# Model-Based Design Toolbox S32K1xx Series

## Release Notes

**An Embedded Target for the S32K1xx Family of Processors**
**Version 4.3.0**

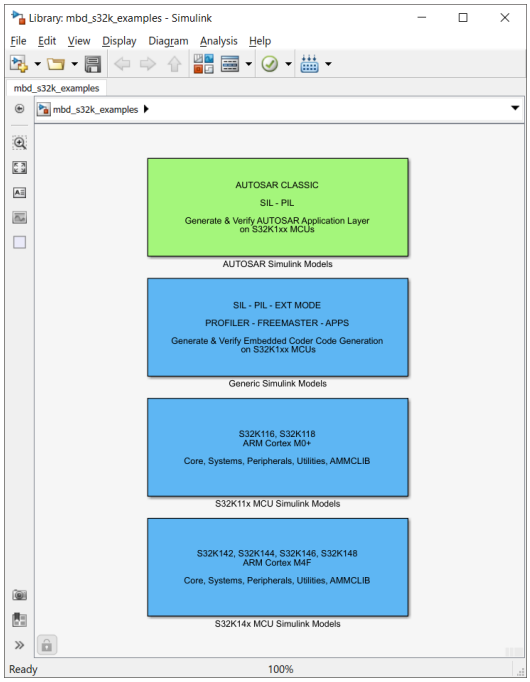
# Summary

# 1 What is New in Version 4.3.0

The latest NXP's Model-Based Design Toolbox for S32K1xx version 4.3.0 is designed to support all S32K1xx microcontrollers and brings the following important enhancements compared with the previous version (version 4.2.0)

- Added support for the RDDRONE-BMS772. This is a standalone BMS reference design suitable for mobile robotics such as drones and rovers, supporting 3 to 6 cell batteries.
- **Battery Management System (BMS)** support for **MC3377xC** – examples, documentation, and FreeMASTER UI are available.
- **Deployment** added support for JTAG deployment
- Added features that came in as **MBDT Community requests** – in our effort to make our toolbox fit the needs of our users:
- **Extended MATLAB support** to following releases: 2020b, 2021a, 2021b, 2022a
- **Expand the example library to cover all Simulink Blocks** provided as part of the NXP Toolbox for S32K1xx devices. The **example library contains approximately 300 examples** that cover a wide range of topics like:
    - I/O control: GPIO, Compare, Capture
    - Timers: Programmable (PIT), Low Power (LPTMR), Real-Time Clock (RTC), Flex (FTM)
    - Motor Control: Pulse Width Modulation (PWM), Programmable Delay (PDB), Analogue Converter (ADC), Sensors (QEP and HALL)
    - Battery management system
    - Power Management Controller (PMC)
    - Communication: CAN/CAN-FD, SPI, I2C, UART, FlexIO, LIN
    - Security: CSEC
    - Core & Systems: DMA, Registers, Memories, Watch-Dog
    - Software-in-the-Loop, Processor-in-the-Loop, and External Mode
    - SRAM and Flash Programming over UART
    - External Devices for BMS, CAN and MOSFET pre-drivers

The Simulink Models are saved in the NXP Toolbox root directory under the `S32_Examples` folder and can be accessed easily from `mbd_s32k_examples.mdl` library file.





For more details about each of the topics highlighted above please refer to the following chapters.

# 2 S32K1xx MCU Support

## 2.1 Packages & Derivatives

NXP's Model-Based Design Toolbox for S32K1xx version 4.3.0 supports:

- S32K142W MCU Packages with 64KB SRAM:
  - 48 QFP;
  - 64 LQFP;

- S32K144W MCU Packages with 64KB SRAM:
  - 48 QFP;
  - 64 LQFP;

- S32K116 MCU Packages with 17KB SRAM:
  - 32 QFP;
  - 48 LQFP;

- S32K118 MCU Packages with 25KB SRAM:
  - 48 LQFP;
  - 64 LQFP;

- S32K142 MCU Packages with 16/32KB SRAM:
  - 48 LQFP;
  - 64 LQFP;
  - 100 LQFP;

- S32K144 MCU Packages with 48/64KB SRAM**:**
  - 48 LQFP;
  - 64 LQFP;
  - 100 LQFP;
  - 100 BGA;

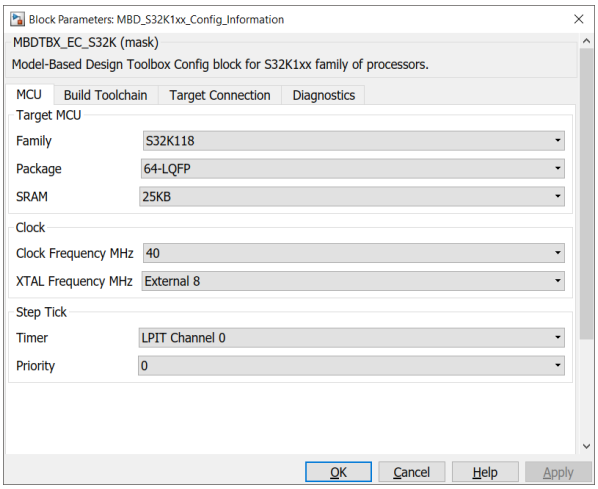
- S32K146 MCU Packages with 128KB SRAM:
  - 64 LQFP;
  - 100 LQFP;
  - 144 LQFP;
  - 100 BGA;

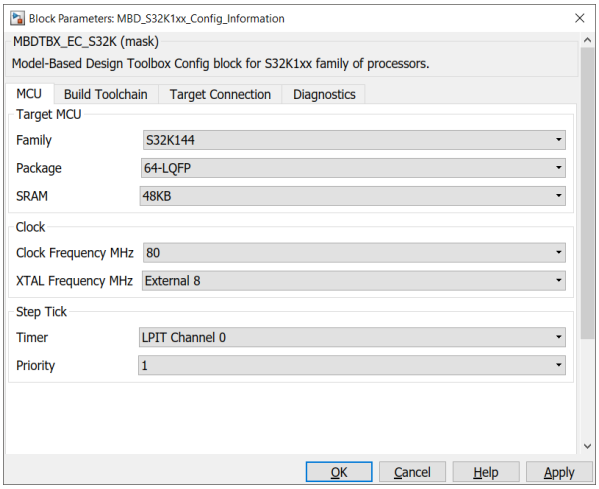
- S32K148 MCU Packages with 192/256KB SRAM:
  - 144 LQFP;
  - 176 LQFP;
  - 100 QFP;
  - 100 BGA;

The toolbox supports operations with 8MHz, 16MHz, 24MHz, 32MHz, 40MHz external XTAL and MCU system clock configuration frequencies of 40/80/112MHz depending on the type of microcontroller selected.

