode
  - Updated with STM32CubeMX V5.1.0
  - *no ioc* use case functional
  - Added example with STM32F302R8
  - STM32F302R8 @ 64 MHz, USART2: OK
  - STM32F401RE @ 84 MHz, USART2: OK
  - STM32F446ZE @ 180 MHz, USART3: OK

## 2.3 Enhancements

- None

## 2.4 Limitations

- DAC with DMA only in continuous mode.
- Issue with STM32CubeMX project generator for IDEs using ECLIPSE™, such as TrueSTUDIO®.  
The workaround is to remove the “*exclude from resource build*” that is set by default on generated project.
- *Warning: Code Generation* popup occurs sometimes after end-user presses *BuildModel* in Simulink® with 32F429IDISCOVERY when STM32CubeMX is opened for configuration and project code generation.  
The workaround is to activate "Yes" on Pop up command to generate the code.

## 3 Information on previous releases

### 3.1 Release 4.4.2 information

- I2S driver

#### Fixed issues

- General:
  - Defined ARM\_MATH\_CMx (x=0..7) automatically created depending on STM32
  - Get STM32CubeMx version before starting code generation
  - Remove warning when default selection of ADC/DAC is meaningless
  - Input type is void\* for GetBuffPtr function
- ADC:
  - DMA buffer acquisition value cannot be less than or equal to NULL, NaN or empty
- CAN:
  - CAN\_MESSAGE type available for input and output
  - Block declared asynchronous to manage interrupts
- GPIO:
  - Connected input is not only equal to 0/1 but it can also be computed
- SPI
  - Add one data output to the Simulink block model
  - 8-bit default data size
  - Full Duplex possibility when configured as SPI\_DIRECTION\_2LINES
- TIMER
  - Output can be ARR or duty cycle for input capture mode
  - \_ETR channel now is managed
  - Start Timer even when no channel is selected
  - Input capture and PWM input processed differently
- USART
  - Write USART speed "WordLenth\_x8" in generated IOC file for External Mode when STM32CubeMX version is > 4.18

#### Enhancements

- I2S capabilities

#### Limitations

- IOC file and Simulink models using this IOC file must be on the same disk volume. For example, it is not possible to have IOC file in volume c: and Simulink model in volume d:

### 3.2 Release 4.4.1 information

- Simulink model setting gives possibility to add external .c/.h files to project
- ADC available with DMA acquisition
- DAC available with DMA generation

#### Fixed issues

- CAN: generate code with Tx pointer instead of Rx pointer for sending message communication
- Timer: generated code update frequency and/or duty cycle only if value has changed
- autoreloadTimer variable becomes volatile

#### Enhancements

- Better performance for External Mode serial communication driver

#### Limitations

- IOC file and Simulink models using this IOC file must be on same volume (disk). For example, it is not possible to have IOC file in volume c: and Simulink model in volume d:

### 3.3 Release 4.4 information

- External Mode monitoring

#### Fixed issues

- Add ADC end-of-conversion interrupt for both regular and injected channels (EOS and JEOS)
- IOC used file name is propagated to the included Reference Models
- For HRTIMER blocks, replace Update TLC function with Output TLC function to better organize block function order call

#### Enhancements

- HRTIMER blocks are "Asynchronous" when there is only interrupt output

#### Limitations

- IOC file and Simulink models using this IOC file must be on the same volume (disk). For example, it is not possible to have IOC file in volume c: and Simulink model in volume d:

### 3.4 Release 4.3 information

- GPIO write with several pins at the same time

### Fixed issues

- Bug regarding code generation for peripherals initialization fixed in MATLAB release R2015b
- Default compiler defined in stm32.tmf file (COMPILER\_TOOL\_CHAIN = default)
- Possibility to use USART to receive multi instances with different number of characters to receive
- ADC parameters management changed for MATLAB R2015b

### Enhancements

- STM32CubeMx IOC configuration file path is now related to the model that uses it.
- Possibility to manage all the Simulink STM32 peripheral model interrupts independently of outputs. When both outputs and interrupts are used in the same model, interrupt Simulink function sample time is "Discrete". When interrupts, only, are set, interrupt Simulink function sample time is "Asynchronous".

