

STM32-MAT/TARGET

Hands On

Rev 1.0



Objectives 2

- Hands-on workshop to show you the steps needed to quickly simulate and develop STM32 graphical applications using MATLAB® Simulink environment.
- Know tools installations and settings to be able to start development.
- Know Simulation and « C » Code Generation possibility
- Know how to develop application from scratch
- Know where to obtain additional technical support



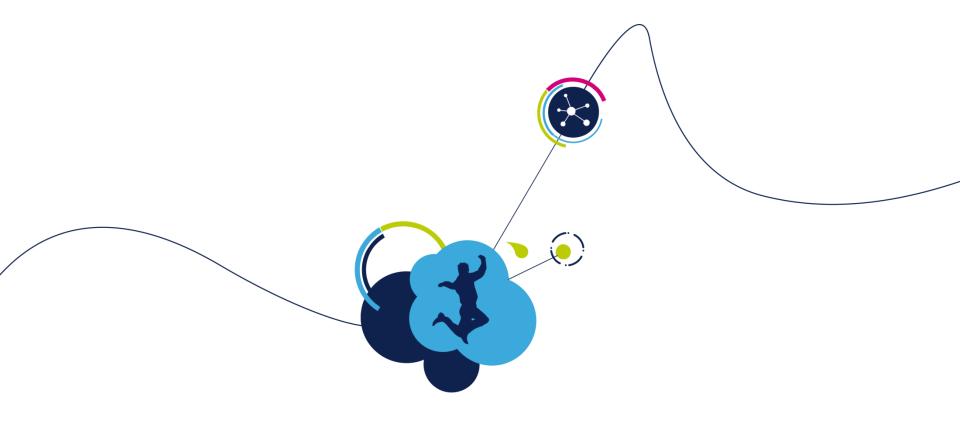
Systems Check 3

Mandatory Software

- From Mathworks
 - MATLAB®, Simulink and Embedded Code
- From STMicroelectronics
 - <u>STM32CubeMX</u>
- One of following Toolchain
 - EWARM from IAR 💥
 - MDK-ARM from Keil 🐺
 - <u>TrueSTUDIO</u> from Atollic 🪍
 - <u>SW4STM32</u> from STMicroelectronics
- STM32-MAT/TARGET toolkit to develop STM32 applications

Hardware

- Any electronic application board with STM32 and SWD/JTAG connection.
- STLinkV2 or 3rd parties dongle if not integrated to STM32 application board.
- USB to Serial adapter.



Hardware setup



Step #1 – Hardware selection 5

- Use one of STM32 boards including STLinkV2
 - Nucleo, Discovery, EvaluationBoard etc...
 - STM32F3348-DISCO and STM32F429i-DISCO will be used during examples.
- Or STM32 application board connected to SWD (Single) Wire Debug)/JTAG dongle
 - STLinkV2, ULink2, Jlink etc..
- For PIL (Processor In the Loop) communication.
 - USB to Serial adapter is needed or board integrating VCP(Virtual COM Port)











Step #2 – Hardware connection

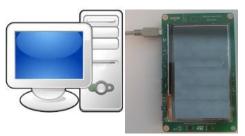
Connect USB dongle port to PC USB port



And connect STM32 HE10 20 pins dongle connector to STM32 target board



Or connect PC USB port to embedded STLinkV2





Step #3 – Hardware connection

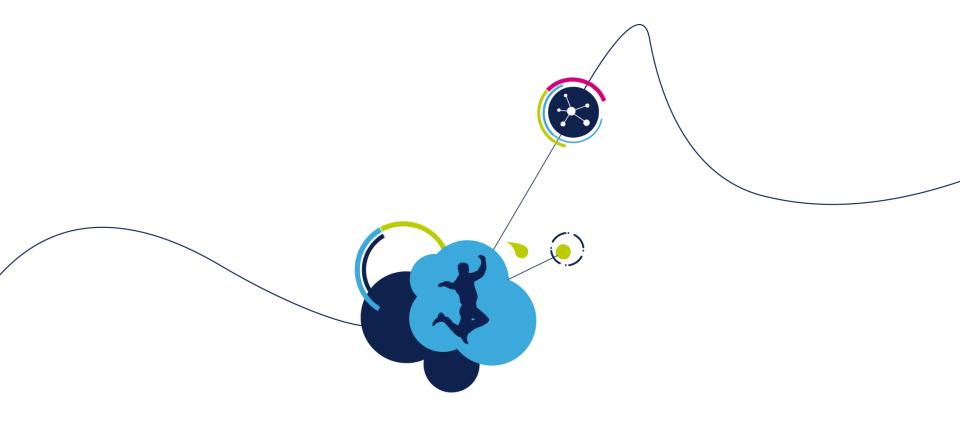
As soon as you are using ST-LINK/V2

- look at http://www.st.com/web/catalog/tools/FM146/CL1984/SC720/SS1450/PF251168?s_ searchtype=partnumber
- « Related Tools and Software » section to check or update firmware

Related Tools and Software

Related Tools and Software		
Part Number	Description	
STSW-LINK004	STM32 ST-LINK utility	
STSW-LINK005	ST-LINK/V2 firmware upgrade	
STSW-LINK009	ST-Link, ST-Link/V2, ST-Link/V2-1 USB driver signed for XP, Windows7, Windows8	

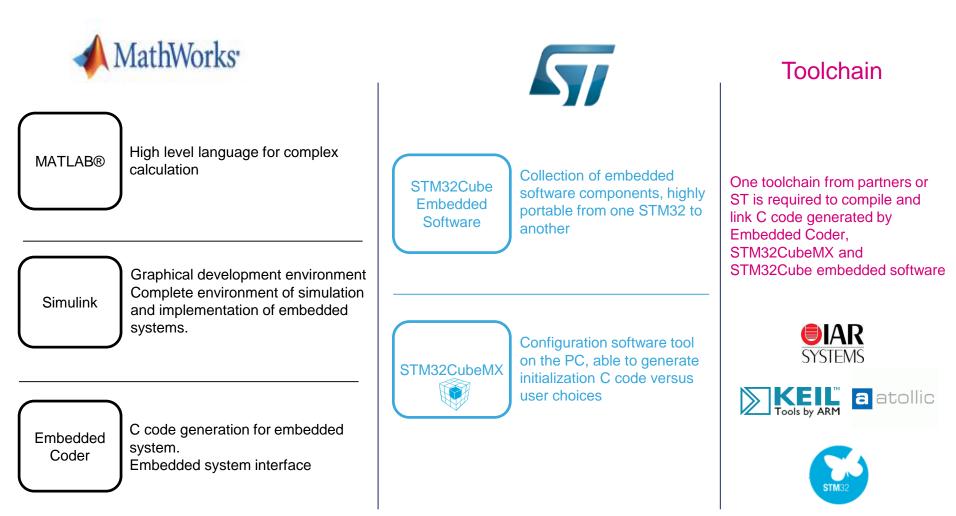




Software setup



Quick description of tools





Step #1 – Software installation 10

- Install MathWorks software (R2015b or later)
 - MATLAB®, Simulink, Embedded Coder are mandatory
 - Add-ons : SimPowerSystems, Simscape and Staflow for motor control applications if needed.
 - <u>http://www.mathworks.com</u>
- Install STM32CubeMX
 - Download and documents available from : <u>www.st.com/microxplorer</u>
- Install toolchain (Cf Slide 3 : « Systems Check »)
 - Cf Slide 3 « Systems Check » to get link to supported 3rd parties download area.



Step #2 – Software installation 11

Install STM32-MAT/TARGET

- STM32 embedded target for MATLAB® and Simulink
- This toolkit is mandatory to integrate STM32 to MathWorks tools.
- Download and documents available from : click here
- Unzip « stm32-mat_target.zip »
- Run «STM32MatTarget_xx_setup.exe » (xx = software version)
- Default installation path is « C:\MATLAB\STM32-MAT » repository





Step #3 – STM32-MAT/TARGET integration 12

- Integrate STM32-MAT/TARGET to MathWorks flow
 - 1. Open MATLAB®
 - 2. Enter « pathtool » command from CommandWindow

MATLAB R2015b		11 - Marca
HOME PLOTS	APPS	
New New Open Compare Script FillE	Import Save	Analyze Code
	AB + STM32-MAT + STM32 +	
Current Folder Name ▲ addSrc blks CRL CRL blks CRL pil pil Frtw STM32demos MCD-ST Liberty SW License Ag Readme.html Startup.bat	Command Window fx >> pathtool freement V2.pdf	



Step #4 – STM32-MAT/TARGET integration 13

- 3. Click « Add with Subfolders... » button on « Set path » window.
- 4. Select STM32 installed path
 - Default path :C:\MATLAB\STM32-MAT\STM32