**Note:** *Make sure the correct XTAL Frequency is selected based on the controller type. If other frequencies are needed, the toolbox provided an easy method to address such cases. In the*

*..\src\tools\ClockConfig\ you can find S32Design Studio project that can help to regenerate the clock configuration file for the toolbox*

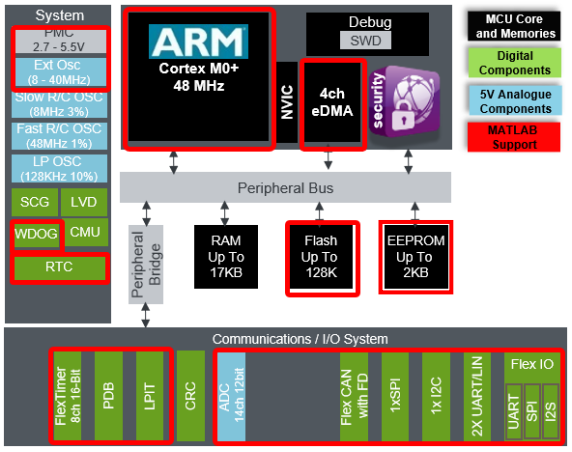


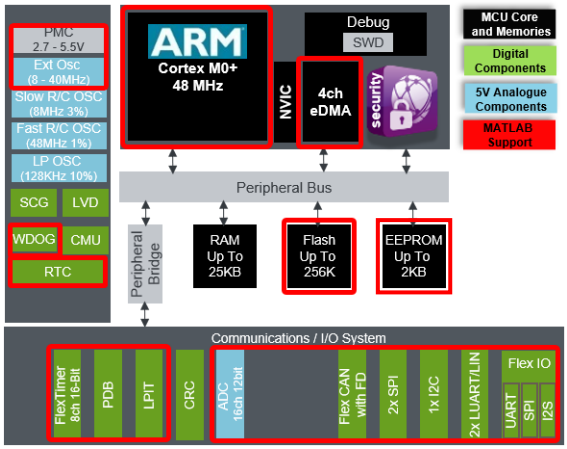
## 2.2 Peripherals & Devices

NXP's Model-Based Design Toolbox for S32K1xx version 4.3.0 supports the following peripherals and devices that are highlighted in red:

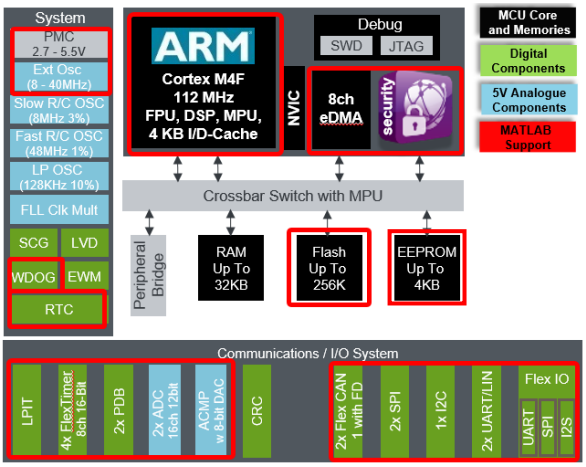
**S32K116**



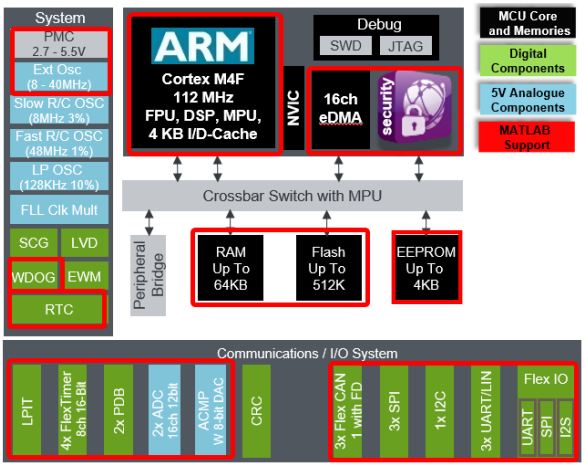

**S32K118**



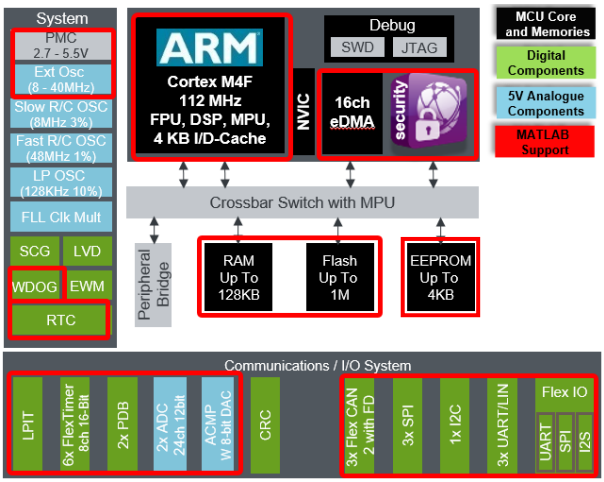

**S32K142**



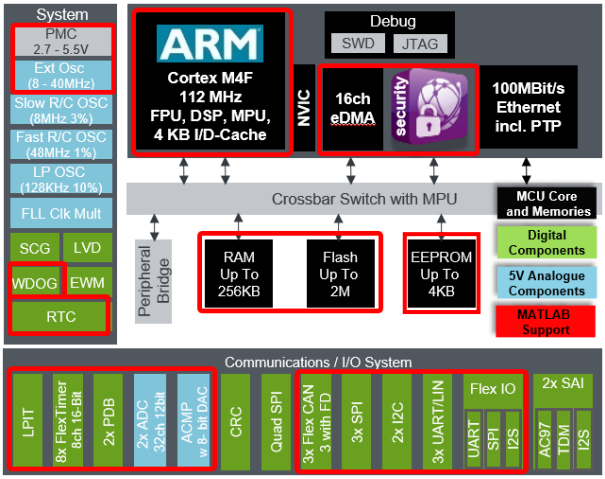

**S32K144**




**S32K146**




**S32K148**



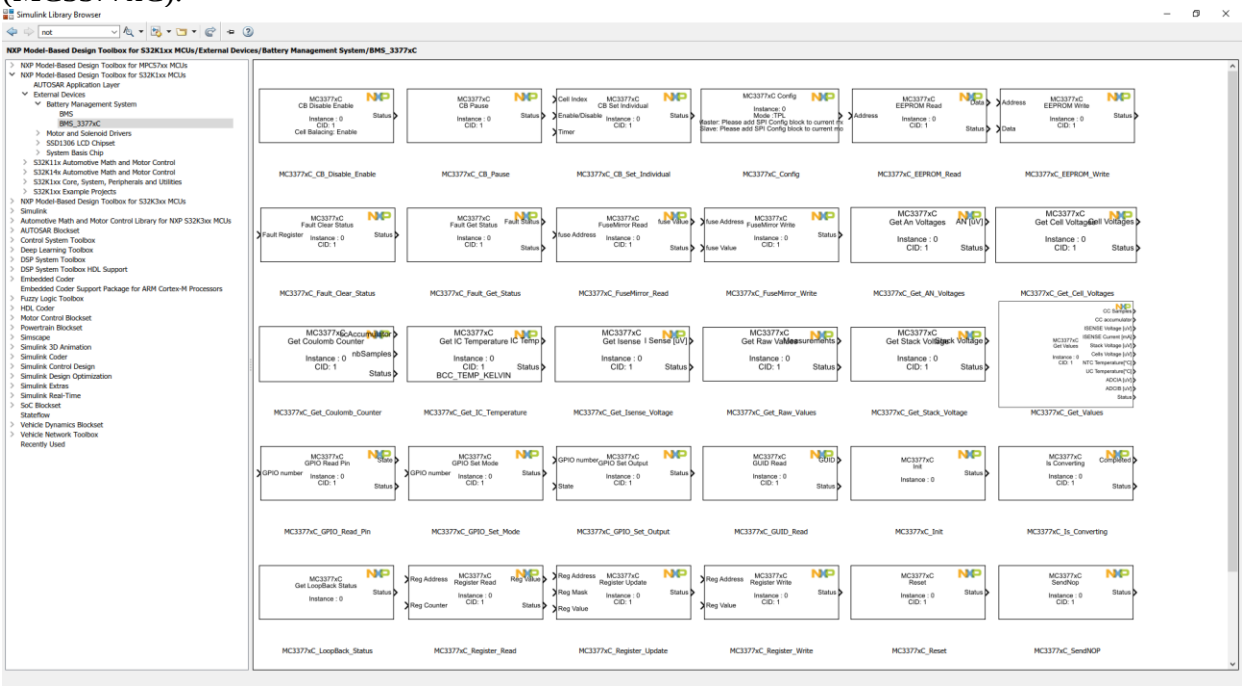
**Note:** *Model-Based Design Toolbox for S32K1xx Series does not support RAM targets and peripheral that needs RAM support on S32K116, S32K118, and S32K142 microcontrollers due to the limited RAM footprint.*