### Limitations

- Project generation:
  - Some project variable definition (Define) are not integrated to project settings and must be manually added (cf Readme.html file paragraph "Known problems and solutions").
  - "Flash loader" is not selected for Keil® V5 generated project and must be manually added. (cf Readme.html file paragraph "Known problems and solutions").
  - Code generation for peripherals initialization for MATLAB release R2014b and R2015a is not working. The message displayed by MathWorks is:  
" The observed behavior is unfortunately a bug in our product that we will probably fix in R2015b.  
For releases R2014b and R2015a refer to the following bug-report page:  
<http://www.mathworks.com/support/bugreports/1181359>  
There you will find a patch to workaround the problem.  
Remark: Please login with your MathWorks account to see the page mentioned above and download the files".  
The number of this patch is 1181359.
  - When "Flash loader" is selected for Keil® V5 generated project, the "Flash Download" window must be opened to take in account the "Flash loader" just selected.

## 3.5 Release 4.2 information

- CAN peripheral driver:
  - C code generation for CAN peripheral based on HAL libraries and depending on STM32CubeMX configuration.
  - No application example. Test example for code generation and project build validation provided.



**Fixed issues**

- Message queue processing for USART Send pooling mode.
- Message buffer size and message buffer definition for I<sup>2</sup>C Read/Write multiple instances.
- Solve SPI known problem regarding warning message when blockset is dragged and dropped into application model.
- PIL bug fix: some USART tag name has been changed for ioc configuration file creation:
  - Parity-Asynchronous replaced with Parity
  - StopBits-Asynchronous replaced with StopBits
  - WordLength-Asynchronous replaced with WordLength

**Enhancements**

- HRTimer and SPI peripheral drivers added.

**Limitations**

- Project generation:
  - Some project variable definition (Define) are not integrated to project settings and must be manually added. (cf Readme.html file paragraph "Known problems and solutions").
  - "Flash loader" is not selected for Keil<sup>®</sup> V5 generated project and must be manually added. (cf Readme.html file paragraph "Known problems and solutions").
  - Code generation for peripherals initialization for MATLAB release R2014b and R2015a is not working. The message displayed by MathWorks is:  
" The observed behavior is unfortunately a bug in our product that we will probably fix in R2015b.  
For releases R2014b and R2015a refer to the following bug reports page:  
<http://www.mathworks.com/support/bugreports/1181359>  
There you will find a patch to workaround the problem.  
Remark: Please login with your MathWorks account to see the page mentioned above and download the files".  
The number of this patch is 1181359.

### 3.6 Release 4.1 information

The following features have been added:

- HRTimer peripheral driver:
  - C code generation for HRTIMERS peripheral based on HAL libraries and depending on STM32CubeMX configuration
  - BuckBoost and Multiphase documented examples
- REGISTERS blockset:
  - This blockset is used to directly generate a C code sentence with comments
  - It is possible to generate a C code sentence for Read/Write register access to every peripheral configured from STM32CubeMX
  - This blockset does not check text sentence and it is mandatory to enter exact register name text sentence in order to avoid compiler error
  - Used in HRTimer Multiphase example
- SPI peripheral driver:
  - C code generation for SPI peripheral based on HAL libraries and depending on STM32CubeMX configuration
  - This is a beta version not fully tested
  - Known problem: Warning generated when the SPI blockset is dragged and dropped into the application model
  - SPI example based on "ADC Click Board" from MikroElektronika
- ADC blockset modification:
  - Up to 16 regular channels and 4 injected channels outputs
  - Selected Channel output name indicates its type and rank
- Documentations updates and Simulink examples

### 3.7 Release 4.0 information

STM32 toolkit is replacing STM32F4xx and STM32F30x toolkits. It includes:

- Smooth integration of STM32CubeMX for MCU configuration and project generation.
- Processor In the Loop (PIL) with fully configurable USART/PC-COM communication.
- STM32 peripheral driver blockset library for ADCs, DACs, GPIOs, I2Cs, Timers, USARTs and xWDG.
- One STM32\_Config blockset that must be used once per Simulink application to select, create or update STM32CubeMX used configuration. \*Code Replacement Library (CRL) for sin/cos functions optimizations.
- Documentations and Simulink examples.

