# **Model-Based Design Toolbox**

## **Quick-Start**

An Embedded Target for the S32K1xx Family of Processors Version 2018.R1

**Target Based Automatic Code Generation Tools** For MATLAB<sup>TM</sup>/Simulink<sup>TM</sup>/Stateflow<sup>TM</sup> Models working with Simulink Coder <sup>TM</sup> and Embedded Coder®



## Summary

1	Inst	allation	
	1.1	System Requirements	
	1.2	Installation Steps	
	1.2.	1 Run Add-on installer	
	1.2.	2 License Registration & Installation	
	1.2.	3 Setting up the Target Compilers	
	1.2.	4 Setting the Path for Model-Based Design Toolbox	
2	Rur	1 Models	
	2.1	Examples Library & Help	
	2.2	Hardware Setup	
	2.3	Hello World Example	
		•	

### **1** Installation

Installing the Model-Based Design Toolbox is the first step in setting up and running automatic C code generation from MATLAB/Simulink for NXP's embedded target processors and development boards.

#### **1.1 System Requirements**

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	Х
Windows 10		Х

#### **1.2 Installation Steps**

NXP's Model-Based Design Toolbox is delivered as MATLAB Toolbox Package that can be installed offline or online from MathWorks Add-ons. This document shows how to install the offline package, assuming you have already downloaded the file from NXP's official web page.

To have the toolbox installed and configured properly the following actions should be executed:

- 1. Run the MATLAB toolbox package file \*.mltbx downloaded from <u>NXP's Model-</u> Based Design Toolbox web page
- 2. Register and install the toolbox license file into ... My Documents \MATLAB \Add-Ons \Toolboxes \NXP MBDToolbox S32K1xx \code \lic
- 3. Set the Target Compiler Environment Variables
- 4. Setup the MATLAB path for Model-Based Design Toolbox

Each of these actions are explained in the following sub-chapters.

#### 1.2.1 Run Add-on installer

Install the NXP's Model-Based Design Toolbox by double clicking the \*.mltbx file. This will activate the MATLAB Add-ons installer that will automatically start the installation process.

After the MATLAB opens, you will be prompted with the following options:

1. The NXP's Model-Based Design Toolbox Installation Wizard dialog will appear. Click "Install" to proceed.



2. Indicate acceptance of the NXP Software License Agreement by selecting "I agree to the terms of the license" to proceed.

4	License Agreement: NXP_MBDToolbox_S32K1xx	$\times$
	IMPORTANT. Read the following NXP Software License Agreement ("Agreement") completely. By selecting the "I Accept" button at the end of this page, you indicate that you accept the terms of the Agreement and you acknowledge that you have the authority, for yourself or on behalf of your company, to bind your company to these terms. You may then download or install the file.	^
	NXP SOFTWARE LICENSE AGREEMENT	
	This is a legal agreement between you, as an authorized representative of your employer, or if you have no employer, as an individual (together "you"), and NXP B.V. ("NXP"). It concerns your rights to use the software identified in the Software Content Register and provided to you in binary or source code form and any accompanying written materials (the "Licensed Software"). The Licensed Software may include any updates or error corrections or documentation relating to the Licensed Software provided to you by NXP under this License. In consideration for NXP allowing you to access the Licensed Software, you are agreeing to be bound by the terms of this Agreement. If you do not agree to all of the terms of this Agreement, do not download or install the Licensed Software. If you change your mind later, stop using the Licensed Software and delete all copies of the Licensed Software in your possession or control. Any copies of the Licensed Software that you have already distributed, where permitted, and do not destroy will continue to be governed by this Agreement.	
	1 DEFINITIONS	~
	✓ I agree to the terms of the lice	nse.
	<u>Q</u> K <u>C</u> ancel	

3. Click "OK" to start the MATLAB installation process. The rest of the process is silent and under MATLAB control. All the files will be automatically copied into default Add-Ons folder within the MATLAB

The default location can be changed prior to installation by changing the Add-Ons path from MATLAB Preferences