The NXP's Model-Based Design Toolbox for S32K1xx version 4.3.0 has been tested using the official NXP Evaluation Boards for S32K1xx:

- S32K116EVB rev X2, sch. Rev B
- S32K118EVB rev X2, sch. Rev B
- S32K142EVB rev X1, sch. Rev C
- S32K144EVB rev X3, sch. Rev B1
- S32K146EVB rev X1, sch. Rev B
- S32K148EVB rev X2, sch. Rev C
- S32K14WEVB rev A, sch. Rev B

### 2.2.1 New Battery Management System Support

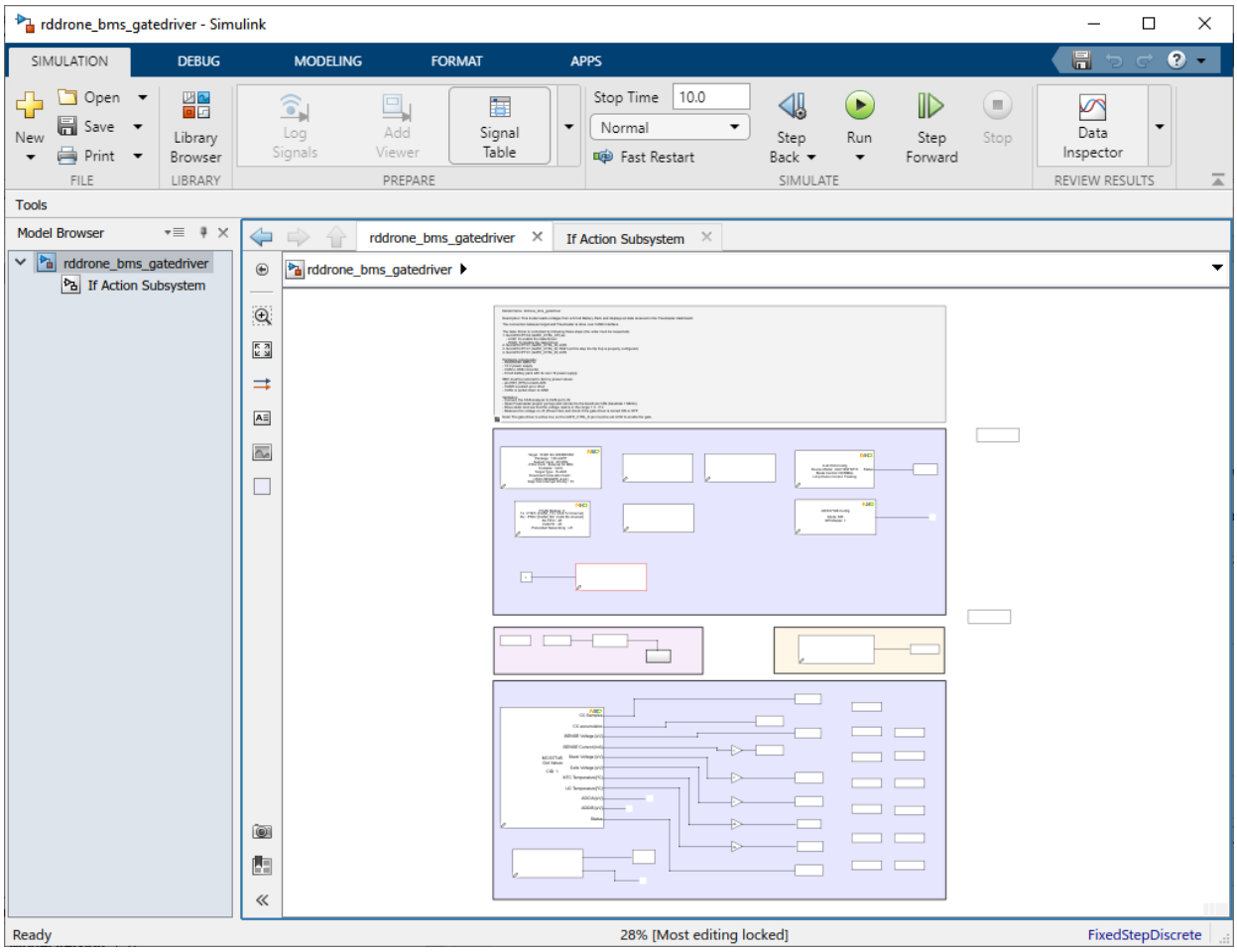
We added a set of blocks for configuring and controlling Battery Cell Controller devices (MC3377xC).




Special Simulink Blocks have been added to simplify the way a user can initialize such a controller, read vital parameters (like cell voltage, stack voltage, temperature, current) and manage the whole system. The entire suite of blocks is easily integrated within the new Battery Management Toolbox from MathWorks, so users can leverage these algorithms on top of our blocks.

### 2.2.2 New RDDRONE-772B Board Support

We added examples for the RDDRONE-BMS772. This is a standalone BMS reference design suitable for mobile robotics such as drones and rovers, supporting 3 to 6 cell batteries. MBDT examples covers applications like BMS, LCD, CAN and SBC.