#### Enhancements

- One STM32 toolkit for all STM32 managed from STM32CubeMx
- Integration of STM32CubeMX for configuration code generation and project generation

### Limitations

- Project Generation:
  - Some project variable definition (Define) are not integrated to project settings and must be manually added. (cf Readme.html file paragraph "Known problems and solutions")
  - "Flash loader" is not selected for Keil® V5 generated project and must be manually added. (cf Readme.html file paragraph "Known problems and solutions")
- STM32F4xx only timers 1 to 5 and 8 are supported:
  - 3 channels only
  - PWM output mode only
  - Only the input capture mode is available to compute frequency
- STM32F30x only timers 1 to 4 and 8 are supported:
  - 3 channels only
  - PWM output mode only
  - Only the input capture mode is available to compute frequency

## 3.8 Release 3.1 information

### STM32F30x toolkit features

- Processor In the Loop (PIL) with fully configurable USART Rx/Tx Port and Speed communication
- Peripheral driver blockset for STM32F302xBxC/STM32F303xBxC devices including: ADC1 to ADC4, GPIOs, USARTs, Timers, External interrupts and ADC Interrupts management
- IAR and KEIL project generation for PIL and code application generator
- Code Replacement library (CRL) for sin and cos functions optimizations
- Documentations and Simulink Examples

### Enhancements:

#### STM32F4xx toolkit

- IAR and KEIL project generation for PIL and code application generator
- Code Replacement library (CRL) for sin and cos functions optimizations

## 3.9 Release 3.0 information

### ADC

- ADC configuration model modified, replace "ADC1\_init", "ADC2\_init" and "ADC3\_init" with:
  - a) Configuration model for common ADC feature (implement only one per application)
  - b) Same configuration model for all ADC (implement and set parameters for each ADC)
- DMA configuration for multichannel acquisition
- Number of ADC read output value is number of selected channels

### Timers

- Add Encoder mode for motor control application
- PWM output trigger out management (TRGO) to synchronize timers and ADC acquisition

### USART

- Possibility to use interrupt for receive
- Possibility to use interrupt for send

### Examples

- Motor Field Oriented Control (FOC) example
- Motor control Hall Sensor example

### Enhancements

Toolchain selection: All installed version are proposed.

## 4 Revision history

**Table 2. Document revision history**

Date	Revision	Changes
03-Apr-2013	1	Initial release.
15-Jul-2013	2	Added <i>Section 2: Release information</i> . Updated <i>Section 4: Revision history</i> .
04-Oct-2013	3	Updated <i>Section 2: Release information</i> and <i>Section 3: Information on previous releases</i> .
24-Apr-2014	4	Release 3.1 – Created <i>Chapter 2: Release information</i> for actual new features and limitations. – Created <i>Chapter 3: Information on previous releases</i> to trace previous releases content – Includes STM32F302xBxC/STM32F303xBxC.
15-Jan-2015	5	Added Release 4.0 to include STM32CubeMX
08-Jul-2015	6	Added Release 4.1: Updated: <i>Summary for STM32 embedded target for MATLAB® and Simulink® release 4.1</i> <i>Section 1.2: Host PC system requirements</i> Created <i>Section 2.1: New features</i> .
26-Oct-2015	7	Added Release 4.2 to add CAN peripheral driver.
23-Feb-2016	8	Added Release 4.3.
09-Sep-2016	9	Added release 4.4.
6-Feb-2017	10	Added release 4.4.1.
09-Aug-2017	11	Added release 4.4.2.
12-Jul-2019	12	Updated <i>General information</i> . Added release 5.1.0.

**IMPORTANT NOTICE – PLEASE READ CAREFULLY**

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, please refer to [www.st.com/trademarks](http://www.st.com/trademarks). All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2019 STMicroelectronics – All rights reserved