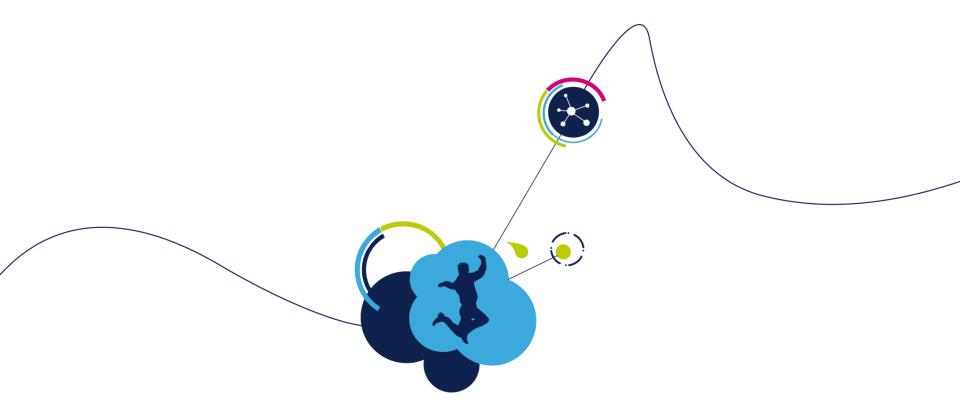
🛛 🍑 MATLAB	*	Name
a 퉲 STM32-MAT		addSrc
a 🌗 STM32		blks
addSrc		

5. Click « Save » then « Close » button.

📣 Set Path	X 0 -
All changes take effect immedia	tely.
	MATLAB search path:
Add Folder	C:\Program Files\MATLAB\R2014b\toolbox\matlab\testframework
Add with Subfolders	C:\Program Files\MATLAB\R2014b\toolbox\matlab\demos
Add with Subfolders	C:\Program Files\MATLAB\R2014b\toolbox\matlab\graph2d
	C:\Program Files\MATLAB\R2014b\toolbox\matlab\graph3d
Move to Top	C:\Program Files\MATLAB\R2014b\toolbox\matlab\graphics
Move Up	C:\Program Files\MATLAB\R2014b\toolbox\matlab\graphics\obsolete
more op	C:\Program Files\MATLAB\R2014b\toolbox\matlab\plottools
Move Down	C:\Program Files\MATLAB\R2014b\toolbox\matlab\scribe
	C:\Program Files\MATLAB\R2014b\toolbox\matlab\scribe\obsolete
Move to Bottom	C:\Program Files\MATLAB\R2014b\toolbox\matlab\specgraph
	C:\Program Files\MATLAB\R2014b\toolbox\matlab\uitools
Remove	
	Save Close Revert Default Help

6. Simulink is then ready to design STM32 application.

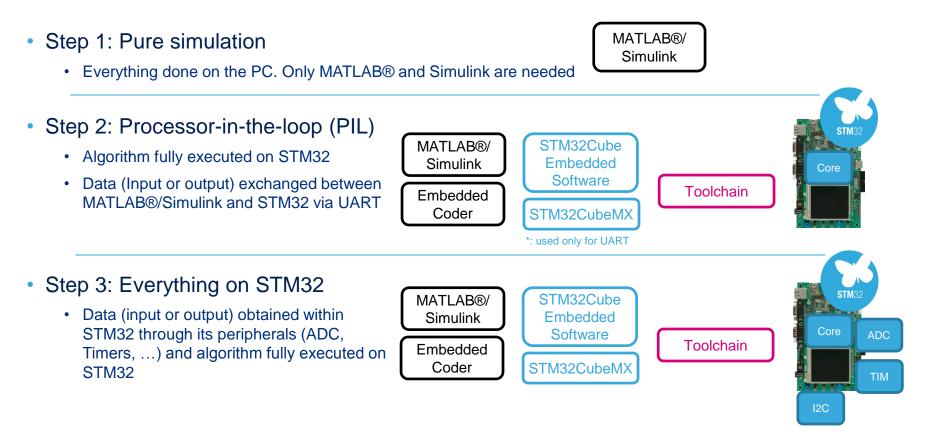




Scenarios and Tools usage



Simulation / PIL / Code Generation _____



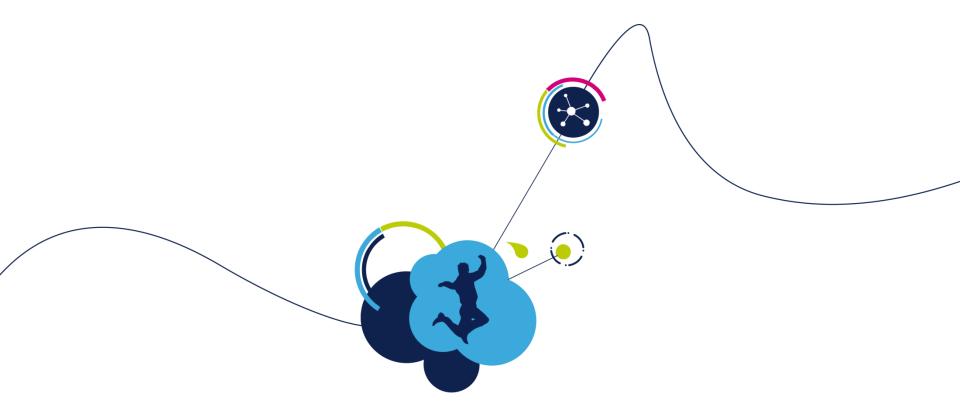
Scenarios are independent and can be done individually



Scenarios 16

- Simulink graphical applications can :
 - Be Simulated on PC.
 - MATLAB® and Simulink needed only.
 - Process Simulink data on STM32 target
 - MATLAB®, Simulink, STM32CubeMX, one toolchain and STM32-MAT/TARGET must have been installed.
 - Generate C code project for this application
 - MATLAB®, Simulink, STM32CubeMX, one toolchain and STM32-MAT/TARGET must have been installed.
 - Simulink model must be created and configured to be able to be able to develop STM32 graphical applications.

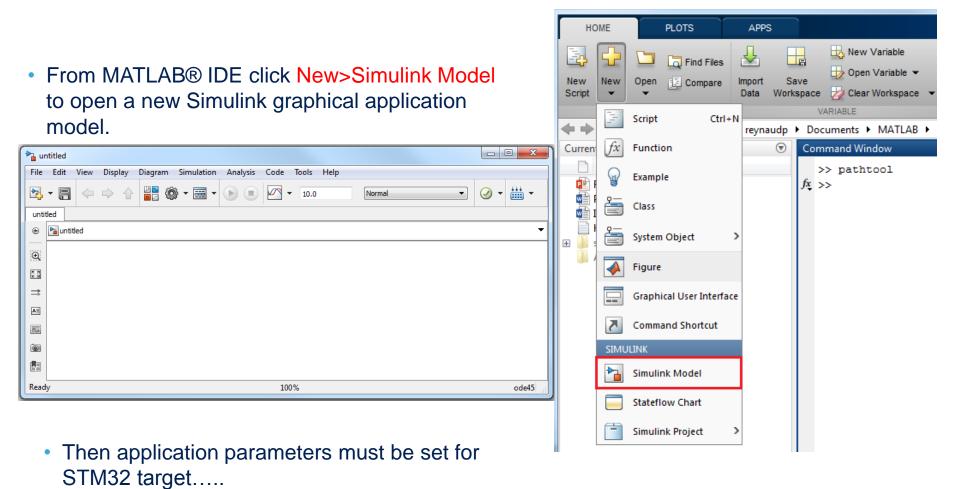




Simulink model configuration



Simulink Model Setting 1/4 18



life.augmented

Simulink Model Setting 2/4 19

• Open Configuration Parameters window and select Code Generation

Pa untitled	 Click « Browse » button 	to select
File Edit View Display Diagram Simul	ation Analy	
🔁 • 🚍 💠 🔶 🖀 🎯 •	stm32.tcl as System targe	
untitled Sconfiguration Parameters: untitled/Configu	iration (Active)	
Select: Target Target	selection	^
Solver Data Import/Export System	a target file: grt.tlc	Browse
Ontimization	ge: C	•
Diagnostics Hardware Implementation Build p	rocess	
Model Referencing	hain settings	
Code Generation	hain: Automatically locate an installed toolchain	▼ Validate
Comments 1000	Microsoft Windows SDK v7.1 nmake (64-bit Windows)	Validate
Symbols Custom Code Build		✓ Show settings
Debug Interface		
Interface	System Target File: Description:	
Code	autosar.tlc AUTOSAR Generativeert.tlc Embedded Coder	
Selec	t objectiv ert.tlc Create Visual C/C++ Solu	
Chec	k model b ert_shrlib.tlc Embedded Coder (host-bas	Check Model
	grt.tlc Generic Real-Time Target	
E Ger	erate cod idelink ert.tlc IDE Link ERT E	Build
Pac	kage code idelink_grt.tlc IDE Link GRT	
	realtime.tlc Run on Target Hardware rsim.tlc Rapid Simulation Target	
	rtwsfcn.tlc S-Function Target	Help Apply
	stm32.tlc stm32 (Embedded Target)	
 Click « OK» buttor 	Full Name: C:\snap_view\mcd_cd\Matlab\STM32\rtw\stm32.tlc	
	<u>OK</u> <u>C</u> ancel <u>H</u> elp <u>A</u> pply	



Simulink Model Setting 3/4 20

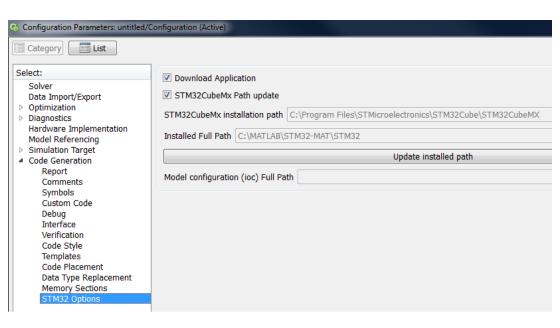
- stm32.tlc has been selected, Build process parameters has changed
 - Select STM32 Options