📣 Preferences	- 🗆 X
<ul> <li>MATLAB</li> </ul>	MATLAB Add-Ons Preferences
Add-Ons App Designer Code Analyzer > Colors Command History Command Window Comparison Current Folder > Editor/Debugger	Installation Folder This is where MATLAB puts installed Add-Ons. MATLAB always searches this folder for installed Add-Ons. C\Users\nxa14941\Documents\MATLAB\Add-Ons

4. After a couple of minutes (2-5min), the NXP's Model-Based Design Toolbox should be visible as a new Add-ons.

						8		901	3
Simulink	Layout	Ø Preferences Set Path	Add-Ons	? Help ▼	Community	rt			
SIMULINK		ENVIRONMENT	臱 Ge	et Add-O anage Ac	ns dd-Ons		_	_	
			Pa	ickage To	oolbox				
			Pa	ckage Ap	op are Support Packages	5			
			Check for	Updates		>			

Add-On Manager –							
Installed	Get Add-Ons	Import					
	Name	Туре	Author	Install Date 💌			
	NXP_MBDToolbox_S32K1xx version 4.0.0	Toolbox	NXP Model- Based Design Toolbox Team	22 July 2018	:		
	NXP_Vision_Toolbox_for_\$32V234 version 1.0.0	Toolbox	NXP Model- Based Design Toolbox Team	20 June 2018	:		
	NXP Support Package S32V234 version 1.0.0	Toolbox	NXP Model- Based Design Toolbox Team	20 June 2018	:		

5. More details about the NXP's Model-Based Design Toolbox can be found by clicking on View Details

📣 Add-On Manager		-		×
< Back				
(2)	NXP_MBDToolbox_S32K1xx version 4.0.0 by NXP Model-Based Design Toolbox Team installed	i on 22 .	July 2018	
	Generate code optimized for NXP's S32K1xx Automotive General Purpose Microcontrollers			
	Toolbox Open Folder Learn More 👻	Unir	istall	
Description				
The NXP's Model-Based It provides an integrated routines and device driv FlexIO and sensor base	d Design Toolbox is a quick solution for prototyping applications on top of NXP's MCUs. I development environment and tool chain support for configuring and generating applications (including in ers) to execute complex applications (e.g.: motor control algorithms, communication protocols CAN, SPI, I d applications).	iitializa 2C, UA	tion ART,	
This toolbox includes int Motor Control Library, G Software and Processor algorithms developed in	tegrated Simulink® embedded target for NXP S32K1xx MCUs, peripheral device blocks and drivers, the M CC compiler and additional tools for debugging and real time data visualization. It provides built-in suppo -in-the-Loop (SIL and PIL) simulations to enable fast prototyping, verification and validation on the real tar MATLAB environment.	lath an rt for get for	d the	
Visit https://community.n	xp.com/community/mbdt - for examples and support.			
S32K1xx Model-Based I Generate code for st Optimized motor con Core&System blocks System Basic Chips On-target profiling and	Design Toolbox Main Features: andalone application with direct download to target support trol library blocks including Park/Clarke transforms, digital filters, and general functions ; for CAN, SPI, I2C, FlexIO, UART, PIT, LPTMR, FTM, PWM, CSEC, WDOG, PMC, PDB, ADC, CMP, DM/ support for NXP analogue devices functions and tasks calibration using EreeMASTER tool	A, RTC		
<ul> <li>Data acquisition and</li> <li>External Mode support</li> <li>Boot loader utility for</li> <li>Seamless integration</li> <li>Ready to run example</li> </ul>	calibration using Preevices FER tool ort programming application in FLASH or SRAM via UART or CAN with embedded coder including SIL and PIL test les for all supported peripherals and Simulink Blocks			
► View File List				

6. NXP's Model-Based Design Toolbox documentation, help and examples are fully integrated with MATLAB development environment. Get more details by accessing the standard Help and Supplemental Software section



#### 1.2.2 License Registration & Installation

The NXP's Model Based Design Toolbox is available free of charge, however, a license is required. If you need to get a license it can be obtained by following the path outlined below. If you encounter issues getting a license please submit a ticket at <u>https://www.nxp.com/support/support:SUPPORTHOME</u> describing the issue.