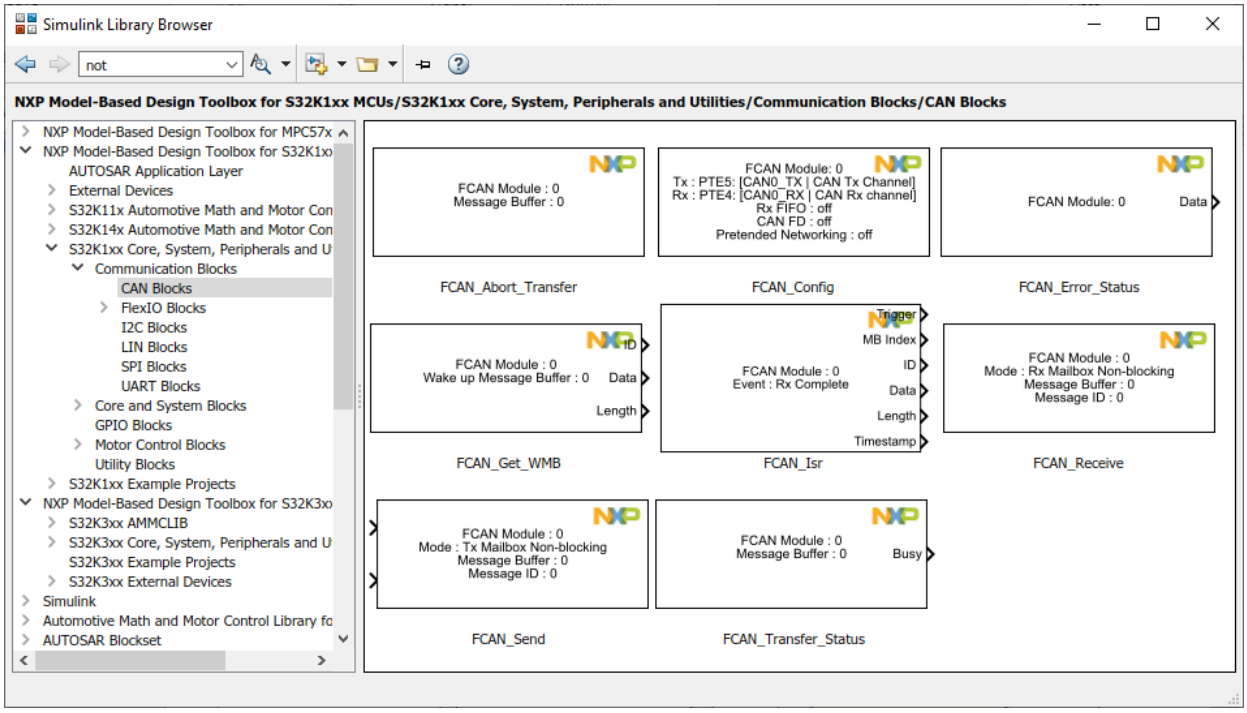



### 2.2.3 Enhanced FCAN Support

We improved the FCAN communication support by addressing the following:

- FCAN communication is now fully functional when external SBC is configured (for both UJA113x and UJA1169)
- The user is able to retrieve the ‘Error Status’ of an ongoing transfer and abort it in case something is wrong.
- The ‘Receive’ and ‘Send’ blocks for FCAN now allow the user to specify when the initialization for the message buffers should be executed (during Model Initialization, at each execution of the block, or none). If ‘none’ option is selected, a newly added block ‘FCAN_MBConfig’ can be used to configure the message buffer.

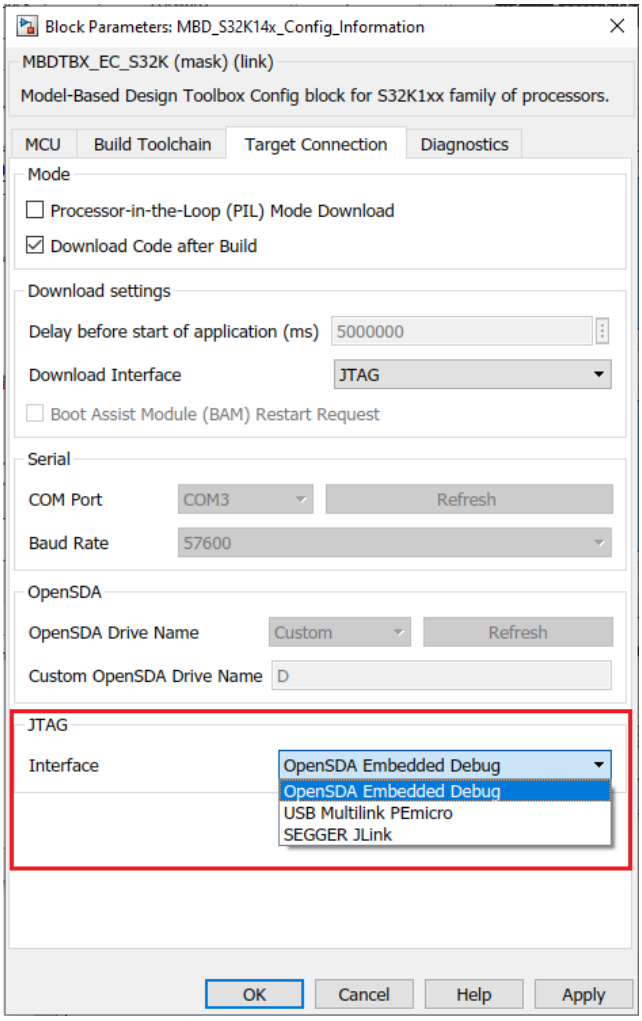
- The FCAN messages ID can also be provided via an input port for both ‘Send’ and ‘Receive’ operations.



### 2.2.4 **Added deployment over JTAG**

We added support for deployment over JTAG directly from Simulink. In the main Config block, under Target Connection, the JTAG option must be selected. Following this action, the JTAG Interface selection becomes active. Here is the following selection:

- OpenSDA Embedded Debug – the EVB Embedded Debugger is used.
- USB Multilink PEmicro – an external PEmicro probe is used.
- SEGGER JLink – this action requires the external selection of the SEGGER JLink installation path.