Configuration Parameters: untitle	ed/Configuration (Active)	and the	or Parcel	
Select: Solver Data Import/Export Optimization Diagnostics Hardware Implementation Model Referencing Simulation Target Code Generation Pagenet	Target selection System target file: stm32.tlc Language: C Description: stm32 (Embe Build process Makefile configuration	edded Target)		Browse
Report Comments Symbols Custom Code Debug Interface Verification	 ✓ Generate makefile Make command: Template makefile: 	make_rtw stm32.tmf		
Code Style Templates Code Placement Data Type Replacement Memory Sections STM32 Options	Data specification override Ignore custom storage classe Code Generation Advisor Prioritized objectives: Unspeci	fied	Ignore test point signals	Set Objectives

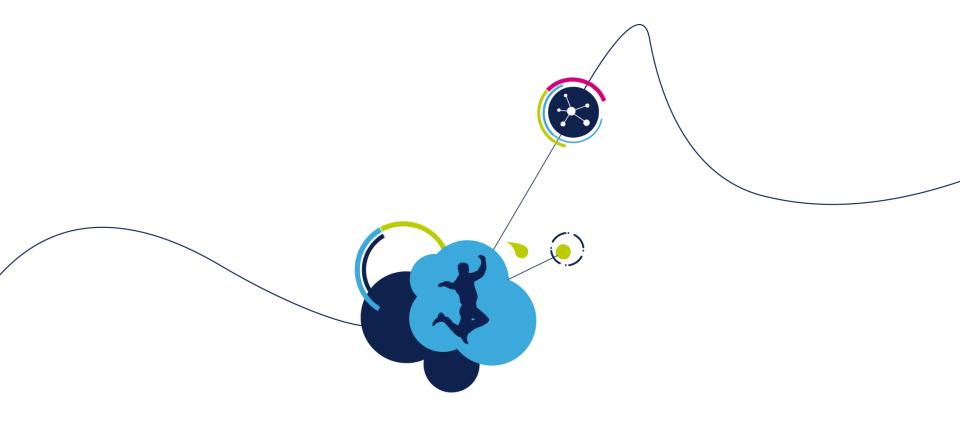


Simulink Model Setting 4/4

- STM32CubeMX Path update is used to automatically update STM32CubeMX installation path.
 - Path selected during STM32CubeMX installation
- Download Application is used to start STM32CubeMX to generate project.
 - Uncheck for code generation only
- Installed Full path is STM32-MAT/TARGET installation path.
 - Default path : C:\MATLAB\STM32-MAT\STM32
- Update installed path to update path when it has changed
- Model configuration (ioc) Full path is hardware configuration file path created using STM32CubeMX for this Simulink application
 - Read only, updated from STM32_Config model.





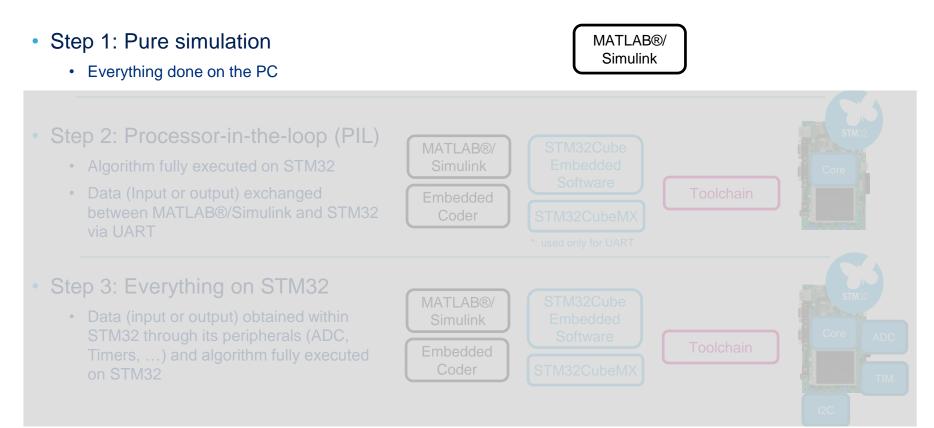


PC pure simulation



Tools usage

23

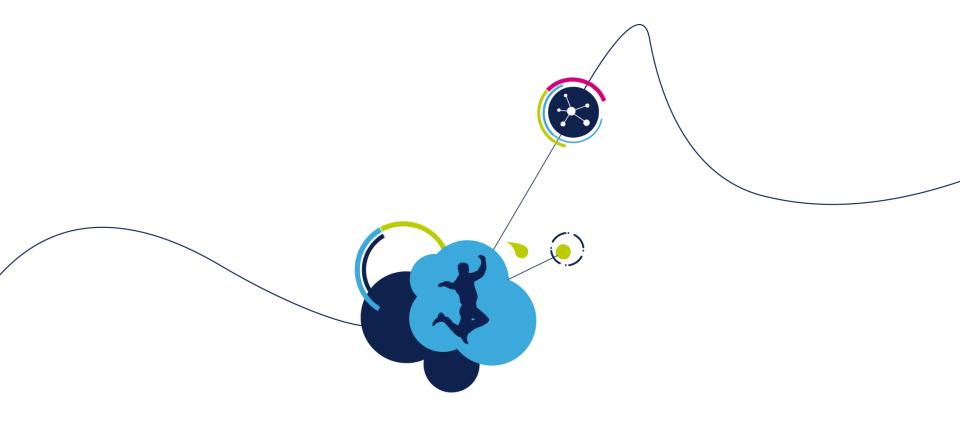




MATLAB® & Simulink 24

- Not a subject for this HandsOn
- General purpose for MATLAB®/Simulink
- Doesn't need STM32-MAT/TARGET toolkit.
- Contact MathWorks for MATLAB®, Simulink trainings



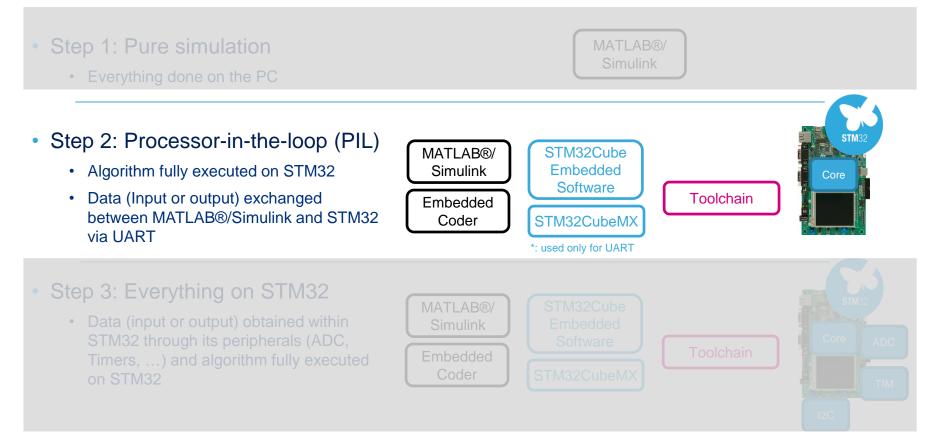


Processor in The Loop (PIL)



Tools usage

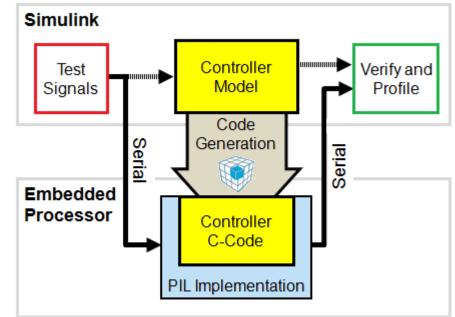
26





PIL Overview 27

- PIL is used to run MATLAB® algorithm on STM32
 - To see if STM32's computational results are numerically equivalent to PC-based simulation results.
 - To measure directly the time to execute the model steps (profiling)
- PIL simulation uses serial port to send data from Simulink to STM32 and receive Simulink processed data back from STM32
- PIL simulation doesn't run in « real-time »
 - Simulink is « master » sending asynchronously data to STM32 through serial port.
 - STM32 waits for data from Simulink, process data (in real-time) and send it back to Simulink through serial-port.
- PIL simulation doesn't process real data from STM32 peripherals.
 - STM32 peripherals (ADC etc...) are not used.
 - Only STM32 USART peripheral is used to communicate with Simulink.