Please perform the following steps to obtain your license:

1. Go to <u>www.nxp.com/mbdt</u> and click on "Download Eval"

#### Model-Based Design Toolbox

OVERVIEW	DOCUMENTATION	DOWNLOADS	DEVELO	OPMENT TOOLS	TRAINING & SUPPORT	
Jump To Overview & Features Supported Devices	Overview The NXP's Mode integrated develo	I-Based Design Toolbox pro-	vides an chain for	Features Generate code for standalone application with direct download to target support		
Target Applications System Requirements	configuring and generating all of the necessary software automatically (including initialization routines and device drivers) to execute complex applications (e.g.: motor control algorithms, communication protocols CAN, SPI, 100, UADT and exchange the indexistencies NNR			<ul> <li>Optimized motor control library blocks including Park/Clarke transforms, digital filters, and general functions</li> <li>I/O blocks including CAN_SPL_PIT timer_Sine Wave</li> </ul>		
	MCUs.	embedded	Generation, eTimer, PWM and A/D. <ul> <li>On-target profiling of functions and tasks</li> </ul>			
	target for NXP MCUs, peripheral device blocks and drivers, the Math and Motor Control library set and bit-		t and bit-	<ul> <li>Data acquisition a tool</li> </ul>	nd calibration using FreeMASTER	
	for Software and			Boot loader utility		
	User Guid	le Downloa	d Eval			

2. Once you have logged in, the "Product Information" page for Model-Based Design Toolbox appears.

Select the appropriate toolbox you wish you evaluate to bring up the Software Terms and Conditions page.

- 3. Click "I Agree" to consent to the software license agreement.
- 4. If you need to download the tool, click on the linked file name. Otherwise, click on "License Keys" tab.
- 5. Verify the correct tool is identified and click on "Generate".
- 6. Enter your Disk Serial Number as the Host ID. If you do not know your Disk Serial Number, go to "Locating the Host ID" in the document **Model-Based Design Toolbox License Installation** to learn how to locate your Disk Serial Number, which is needed to generate your license.
- 7. Click "Generate"
- 8. Either click on "Save All" or copy and paste the file into a text editor and save the file as "license.lic" to the folder "..\MATLAB\Add-Ons\Toolboxes\NXP MBDToolbox S32K1xx\code\lic"

⊠<

#### 1.2.3 Setting up the Target Compilers

The target compiler for Model-Based Design Toolbox needs to be configured. Use the notation below to setup 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 7.3
GHS TOOL = {GHS installation path}/multi517
```

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.



#### 1.2.4 Setting the Path for Model-Based Design Toolbox

If the toolbox installation as Add-ons was successful, then the MATLAB will automatically detect the NXP's Model-Based Design Toolbox. Nonetheless, in special circumstances you might need to run a special script to instruct the MALAB to recognize the NXP's Model Based Design Toolbox. This is done by changing the MATLAB Current Directory to the toolbox installation directory (e.g.: ..\MATLAB\Add-Ons\Toolboxes\NXP\_MBDToolbox\_S32K1xx\code) and running the "mbd\_s32k path" script.