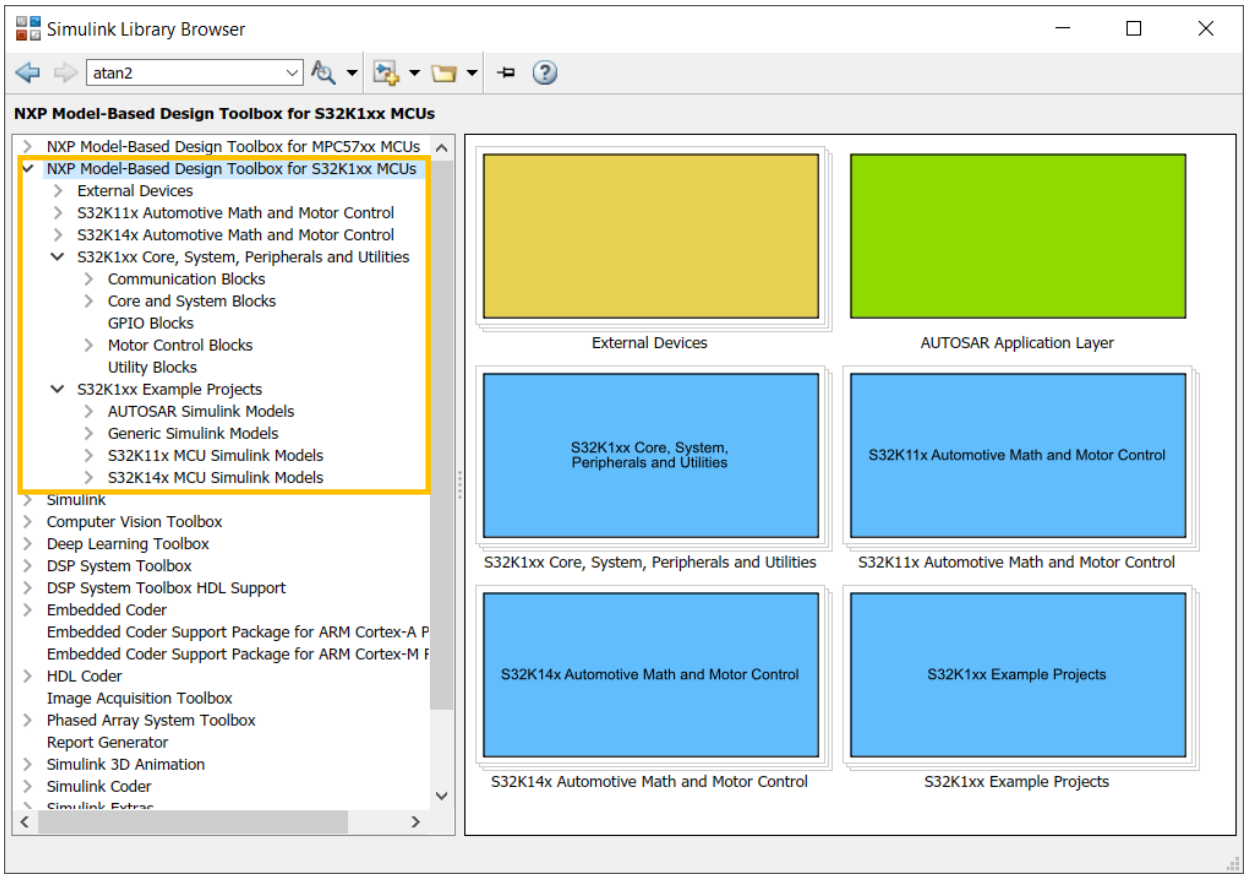


# 3 Model-Based Design Toolbox Features

NXP's Model-Based Design Toolbox for S32K1xx version 4.3.0 is delivered with a complete S32K1xx MCU Simulink Block Library as shown below.

There are six main categories:

- **S32K1xx Core, System, Peripherals and Utilities** which contains all blocks related with MCU configuration
- **S32K1xx Example Projects** which contains all the examples that exercise all the other blocks
- **S32K14x and S32K11x Automotive Math and Motor Control** which generic 16bit, 32bit, and floating-point single precision blocks
- **External Devices** which contains blocks related to System Basis Chips, Three-phase Pre-drivers configuration, and Battery Management Systems
- **AUTOSAR Application Layer** which contains configuration blocks for Classic AUTOSAR PIL simulations



## 3.1 New features requested by the community

In our efforts to make NXP's Model-Based Design Toolbox for S32K1xx version 4.3.0 fit the needs of the users, we have added the following requested features/updates:

- Added OSIF Time Delay Block
- Added Code Size Report
- Added NONE option for the SPI SIN/SOUT pins
- Enabled support for more than one ADC Interleave Block
- Updated Profiler functionality
- Updated FCAN Blocks
- Updated I2C Blocks
- Updated LPUART Blocks
- Fixed Registers dropdowns not displaying options on some machines

## 3.2 Extended MATLAB version support

We extended support for our toolbox to cover a wider range of MATLAB releases – starting from R2016a and going up to R2022a. This way we want to avoid locking out users that have constraints regarding MATLAB versions.

## 3.3 S32K1xx Automotive Math and Motor Control

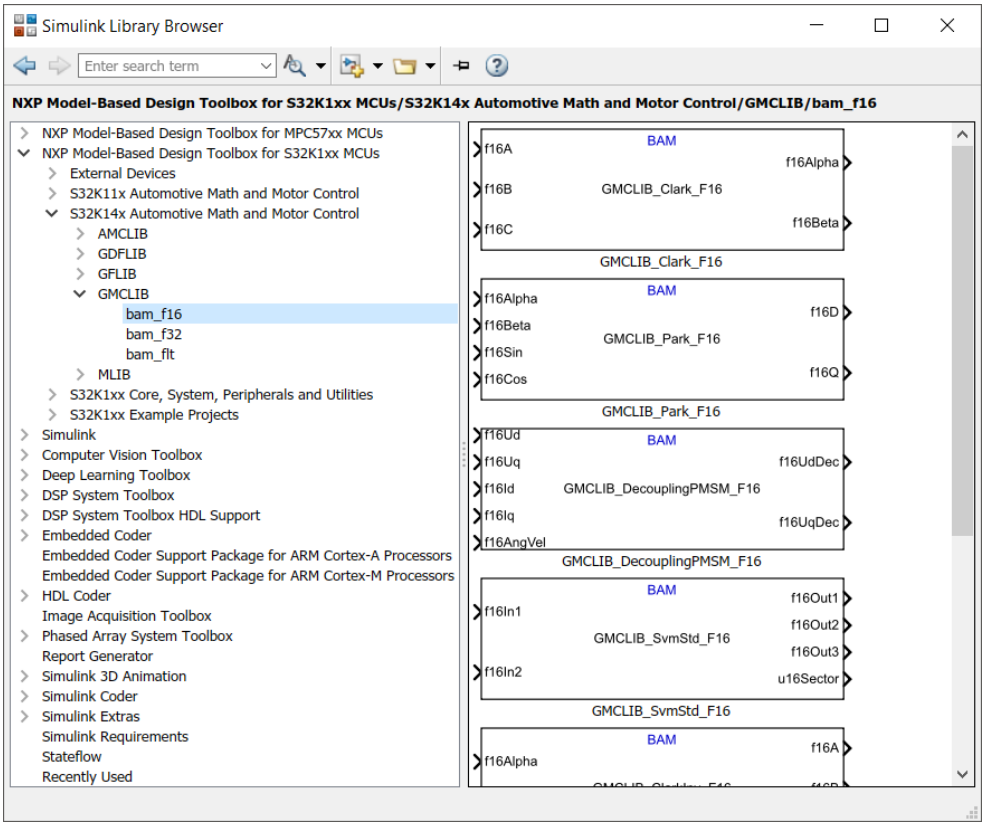
All functions in the Automotive Math and Motor Control Functions Library v1.1.21 are supported as blocks for simulation and embedded target code generation for:

- Bit Accurate Model for 16-bit fixed-point implementation;
- Bit Accurate Model for 32-bit fixed-point implementation;
- Bit Accurate Model for floating-point single precision implementation;

There are two different AMMCLIB libraries integrated with the NXP Model-Based Design Toolbox S32K1xx Simulink Library. Depending on the microcontroller in use, you need to select between:

- S32K11x Automotive Math and Motor Control
- S32K14x Automotive Math and Motor Control

**Note:** *Due to Simulink limitations it is not possible to mix the AMMCLIB blocks from S32K11x with the ones from S32K14x. Depending on the MCU selection made in main configuration block one version of the other of the AMMCLIB is active*