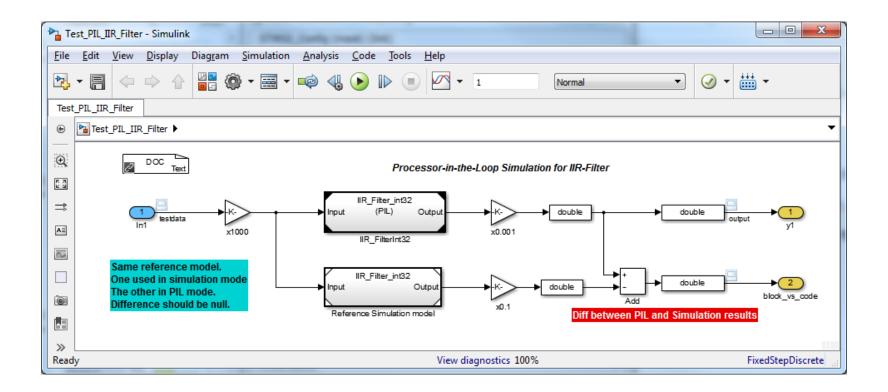




PIL Example 28

• IIR Filter example

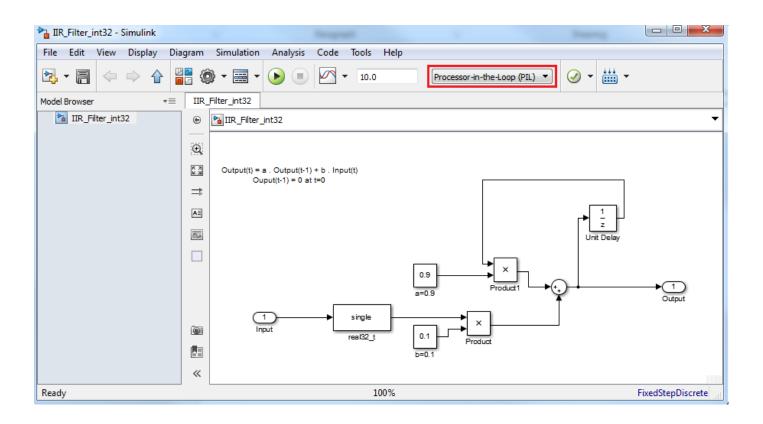
 Open Test_PIL_IIR_Filter.mdl model example from C:\MATLAB\STM32-MAT\STM32\STM32demos\PIL\Filter (Default STM32-MAT/TARGET installation path)





Reference Model for PIL 29

- It uses IIR_Filter_int32.mdl as reference model.
- IIR_Filter_int32.mdl is a simple algorithm for one order filtering.
- IIR_Filter_int32.mdl is set to be used for Processor-in-the-loop (PIL)





PIL setting vs Simulation 30

Function Nock Parameters: DR_FilterInt32

• IIR filter int32 model is used twice :

life.augmented

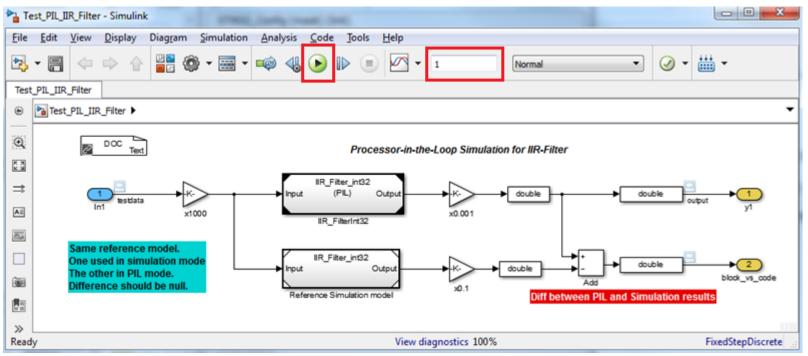
		Open		Model Reference
		open		Reference the specified model. If the referenced model requires any model arguments, enter them as a comma separated list.
	*	Cut	Ctrl+X	If the referenced model has more than one instance simulating in Normal
	-			mode, you might need to turn on Normal Mode Visibility for this Model block. Normal Mode visibility can be controlled by going to the top model
	ĥ	Сору	Ctrl+C	and editing Normal Mode Visibility.
	Ē.	Paste	Ctrl+V	Parameters Model name:
		Comment Through	Ctrl+Shift+Y	IIR_Hiter_Ints/ Browse Open Model
		2		Model arguments:
One is set for PIL		Comment Out	Ctrl+Shift+X	
→ Input (PIL) Output		Delete		Model argument values (for this instance):
				Simulation mode: Processor-in-the-loop (PIL)
IIR FilterInt32		Find Referenced Variables		Code interface: Model reference
-				
		Subsystem & Model Reference	•	
		Format		<< Enable variants
			,	OK Cancel Belp Apply
		Rotate & Flip	*	
		Arrange	+	Punction Block Parameters Reference Simulation model
IIR Filter int32		-		Model Reference Reference the specified model. If the referenced model requires any model
► Input Output		Mask		arguments, enter them as a commo separated list.
		Library Link		If the referenced model has more than one instance simulating in Normal mode, you might need to turn on Normal Mode Visibility for this Model
Reference Simulation model				block. Normal Mode visibility can be controlled by going to the top model and editing flormal Mode Visibility.
		Signals & Ports	+	Porometera
 One is set for simulation model (Normal) 		-		Model name:
		Requirements Traceability		IR_Fiter_int32.md(Browse Open Model
				Model arguments:
 Click right mouse button on each 		Fixed-Point Tool		Model argument values (for this instance):
•				Model argument values (for this instance):
reference model to open select Block		C/C++ Code	*	Simulation mode: Normal
Parameters		Dia de Danamatana (MandalD. (
		Block Parameters (ModelReference)		
		Properties		<< Enable variants
		Help		O QK Cancel Help Appy

Explore

Simulink PIL start

Set simulation duration time and click run simulation green button

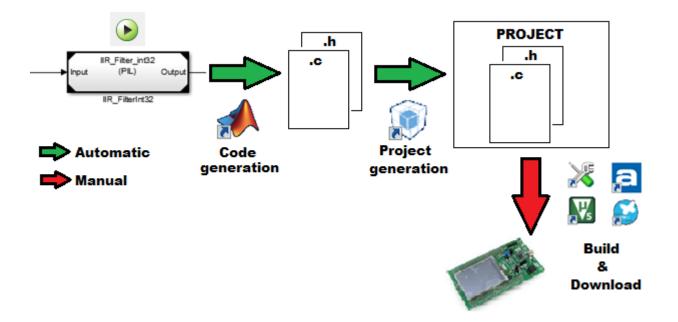
Example : Simulate 1second





Code generation overview 32

- PIL reference model runs into STM32 target as simulation model runs. on PC
 - Automatic convertion of PIL model to .c/.h files
 - Automatic call to STM32CubeMX to create project
 - Manually build and download project to STM32 target from selected toolchain





PC/STM32 communication overview

Some parameters are requested when you start PIL

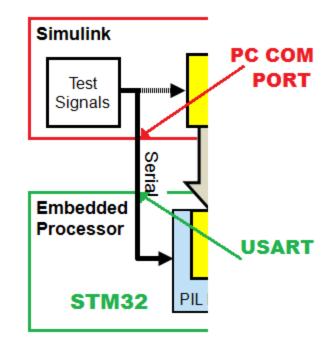
- STM32 Target
 - STM32 device that will run the application

• PC Communication Port

- COMx
 - Default parameters : 115200bds, 8b, no parity, 1 stop

STM32 Communication Port

- USARTx
 - USARTx, Port and Pins for Rx/Tx
 - Same parameters as PC COM Port selection





PIL Processing 1/5

PC COM Port from Device Manager

 8b is fixed, other paramet changed

Look at Windows «Device Manager» to find PC COM Port

 Time Out: Simulink mess seconds without commun

ieters can t)e	Simulink	
ssage error unication.	after 10	Test Signals	PC COM PORT
Device Manager Ports (COM & LPT	- · · ·) -Serial Comm Port COM6		Set parameters
Figure 1: PC COM Port se			
Set PC COM	Port for PIL settings and pro	ess continue.	
сом	Parity	none 💌	
Bits per second 115200	▼ Stop bits	1	
Data bits ⁸ (8)	Flow control	none 💌	
	Time out (s) 10		
	Continue		



PIL Processing 2/5 35

STM32 Selection

Example with STM32F429i-DISCO board

- Family : STM32F4
- Name: STM32F429ZITx
- Frequency:168MHz (by default STM32max speed is selected)

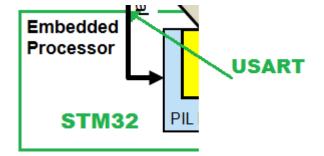
	Figure 1: STM32 MCU selection	
	Select MCU Family and	MCU name and press continue.
Frequency value is used for Profiling. To compute STM32 processing time.	STM32 Family	STM32 Name STM32F429ZITx 💌
		quency value (Hz) 168000000
		Continue



PIL Processing 3/5

STM32 COM Port

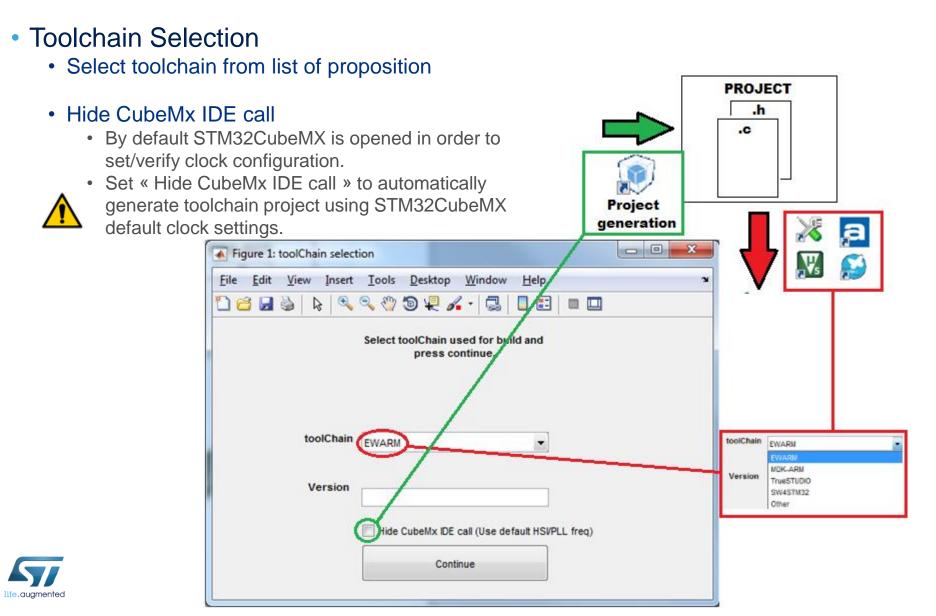
- Example
 - USART1
 - Rx: PA10 & Tx: PA9



承 Figure 1: PIL USART parameters se	election			
Select PIL MCU communication parameters and press continue.				
	Rx port	PA10 💌		
Usart name USART1 -	Tx port	PA9 V		
	Continue			



PIL Processing 4/5



PIL Processing 5/5

- It is an asynchronous process.
 - Simulink is waiting.
 - Simulink must send data through COM port, only when project is built and downloaded to STM32 target.
 - Then, you will press « Continue » button to start data flow PC/STM32.
 - STM32CubeMX is automatically opened and you can verify or modify STM32 settings.