```
>> mbd_s32k_path
Treating 'C:\Users\b21307\Documents\MATLAB\Add-
Ons\Toolboxes\NXP_MBDToolbox_S32K1xx\code' as MBD Toolbox installation
root.
MBD Toolbox path prepended.
Successful.
>>
```

## 2 Run Models

#### 2.1 Examples Library & Help

NXP's Model Based Design Toolbox comes with an Examples Library collection that let you test different MCU on-chip modules and run complex applications.

The Examples Library mbd\_s32k\_examples.mdl can be opened from "{Model Based Design Install Directory}\mbdtbx\_S32K\mbdtbx\_s32k14 \Examples\" folder.



Each category contains multiple examples that showcase different Model-Based Design Toolbox capabilities that are clarified in three layers:

- **On-chip modules** and peripherals examples covers the basic functionalities of the MCU;
- **Modelling, Verification and Validation** examples covers the SIL, PIL and additional blocks supported by toolbox to help faster prototyping;
- Applications covers the AMMCLIB and Motor Control scenarios;

The examples are also available from standard MATLAB Help for NXP's Model-Based Design Toolbox Example

🚳 Help		- 🗆 X				
📥 🍓 🔆 - 🞯 📔 🛛 Custom Documenta	ion 🗶 🕂	🖽 🗆 🖯 🗖 🎽				
Documentation	Search Help	٩ 🔒				
< Documentation Home Examples (Supplemental Software) S32K14x Generic Simulink Models	EXAMPLES NXP's Model-Based Design Toolbox for S32K1xx Automotive Microprocessors Family Website					
<ul> <li>S32K14x Targeted Simulink Models Examples</li> </ul>	S32K14x Generic Simulink Models					
Supplemental Software	Core & Systems					
	PMC GPIO	ởs Model				
	PMC Callbacks	🛐 Model				
	PMC sleep on exit	🋐 Model				
	PMC very low power run	Model				
	RTC	🌆 Model				
	LPTMR Time Counter	Model				
	LPTMR Pulse Counter	5 Model				
	LPTMR Complex Time Counter	🋐 Model				
	CSEC	🋐 Model				
	EDMA Scatter Gather Transfer	Model				
	EDMA Simple Single Block Transfer	Pa Model				

#### 2.2 Hardware Setup

All examples provided with the Model-Based Design Toolbox were developed on S32K144EVB-Q100 as primary target. Additional information about this development kit can be found on NXP official web page <u>here</u>.



Before running any example on the S32K144EVB-Q100 a proper communication setup between the board and the host PC must be enabled. Please follow the next steps to ensure a working setup:

- 1. Place J107 jumper on positions 2-3 to power the S32K144EVB via the USB micro connector;
- 2. Place J104 jumper on positions 2-3 to enable the S12K144 MCU reset;
- 3. Connect the EVB micro USB connector J7 to a host PC USB connector using supplied USB cable;
- 4. Allow the PC to automatically configure the USB drivers if needed. Windows OS should automatically detect the S32K144 EVB and should assigned a virtual COM port and a virtual mass storage device to the host;
- 5. When powered through USB, LEDs D2 and D3 should light green;
- 6. Once the board is recognized, it should appear as a mass storage device in your PC with the name EVB-S32K144;



#### 2.3 Hello World Example

If the hardware setup is completed successful and a virtual COM port is created and visible in Control Panel -> Device Manager -> Port (COM & LPT) and a virtual mass storage device is present, then all ingredients for running a Model-Based Design Toolbox for S32K1xx Example.

Open the Examples Library and go to Communications block and double click on UART Hello World block.



The Simulink lpuart\_hello\_world\_s32k14x.mdl model will open. This model programs the S32K144 MCU to sent a message over the UART.



Follow the next steps to run the example:

1. Double click on MBD\_S32K14x\_Config\_Information block and setup the Download Config parameters according with your PC and HW setup

皆 Block Parameters: N	MBD_S32K14	x_Config_Informatior			×			
MBDTBX_EC_S32K14 (mask) (link)								
Model-Based Design Toolbox Config block for \$32K14x family of processors.								
MCU Build Tool	chain Ta	arget Connection	Diagnostics					
Processor-in-the	-Loop (PIL)	Mode Download						
	after Build							
Download settings								
Delay before start	of applicatio	on (ms) 5000000			<u> </u>			
Download Interface	е	OpenSDA			•			
Boot Assist Mod	ule (BAM) R	Restart Request						
Serial								
COM Port	Custom	v	Refr	esh				
Custom COM Port	COM1							
Baud Rate	57600				~			
OpenSDA								
OpenSDA Drive Na	me	D: EVB-S32K1 -	A	Refresh				
		OK	Cancel	Help	Apply			

2. Apply and close this windows.

3. Press Build Model button and wait until the code is generated, compiled and downloaded to the evaluation board.



- 4. Open any UART terminal for the virtual COM port assinged and set up the baud rate at 115200, data bits 8 and parity none.
- 5. Press the reset button on the evaluation board.
- 6. The S32K144 MCU sends "Hello World!!! Press RESET to see this msg again" message over the UART and the UART terminal should display it.



Congradulations! You succedded with running your first example created with Model-Based Design Toolbox for S32K1xx 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, FlexIm, 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

©2018 NXP Semiconductors. All rights reserved.