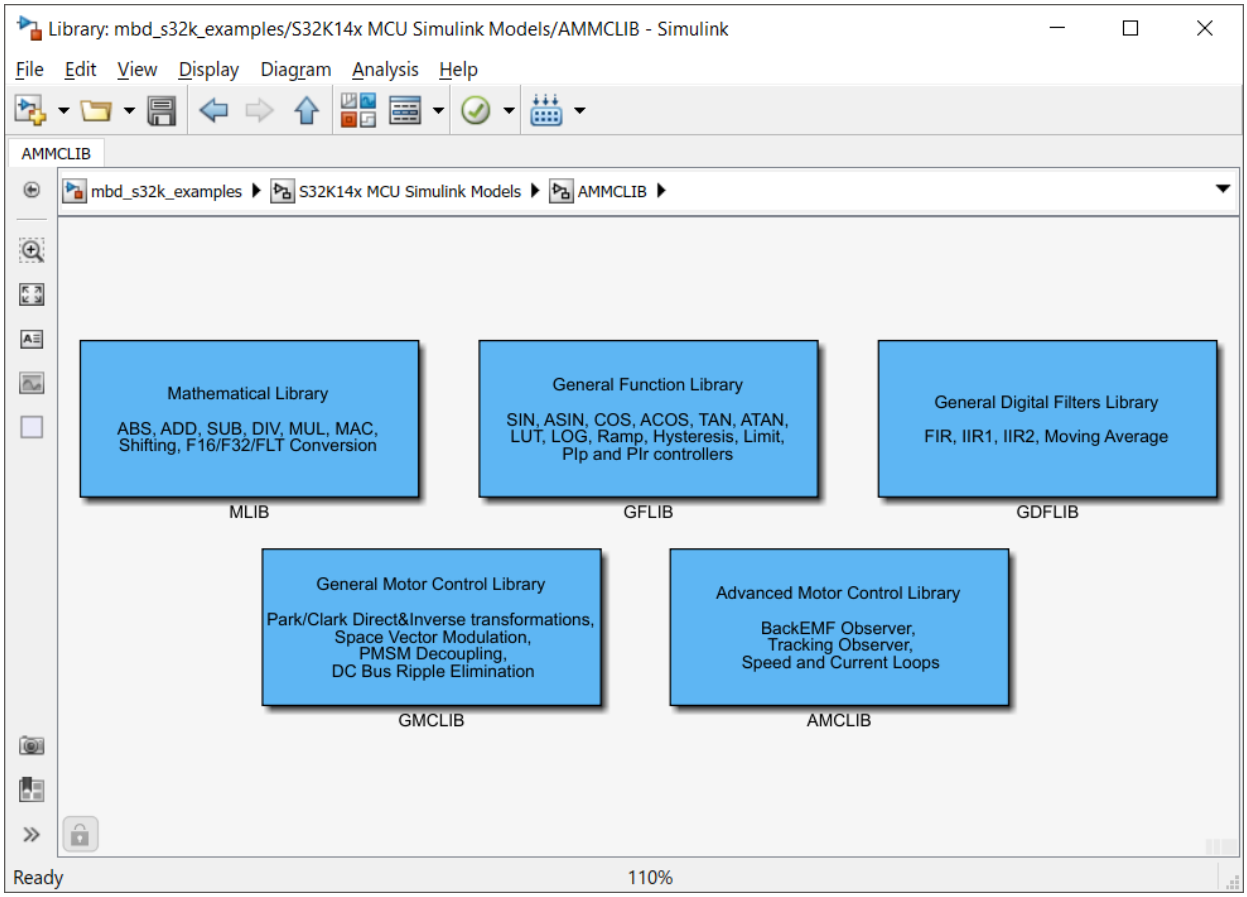

The main functionalities supported are:

- Mathematical Function Library (MLIB) – supports basic mathematical operations such as addition, multiplication, etc;
- General Function Library (GFLIB) – supports basic trigonometric and general math functions such as sine, cosine, tan, hysteresis, limit, etc;
- General Digital Filters Library (GDFLIB) – comprising digital IIR and FIR filters designed to be used in a motor control application;
- General Motor Control Library (GMCLIB) – supports standard algorithms used for motor control such as Clarke/Park transformations, Space Vector Modulation, etc;
- Advanced Motor Control Function Library (AMCLIB) – comprising advanced algorithms used for motor control purposes;

For more details about the latest changes check the:

- `\mbdtbx_s32k11x\AMMCLIB_s32k11x\S32K11x_AMMCLIB_RTM_1_1_21_ReleaseNotes.txt`
- `\mbdtbx_s32k14x\AMMCLIB_s32k14x\S32K14x_AMMCLIB_RTM_1_1_21_ReleaseNotes.txt`

For each AMMCLIB block, there is a Simulink Example available NXP Toolbox root directory `..\S32_Examples\common\` folder.

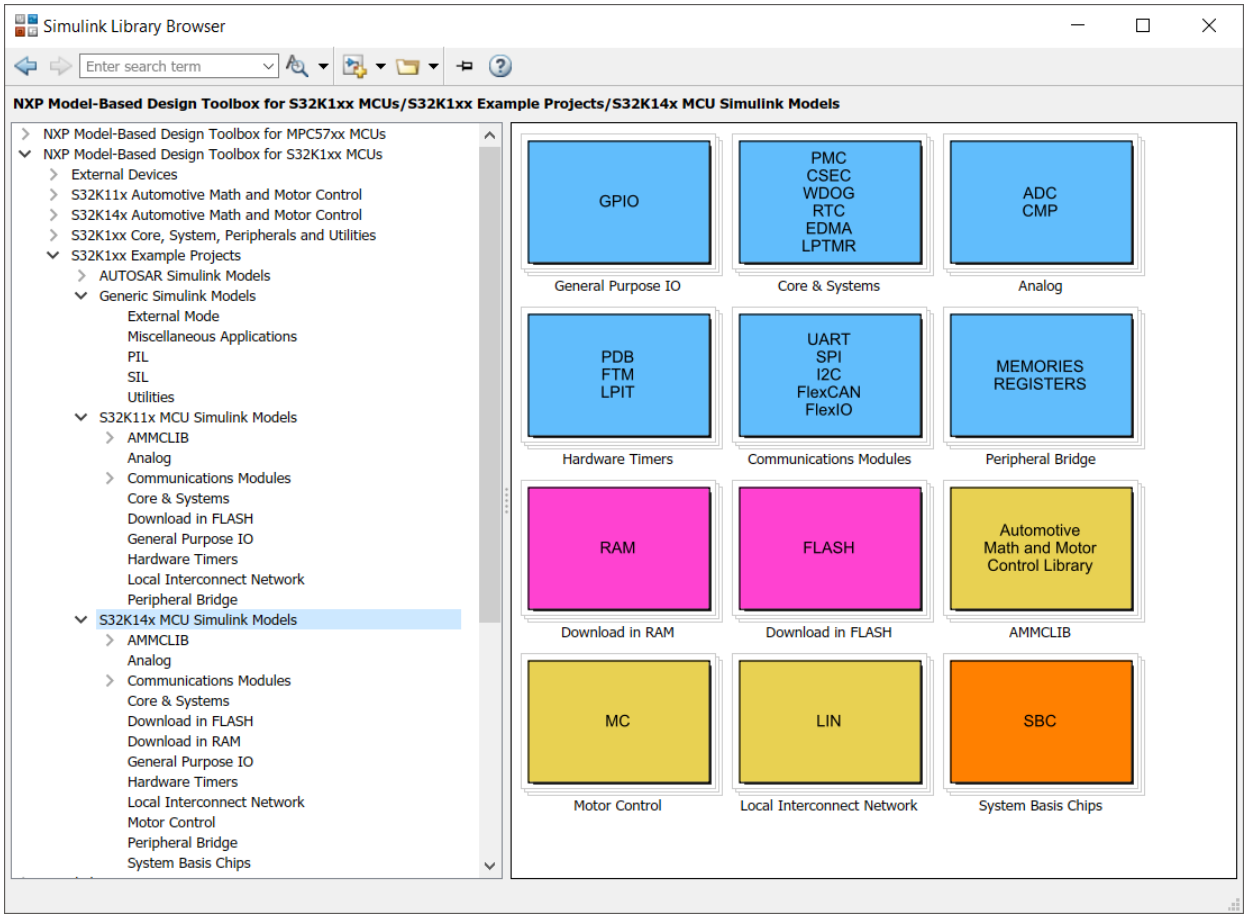

Library: mbd_s32k_examples/S32K14x MCU Simulink Models/AMMCLIB - Simulink
File Edit View Display Diagram Analysis Help
AMMCLIB
mbd_s32k_examples
S32K14x MCU Simulink Models
AMMCLIB
Mathematical Library
ABS, ADD, SUB, DIV, MUL, MAC, Shifting, F16/F32/FLT Conversion
MLIB
General Function Library
SIN, ASIN, COS, ACOS, TAN, ATAN, LUT, LOG, Ramp, Hysteresis, Limit, PIp and PIr controllers
GFLIB
General Digital Filters Library
FIR, IIR1, IIR2, Moving Average
GDFLIB
General Motor Control Library
Park/Clark Direct&Inverse transformations, Space Vector Modulation, PMSM Decoupling, DC Bus Ripple Elimination
GMCLIB
Advanced Motor Control Library
BackEMF Observer, Tracking Observer, Speed and Current Loops
AMCLIB
Ready
110%