	Figure 1: PIL start	
	<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>I</u> nsert <u>T</u> ools <u>D</u> esktop <u>W</u> indow <u>H</u> elp	צי
3	🗋 😂 🛃 🎍 🗞 🔍 🤍 🥙 🕲 🐙 🖌 - 🛃 🔲 🖽 🔲 🛄	
	Configure MCU. Generate, buid and start project. Press Continue to start PIL.	
	Continue	



STM32CubeMX & PIL 1/2 39

STM32CubeMX settings done

- Pinout :
 - Usart1, Rx/Tx
- USART1 Configuration

Basic Parameters	
Baud Rate	115200 Bits/s
Word Length	8 Bits (including Parity)
Parity	None
Stop Bits	1



- Clock Configuration
 - Default value



Reference value already selected for Profiling

Max frequency value (Hz)	
168000000	

- Clock setting : Enter 168 and return
 - · Then Clock tree is automically updated







STM32CubeMX & PIL 2/2 40

STM32CubeMX project generation

- Generate source code based on user settings
 - Press Project Settings « OK »
- Generate Project

STM32CubeMX IIR_Filter_int32.ioc*: STM32F429ZITx										
File Project Pinout Window Help										
🕼 📂 📗 🖶 堤 🚺 🎒 🖉 🕼 Keep Current Signals Placement 🤘										
Pinout Clock Configuration Configuration Power Consumption Calculator										

Generating EWARM project								

• Open Project

Code Generation	21	
The Code is successfu	ully generated under C:/snap_view/mcd_cd/M	latlab/STM32/STM32demos/CodeGeneration/IIR_Filter_int32_PIL
	Open Folder Open Project	

Close STM32CubeMX





Toolchain & PIL 1/2 41

- Toolchain project
 - Generated from STM32CubeMX
 - Includes
 - Application files generated from MATLAB®
 - main.c generated from MATLAB® and modified by STM32CubeMX
 - HAL mandatory peripherals drivers

Files	
IIR_Filter_int32 - IIR_Filter_int32 Configuration	
- I EWARM	Generated
🗕 🗠 startup_stm32f429xx.s	from
🖵 📮 🗀 User	MATLAB®
- 🕀 🗊 main.c	
-⊞ [c] stm32t4∞_hal_msp.c	and
L = Image: Stm32f4xx_it.c	Modified
	by
	STM32CubeMx
L → ⊕ 🔁 system_stm32f4∞.c	ormozousciix
└─── STM32F4xx_HAL_Driver	
-⊞ 🔁 stm32f4xx_hal.c	
- 🖽 🚺 stm32f4xx_hal_cortex.c	
-⊞ 🗈 stm32f4∞_hal_dma.c	
- 🖽 💽 stm32f4xx_hal_dma_ex.c	
-⊞ 🔁 stm32f4∞_hal_flash.c	
- ⊞ 🔁 stm32f4xx_hal_flash_ex.c	
- stm32f4∞_hal_flash_ramfunc.c	
-⊞ 🗈 stm32f4∞_hal_gpio.c	
- ⊞ 💽 stm32f4xx_hal_pwr.c	
- ⊞ 🛅 stm32f4∞_hal_pwr_ex.c	
- ⊞ 🔁 stm32f4xx_hal_rcc.c	
-⊞ 🔁 stm32f4xx_hal_rcc_ex.c	
└─⊞ 🔂 stm32f4xx_hal_uart.c	
- 🛱 🧀 Middlewares	
- Codeinstr_data_stream.c	
- IIR_Filter_int32.c	
Hereit in the stream_utils.c	
Here STM32SerialRtiostream-ProfilerTimer.c	Generated
- 🖽 🛅 STM32SerialRtiostream-Rtiostream.c	from
Hereit Stream.c	MATLAB®
Here and the structure that the	MAILAD®
Hereit State and	
Here interface_lib.c	
└─⊞ 🔂 xilcomms_rtiostream.c	
Lep 🗀 Output	
IIR_Filter_int32 Configuration.out	



Toolchain & PIL 2/2



Add Group		
Import File List		
Add Project Connection		
Edit Configurations		
Remove		
Create New Project		
Add Existing Project		
Options	ALT+F7	
Version Control System	÷.	
Make	F7	

Project Tools Window Help

Add Files...









Run project

• Then, STM32 is running and waits for data from Simulink.



Running PIL 43

_ 🗆 🗙

🗼 Figure 1: PIL start

Eile Edit View Insert Iools Desktop Window Help

Configure MCU. Generate, buid and start project. Press Continue to start PIL.

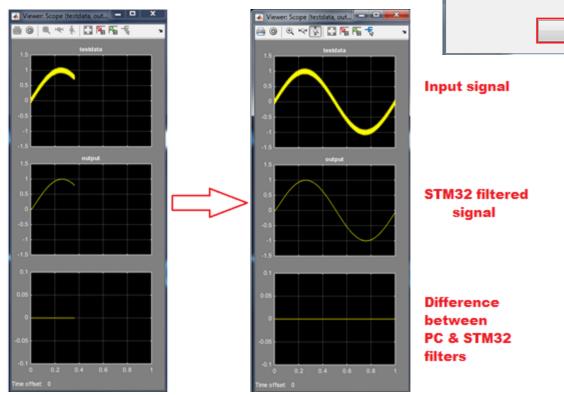
Continue



« Continue » button of PIL start window

PIL results

life.augmented



PIL Profiling 1/3 44

- Profiling = PIL Timing analysis
 - executionProfile has been generated in MATLAB® Workspace
 - « executionProfile » default Workspace variable name
 - Can be modified from « Configuration Parameters » window Code Generation > Verification of the model.

Workspace						
Name 📥	Value					
🕂 a1	0.9000 0.1000					
ь0						
🕫 executionProfile	1x1 ExecutionTime					
t	1001x1 double 1001x1 double 1001x1 double					
🛨 testdata						
🛨 tout_PIL						
🛨 yout_PIL	1001x2 double					

🗞 Configuration Parameters: Test_PIL_IIR_Filter/Configuration (Active)									
Select:	Code profiling								
Solver	Measure task execution time								
Data Import/Export Optimization 	Measure function execution times								
 Diagnostics Hardware Implementation Model Referencing 	Workspace variable: executionProfile								
 Simulation Target 	Code coverage for SIL or PIL								
 Code Generation Report Comments 	Code coverage tool: None								
Symbols	SIL or PIL verification block								
Custom Code Debug Interface	Create block: None								
Verification	Enable portable word sizes								



PIL Profiling 2/3 45

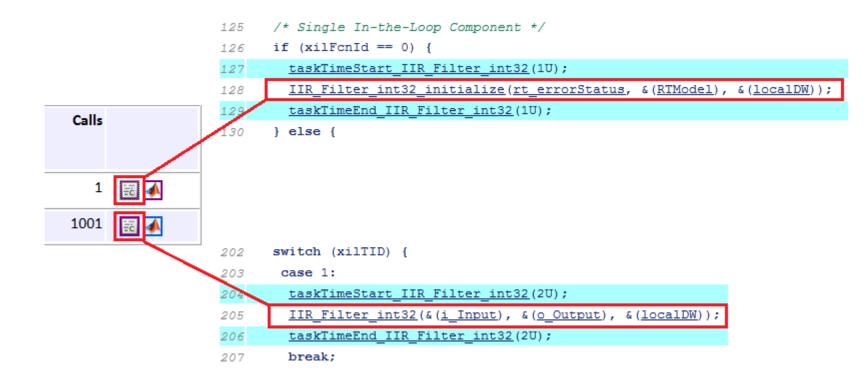
- Enter «executionProfile.report» on MATLAB® command window to open profiling report window to know
 - STM32 processing time at selected frequency
 - Execution time and number of calls per function

Code Execution Prof	iling Report	for IIR_	Filter_int	132					
The code execution profiling report provi rom data recorded by instrumentation p Code Execution Profiling for more inform	robes added to the SIL o								
the excellent round for more more			STM3	2 processing					
1. Summary		time ~549µs							
Total time (seconds × 1e-09)		548905							
Measured time display options		('Units', 'Seconds', 'ScaleFactor', '1e-09', 'NumericFormat', '%0.0f')							
Timer frequency (ticks per second)		1.68e+08							
		01-Feb-2016 14:59:19							
Profiling data created		01-Feb-2	016 14:59:19						
Profiling data created 2. Profiled Sections of Code Section	Maximum Execution Time		016 14:59:19 Maximum Self Time	Average Self Time	Calls				
2. Profiled Sections of Code	Execution	Average Execution	Maximum Self		Calls 1	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1			