## 3.4 S32K1xx Example Library

S32K Examples Library represents a collection of Simulink models that let you test different MCU on-chip modules and run complex applications. The example library is split into four categories:

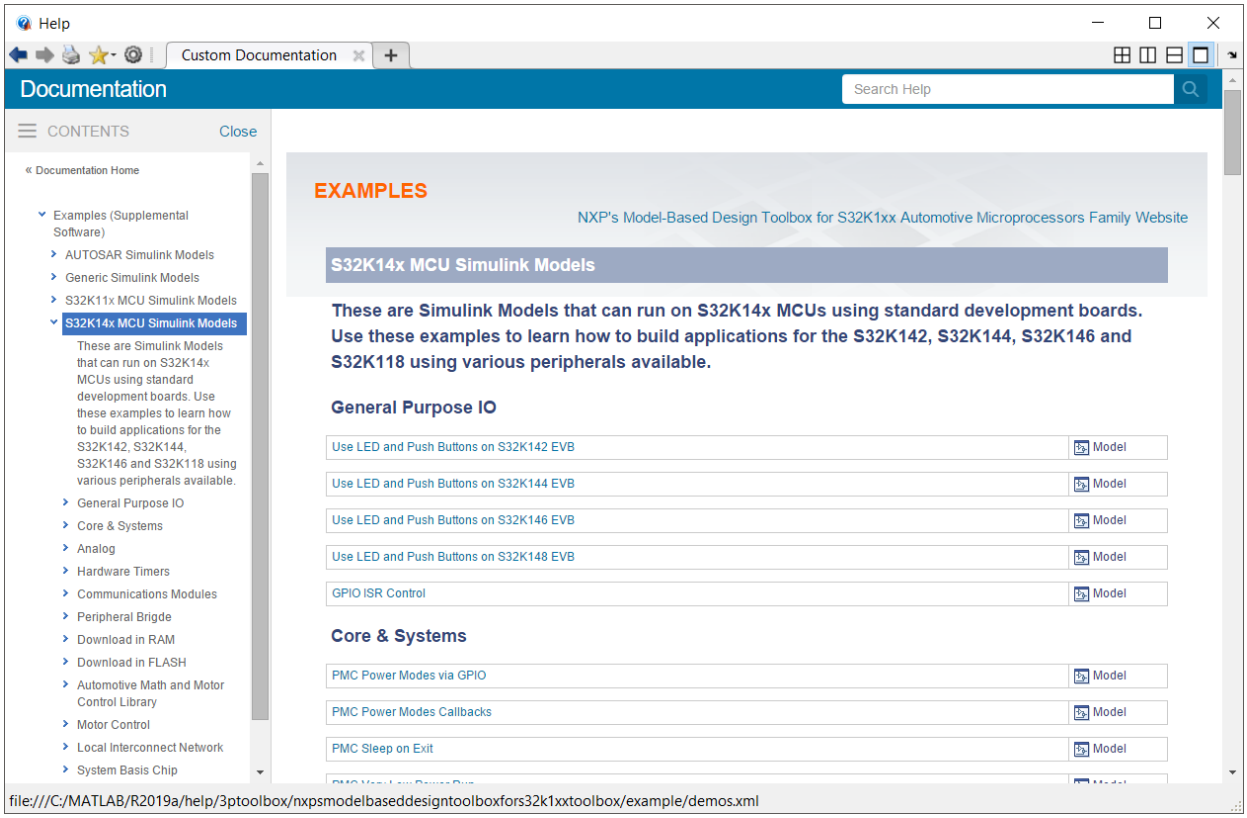
- AUTOSAR Simulink Models that allows PIL simulations for Classic AUTOSAR applications
- Generic Simulink Models that show how to configure the S32K1xx for SIL, PIL, External Mode and other applications that can run NXP devices
- S32K11x MCU Simulink Models that can be run on any of the S32K11x Evaluation Boards
- S32K14x MCU Simulink Models that can be run on any of the S32K14x Evaluation Boards

The examples are grouped in different layers that mimics a typical development flow: starting with basic building blocks that expose the MCU HW functionalities, build SIL and PIL models for verification and validation purposed, and ending up with more complex applications that incorporate multiple building blocks.




The Simulink models shown as examples are enhanced with a comprehensive description to help users understand better the functionality that is exercised, hardware setup instructions whenever are necessary, and a result validation section.

The examples are available from the MATLAB help page too:



# 4 New Model-Based Design Toolbox Extras

NXP's Model-Based Design Toolbox for S32K1xx version 4.3.0 enables additional functionalities that are not MCU specific but could help faster prototyping, validation, and verification of the developed models.

## 4.1 Processor-In-the-Loop Support (PIL)

The NXP's Model-Based Design Toolbox for S32K1xx version 4.3.0 provides PIL support for purposes of ASIL software development processes, "Model PIL Block" (Model Reference), and "PIL Block" modes of operation are supported "Top Model PIL" mode is not supported. PIL contains full support for Math and Motor Control Blocks and limited support for peripheral blocks.

For PIL support you need to install the MATLAB Support for MinGW-w64 C/C++ Compiler toolbox All models, mex files generation and tests were developed using the MinGW Compiler

An alternative is Microsoft Windows SDK 7.1. Please refer to the following links for troubleshooting:

- https://www.mathworks.com/matlabcentral/answers/95039-why-does-the-sdk-7-1-installation-fail-with-an-installation-failed-message-on-my-windows-system
- https://www.mathworks.com/matlabcentral/answers/101105-how-do-i-install-microsoft-windows-sdk-7-1

## 4.2 Boot Loader

Internal Boot Loader is a standalone application which requires the Microsoft .NET Framework version 4.0 installed on PC. The board should be configured to work with BAM to use Internal Boot Loader. Please check the board documentation.

The bootloader files located under `..\tools\BootLoader\RBF_Files\` have to need updated to support all S32K1xx derivatives and allows applications to be downloaded via UART or CAN interface to the boards.