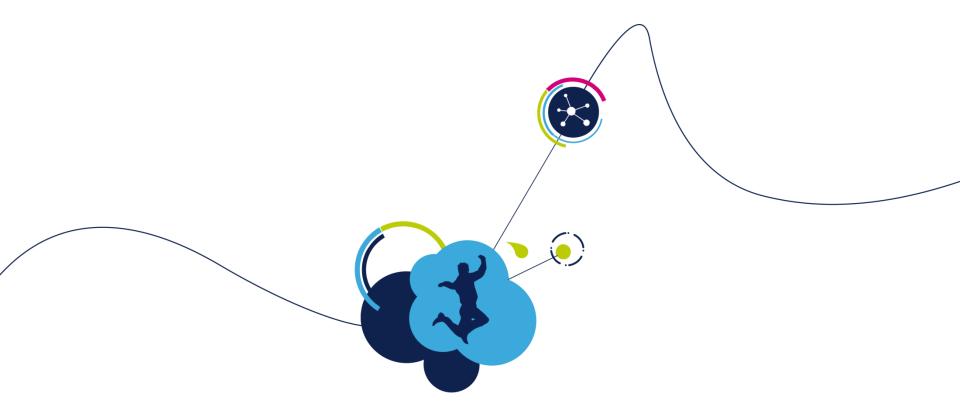


PIL Profiling 3/3 46

- Click on C code generated
 - To see MATLAB® generated code for STM32
 - or profiled sections





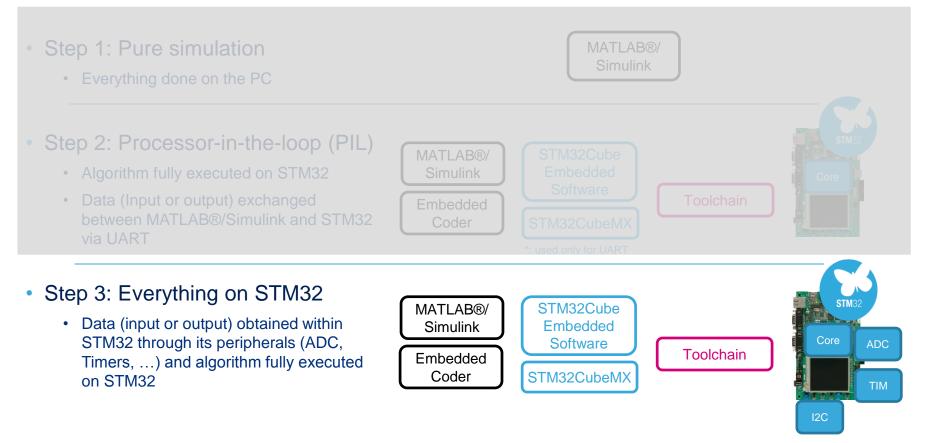


Application Code Generation



Tools usage

48





Simulink application development 49

- Look at Simulink model Setting 1/4 to 4/4 to open and configure new model.
- Save Simulink model and open Library Browser
 - For example: Save model as test.slx into C:\TEMP\test repository



te:	st - Sim	ulink		_	-				_									x
File	Edit	View	Display	Diagrar	m Si	mulation	Ana	lysis	Code	Tools	Help							
▶	• 🗐	\Leftrightarrow	⇒ 1		÷ ۞	• 📰 •	¢					1	10.0		»		Ē	.
test																		
۲	🍋 test																	•
Q																		
5 7 2 9																		
⇒																		
ΑΞ																		
~																		
١ <u>0</u>																		
Read	/									100%					Fixe	edStepD	iscre	te 🔡

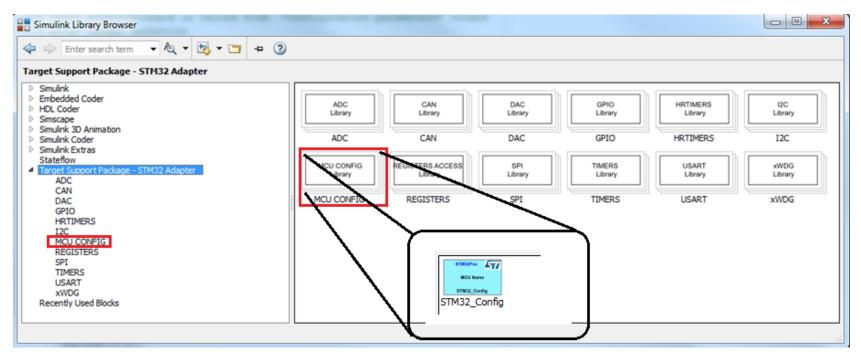


STM32 Configuration 1/3 50

- STM32 Drivers Library
 - Several STM32 peripheral drivers are available.

MCU Configuration

- « MCU CONFIG » is the first library to open and use
- Drag&Drop STM32_Config to your model.
- STM32_Config is used to select STM32 configuration through STM32CubeMX configuration ioc file.





STM32 Configuration 2/3 51

- IOC file selection
 - .ioc file is a text file created and used by STM32CubeMX as STM32 descriptor.
 - .ioc file contains STM32 configuration (pinout, peripheral selection & configuration)
 - Double click STM32_Config to open block parameters window
 - Select ioc file describing STM32 you want to use for your application with « Select STM32 configuration file »
 - It is the only and mandatory way to select .ioc file for Simulink application.
 - You can also modify ioc file or create a new one and STM32CubeMX is automatically opened.

test * - Simulink	Block Parameters: STM32_Config	
File Edit View Display Diagram	STM32_Config (mask) (link) STM32 Configuration : Select STM32 .ioc configuration file or Start STM32CubeMx configuration tool for - Selected ioc configuration file - New ioc configuration file - Browse to select ioc configuration file to modify	- @ - ## - -
⇒ STM32_Config	Parameters STM32 configuration file path Configuration file No selected file Select STM32 configuration file	ioc file selection
	Start STM32CubeMx configuration tool Current ioc file New ioc file Browse for ioc file Start STM32CubeMx configuration tool	ioc file creation or modification
ina ₽= >>	MCU Name MCU Name OK Cancel Help Apply	
Ready	Concer Hep Hppy	FixedStepDiscrete



STM32 Configuration 3/3 52

IOC file creation

- Select « New ioc file » and click « Start STM32CubeMX configuration tool » button.
- STM32CubeMX opens.
- Look at http://www.st.com/web/catalog/tools/FM147/CL1794/SC961/SS1533/PF259242?s_searchtype=partnumber to get STM32CubeMX User Manual.
- Save STM32CubeMX ioc configuration file and select it using « Select STM32 configuration file » button.
 - · ioc file is not automatically selected for Simulink application from STM32CubeMX

	STM32CubeMX Untitled
	File Project Window Help
Block Parameters: STM32_Config	□ = h = 4
STM32_Config (mask) (link)	
STM32 Configuration : Select STM32 .ioc configuration file or	
Start STM32CubeMx configuration tool for - Selected ioc configuration file - New ioc configuration file	
- Browse to select ioc configuration file to modify	New Project
Parameters	
STM32 configuration file path Configuration file No selected file Select STM32 configuration file	Load Project
Start STM32CubeMx configuration tool Current ioc file New ioc file Start STM32CubeMx configuration tool	Help
Start ST MS2CubeMX configuration tool	
MCU Name	
MCU Name	
QK Cancel Help Apply	



Simulink application example 53

Hardware :

- Example based on STM32F3348-DISCO
- Configuration :
 - Leds (LED3/4/5/6)
 - Push Button (User blue button)
 - USART2 Virtual Com Port (SB14&SB16 soldered)
 - ADC1 •
 - TIM1 & TIM6



Software application :

- Use TIM1 to blink LED3 at 1Hz
- Use TIM6 to blink LED4 at 2Hz
- Use TIM6 to trig ADC1 channels 2&3 conversion
- · Blink Led6 when user push button is pressed
- Send ADC1 channel 3 values on USART2 when user push button is pressed



STM32CubeMX STM32F3348 Pinout

Hardware pinout configuration

- PA0 : GPIO EXTIO
- PA1 : ADC1 IN1
- PA2 : ADC1_IN2
- PB3 : Usart2_Tx
- PB4 : Usart2 Rx
- PB6 to PB9 : GPIO_Output
- Hardware setting
 - USART2 is Asynchronous
 - ADC1 IN2 & IN3 Single-ended

🗄 🔔 🗚	[1
IN1	Disable 👻
····IN2	IN2 Single-ended 🗸 🗸
IN3	IN3 Single-ended 🗸 🗸

TIM1 Channel1 as Ouput Compare No output 6.0

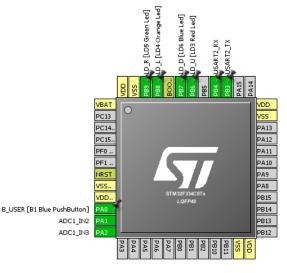
© 11M1				
Slave Mode Dis	sable 👻			
Trigger Source	Disable 👻			
Clock Source	Disable 👻			
Channel1 Output Compare No Output				

• TIM6 Activated (No Output)



USART2 Mode Asynchronous





STM32CubeMX Peripheral settings 1/2 55

Peripheral configuration : •

- USART2
 - Baud Rate : • 115200
 - Word Length: 8 Bits
 - Parity: •
 - Stop Bits: •
 - Enable global interrupt
- ADC1
 - Injected Channels 2&3
 - · Interrupt at end of sequence of conversion

None

1

- Conversion triggered from Timer6
- Interrupt Enabled

ADC1 Configuration

Parameter Settings	火 User Co	onstants	NVIC Settings	V
Interrupt Table		Enabled		
ADC1 and ADC2 interrupts			V	

USART2 Configuration			23
Parameter Settings 🖋 User Constants 🖋 NVIC Settings 🎻	GPIO Settin	gs 🦪 DMA Settings	
Interrupt Table	Enabled	Preemption Priority	Sub Priority
USART2 global interrupt / USART2 wake-up interrupt through EXT line	V	0	0

		L
ADCs_Common_Settings	Independent mode	
Mode	independent mode	
ADC_Settings Clock Prescaler	ADC Asynchronous dock mode	
Resolution	ADC 12-bit resolution	
Data Alignment	Right algoment	
Scan Conversion Mode	Enabled	
Continuous Conversion Mode	Disabled	
Discontinuous Conversion Mode	Disabled	
DMA Continuous Requests	Disabled	
End Of Conversion Selection	End of sequence of conversion	
Overun behaviour	Overun data overwritten	
Low Power Auto Wait	Disabled	
ADC_Regular_ConversionMode		
Enable Regular Conversions	Disable	
ADC_Injected_ConversionMode		
Enable Injected Conversions	Enable	
Number Of Conversions	2	
External Trigger Conversion Edge	 Trigger detection on the rising edge 	
External Trigger Source	Timer 6 Trigger Out event	
Injected Conversion Mode	None	
Queue Injected Context	Disabled	
Rank	1	
Channel	Channel 2	
Samping Time	7.5 Cydes	
Offset Number	No offset	
Injected Offset	0	
Rank	2	
Channel	Channel 3	
Sampling Time	7.5 Cydes	
Offset Number	No offset	
Injected Offset	0	



STM32CubeMX Peripheral settings 2/2

• Peripheral configuration :

- TIM1
 - Default configuration
 - TIM1 Update interrupt enabled

TIM1 Configuration			
Parameter Settings	🖉 User Constants 🗹 N	VIC Settings 🧹 DM	A Settings
Interrupt Table		Enabled	Preemption Priority
TIM1 break and TIM15 int	errupts		0
TIM1 update and TIM16 in	terrupts	v	0
TIM1 trigger and commuta	ation and TIM17 interrupts		0
TIM1 capture compare int	errupt		0

• TIM6

- Trigger event :Update Event
- TIM6 global interrupt enabled

TIM6 Configuration						
Parameter Settings 🖋 User Constants 📢	NVIC Settings	nter de la companya d				
Interrupt Table	Enabled	Preemption				
TIM6 global and DAC1 underrun error interrupts	V	0				

TIM6 Configuration	
🖋 Parameter Settings 🗹 User Constants 🗹 NVIC Set	tings 🦪 DMA Settings
Configure the below parameters :	
 Counter Settings 	
Counter Settings Prescaler (PSC - 16 bits value)	0
	0 Up
Prescaler (PSC - 16 bits value)	Up
Prescaler (PSC - 16 bits value) Counter Mode	Up

GPIO External interrupt

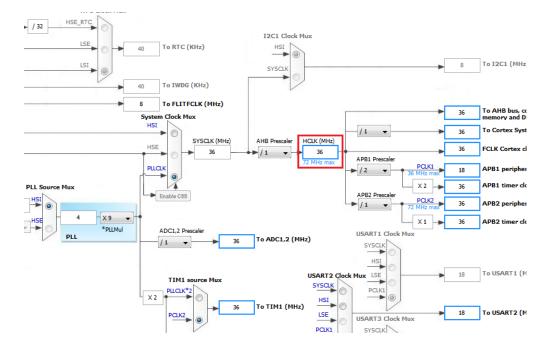
• External Interrupt Mode with Falling edge trigger detection

Pin Configu	ration						×
GPIO ADC1	USART2						
Search Signals Search (Crti+						Show only	Modified Pins
Pin Name	Signal on Pin	GP1O mode	GPIO Pull Up	Maximum out	Fast Mode	User Label	Modified
PAO	n/a	External Interr	No pull up pull	n/a	n/a	8_USER [8181	2
P86	n/a	Output Push Pull	No pull up pull	Low	Disable	LD_U [LD3 Red	3
P87	n/a	Output Push Pull	No pull up pull	Low	Disable	LD_D [LD6 Blu	1
P88	n/a	Output Push Pull	No pull up pull	Low	Disable	LD_L [LD4Ora	1
P89	n/a	Output Push Pull	No pull up pull	Low	Disable	LD_R [LD5 Gre	1
PA0 Configural GP10 mode GP10 Pull Up P			ternal Interrupt Mo	ide with Falling ed	lge trigger detect	tion	•
User Label		_	USER (B1 Blue Push	iButton]			
Group By I	р				App	y Ok	Cancel



STM32CubeMX Clock Configuration 57

- Clock Default Configuration:
 - 16 MHz
- Modification not mandatory
 - Can be 36 MHz for example





STM32CubeMX project Settings

- Project Name:
 - «test» for this example
- Project Location :
 - C:\TEMP for this example
- Save the current project
 - test.ioc file is available from c:\TEMP\test repository

STM32CubeMX test.ioc: STM32F334C8Tx STM32F3348DISCOVERY						
File Project Clock Configuration Window Help						
🗴 🖉 🖉 🕫 🖉 🖉 🖉 🚛 🖬 🖬	P					
Pinout Clock Configuration Configuration Power Consumption Calculato	-					

• You can save ioc file anywhere.



 It is preferable to save ioc file in same repository as model that will use it. Repository and ioc file must have same name.

ject Code Generator	
Project Settings	
Project Name	
test	
Project Location	
C:\TEMP	
Toolchain Folder Location	
C:\TEMP\test\	
Toolchain / IDE	
EWARM	•
EWARM Mcu and Firmware Package Mcu Reference STM32F334C8Tx Firmware Package Name and Version STM32Cube FW_F3 V1.4.0	•
Mcu and Firmware Package Mcu Reference STM32F334C8Tx Firmware Package Name and Version	
Mcu and Firmware Package Mcu Reference STM32F334C8Tx Firmware Package Name and Version	
Mcu and Firmware Package Mcu Reference STM32F334C8Tx Firmware Package Name and Version	



Simulink application IOC file selection 59

IOC file selection

- · loc file has been created and can be selected
- Open (double click) STM32_Config block parameters window
- STM32 configuration path is related to model
 - .\ because test.ioc and test.slx in same repository

Block Parameters: STM32 Config

STM32Fxx	
MCU Name	
STES2_Config	
Double clic	k

-STM32_Config (mas	k) (link)				
STM32 Configuration : Select STM32 .ioc configuration file or Start STM32.CubeMx configuration tool for - Selected ioc configuration file - New ioc configuration file - Browse to select ioc configuration file to modify					
Parameters					
STM32 configuration file path .\					
Configuration file test.ioc Select STM32 configuration file					
Start STM32CubeMx configuration tool					
Current ioc file	New ioc file	Brow	se for ioc file		
Start STM32CubeMx configuration tool					
MCU Name					
STM32F334C8Tx					
_	<u>O</u> K	<u>C</u> ano	cel <u>H</u> elj	<u>Apply</u>	

«Select STM32 configuration file » Browse and select test.ioc file



Simulink application for STM32F334C8Tx MCU



USE TIM1 to Blink LED3 at 1Hz

- Software application example:
 - Use TIM1 to blink LED3 at 1Hz
 - Use TIM6 to blink LED4 at 2Hz
 - Use TIM6 to trig ADC1 channels 2&3 conversion
 - Blink Led6 when user push button is pressed
 - Send ADC1 channel 3 values on USART2 when user push button is pressed



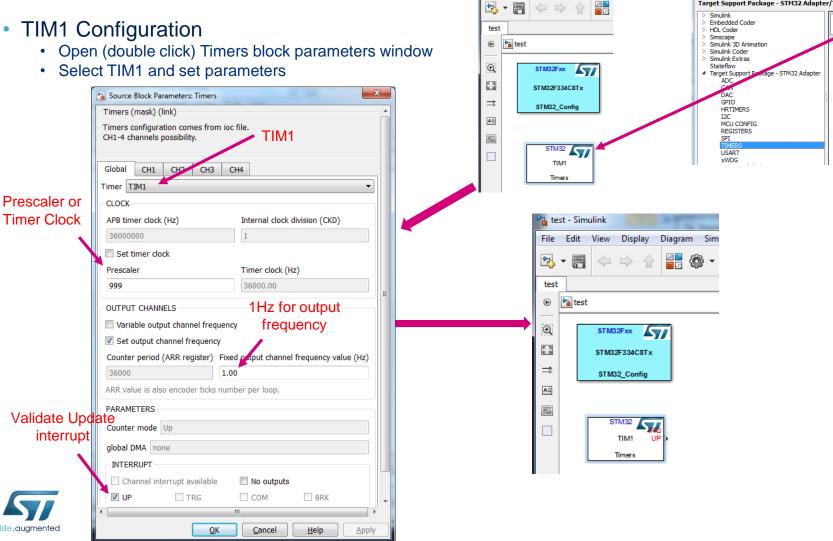
TIM1 Selection & Configuration

File Edit View Display

Diagra

🚹 test - Simulink

- TIM1 Selection
 - Drag&Drop Timers block from Simulink Library Browser
- TIM1 Configuration
 - · Open (double click) Timers block parameters window
 - Select TIM1 and set parameters