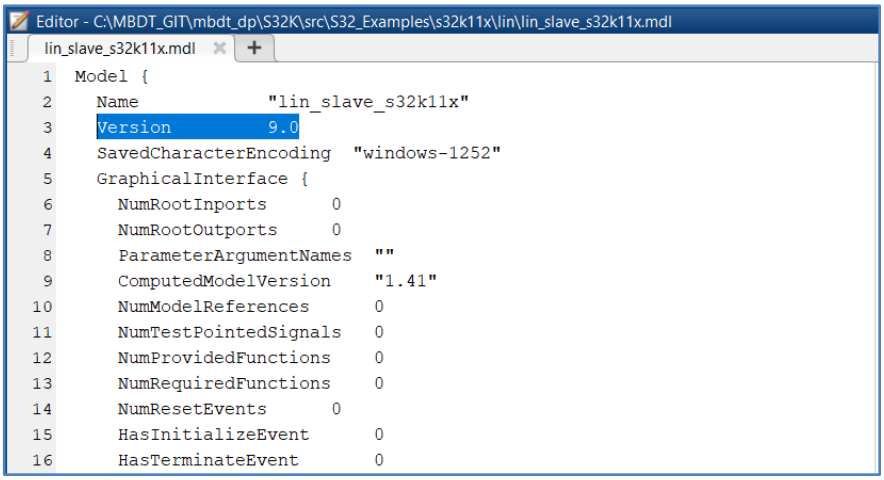
# 5 Prerequisites

## 5.1 MATLAB Releases and OSes Supported

This toolbox is developed and tested to supports the following MATLAB releases:

- R2016a;
- R2016b;
- R2017a;
- R2017b;
- R2018a;
- R2018b;
- R2019a;
- R2019b;
- R2020a;
- R2020b;
- R2021a;
- R2021b;
- R2022a;

In general, the older version of MATLAB may be supported with small updates that can be done after installation. As a rule, if you are planning to use an older version you need to update the Simulink version in every MDL file delivered with the toolbox




For a flowless development experience the minimum recommended PC platform is:

- *Windows® OS*: any x64 processor
- At least 4 GB of RAM
- At least 6 GB of free disk space.
- Internet connectivity for web downloads.

**Operating System Supported**

| | SP Level | 64-bit |
|---|---|---|
| Windows 7 | SP1 | X |
| Windows 10 | | X |

## 5.2 Compiler Support

The following compilers are supported:

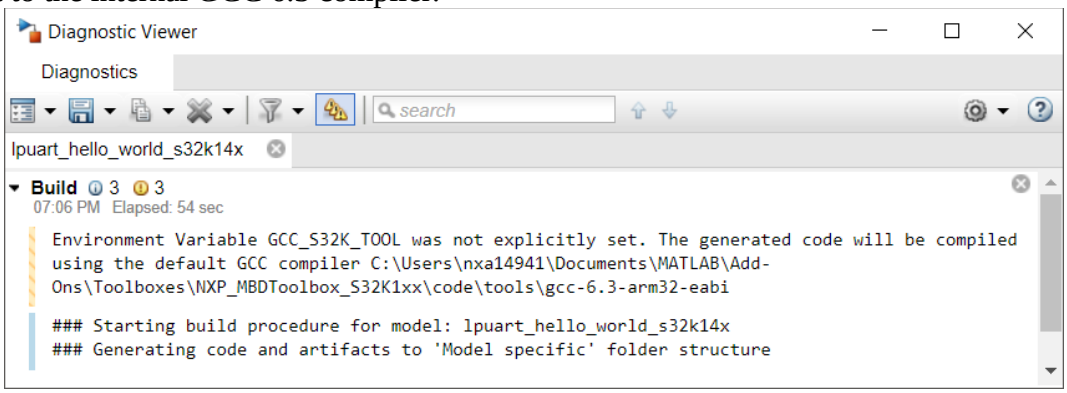
| Compiler Supported | Release Version |
|---|---|
| GCC for ARM Embedded Processors | V6.3.1 |
| IAR ANSI C/C++ Compiler (only for S32K14x) | V8.11.2 |
| GreenHills MULTI for ARM | 2017.1.4 |

The target compiler for the Model-Based Design Toolbox needs to be configured. Use the notation below to set up these compiler environmental variables. Ensure a system environment variable called <COMPILER_STRING>_TOOL, corresponding to the compiler(s) you have installed, is defined to compiler path value as shown below:

```
GCC_S32K_TOOL = {Toolbox installation path}\tools\gcc-6.3-arm32-eabi
IAR_TOOL = {IAR installation path}/IAR Systems/Embedded_Workbench
GHS_TOOL = {GHS installation path}/multi
```

Note: Paths shown are for illustration, your installation path may be different. Once environmental variables are setup you will need to restart MATLAB for the IDE environment to see these system variables.

In case there is no compiler installed, the NXP's Model-Based Design Toolbox is going to default to the internal GCC 6.3 compiler.



# 6 Known Limitations

NXP's Model-Based Design Toolbox for S32K1xx version 4.3.0 has the following limitations:

- Version 4.3.0 cannot coexist with the previously installed version of the NXP Model-Based Design Toolbox for S32K1xx. When the toolbox is installed as Add-Ons the MATLAB will uninstall the previous version automatically. In case you have files, custom files saved with previous versions will not be deleted.

  To avoid any complications, it is recommended to uninstall the previous toolboxes for S32K1xx manually from Add-on Manager before install the 4.3.0 release

- Download to target for Simulink is supported only via the UART interface. For CAN download, you can use the RAppID Bootloader as a separate tool to load the generated application into target RAM/Flash Memories.
- Due to SRAM memory sizes available on S32K116, S32K118, and S32K142 the support for generating the application in RAM from Simulink has been disabled. There code size of the applications does not fit into the target SRAM memory.
- Once the CSEC is enabled the only reliable way to reprogram the target is to use the UART communication channel.
- For SBC testing the correct factory settings are required. FNMC bit must be disabled, SBC_UJA_SBC_SDMC_DIS must be disabled, and Sleep Control Bit SLPC must be allowed. For running other applications is recommended to move the SBC in FNMC after running this example, otherwise, CAN transceiver can be disabled because SBC goes in LIMP mode due to lack of watchdog feed. The S32K142 and S32K144 Evaluation might fail to exit from reset mode.
- All examples were tested by default on the S32K144 Evaluation Board. Some examples might need manual settings when compiled and run for other versions of the boards.
- Flash partitioning on S32K144 via OpenSDA does not work - only via serial communication

# 7 Support Information

For technical support please sign on to the following NXP's Model-Based Design Toolbox Community: https://community.nxp.com/community/mbdt

***How to Reach Us:***

**Home Page:**
www.nxp.com

**Web Support:**
www.nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

NXP Semiconductor reserves the right to make changes without further notice to any products herein. NXP Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals”, must be validated for each customer application by customer’s technical experts. NXP Semiconductor does not convey any license under its patent rights nor the rights of others. NXP Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the NXP Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use NXP Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold NXP Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that NXP Semiconductor was negligent regarding the design or manufacture of the part.

MATLAB, Simulink, Stateflow, Handle Graphics, and Real-Time Workshop are registered trademarks, and TargetBox is a trademark of The MathWorks, Inc.
Microsoft and .NET Framework are trademarks of Microsoft Corporation.
Flexera Software, Flexlm, and FlexNet Publisher are registered trademarks or trademarks of Flexera Software, Inc. and/or InstallShield Co. Inc. in the United States of America and/or other countries.
NXP, the NXP logo, CodeWarrior and ColdFire are trademarks of NXP Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. Flexis and Processor Expert are trademarks of NXP Semiconductor, Inc. All other product or service names are the property of their respective owners