57

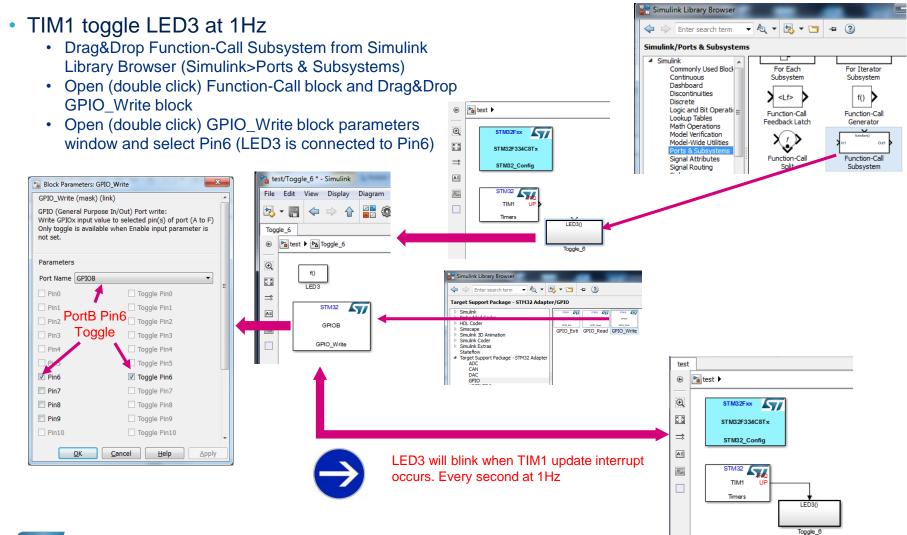
TIM

Timers

Simulink Library Browser

< i>Enter search term 🛛 🔻 🗞 🔻 🛅 中 Target Support Package - STM32 Adapter/TIMERS

TIM1 Application



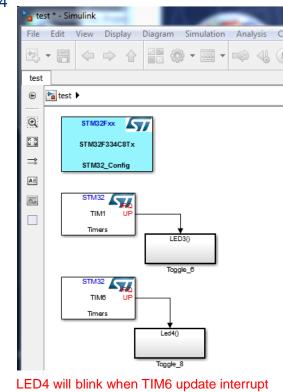
USE TIM6 to Blink LED4 at 2Hz 63

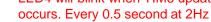
- Software application example:
 - Use TIM1 to blink LED3 at 1Hz
 - Use TIM6 to blink LED4 at 2Hz
 - Use TIM6 to trig ADC1 channels 2&3 conversion
 - Blink Led6 when user push button is pressed
 - Send ADC1 channel 3 values on USART2 when user push button is pressed



TIM6 Application 64

- TIM6 toggle LED4 at 2Hz
 - Make the same thing as for TIM1 but frequency is 2Hz and PortB Pin8 toggle as it is connected to Led4





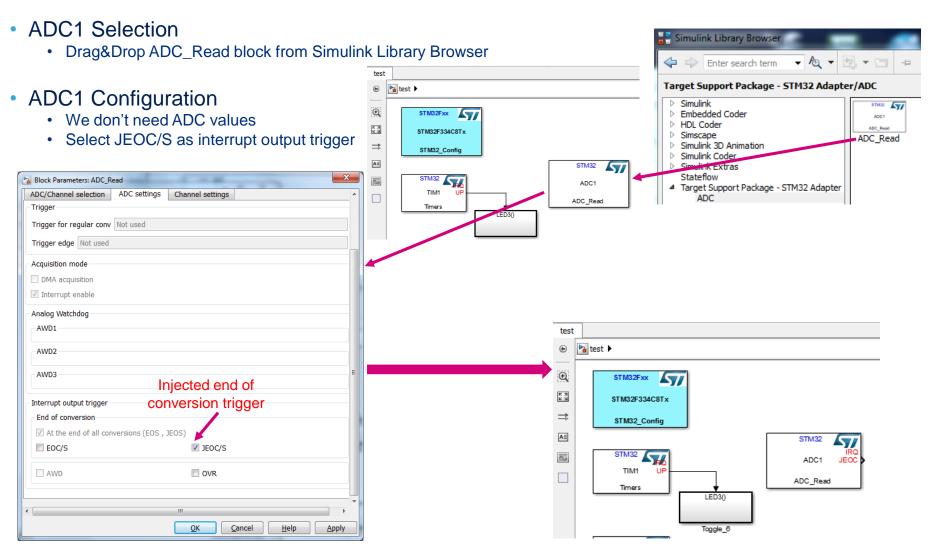


USE TIM6 to trig ADC1 channels 2&3

- Software application example:
 - Use TIM1 to blink LED3 at 1Hz
 - Use TIM6 to blink LED4 at 2Hz
 - Use TIM6 to trig ADC1 channels 2&3 conversion
 - Blink Led6 when user push button is pressed
 - Send ADC1 channel 3 values on USART2 when user push button is pressed



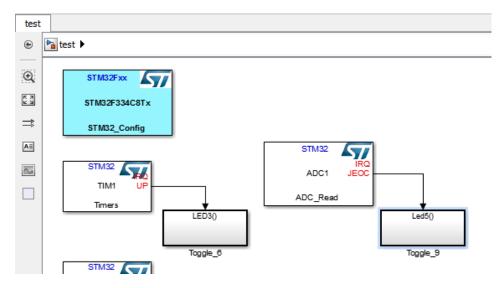
ADC1 Selection & Configuration





ADC1 Application 67

- TIM6 trig ADC1 channels conversion
 - Blink LED5 at end of ADC1 conversion to verify that TIM6 has triggerer it.
 - Drag&Drop Function-Call Subsystem from Simulink Library Browser (Simulink>Ports & Subsystems) and add GPIO Write block in.
 - Set GPIO_Write block parameters window to toggle Pin9 (LED5 is connected to Pin9)



LED5 will blink when ADC1 injected channels 2&3 has been converted.

Start of Conversion is triggered from TIM6

Channels 2&3 values are available at the end of conversion

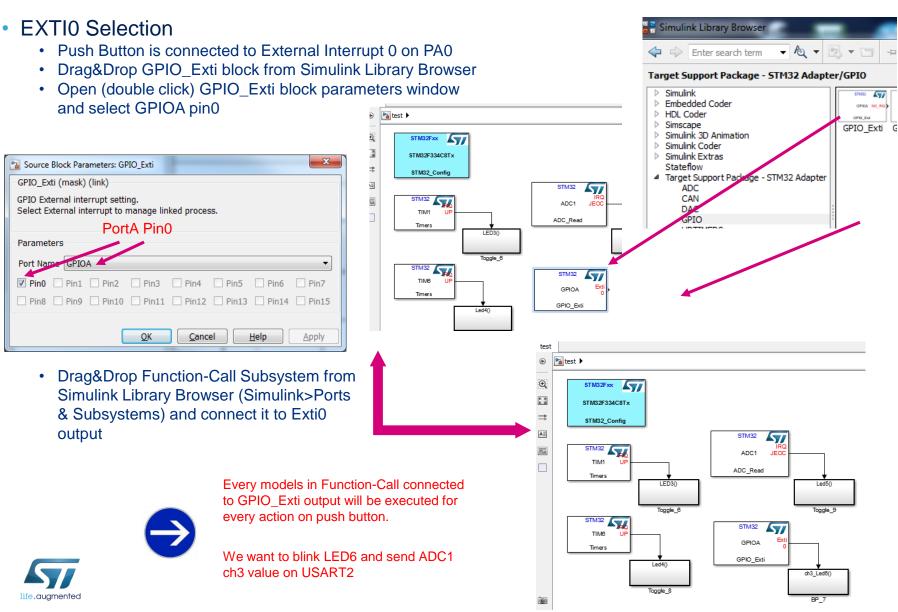


Push Button functions 68

- Software application example:
 - Use TIM1 to blink LED3 at 1Hz
 - Use TIM6 to blink LED4 at 2Hz
 - Use TIM6 to trig ADC1 channels 2&3 conversion
 - Blink Led6 when user push button is pressed
 - Send ADC1 channel 3 values on USART2 when user push button is pressed



EXTIO Selection & Configuration



Push Button Action 1/2

Blink LED6

- Open (double click) Function-Call block and Drag&Drop **GPIO** Write block
- Open (double click) GPIO_Write block parameters window and select Pin7 (LED6 is connected to Pin7)
- Send ADC1 Ch3 value on USART2
 - Drag&Drop ADC_Read block from Simulink Library Browser
 - Drag&Drop USART_Send block from Simulink Library Browser



 Open (double click) ADC_Read block parameters window and select ADC1 Ch3

USART2 Settings

 Open (double click) USART2 Send block parameters window and set buffer size.



It is mandatory to set Buffer Size as close as messages sent in order to avoid memory waste.

STM32

GPIOB

GPIO_Write

STM32

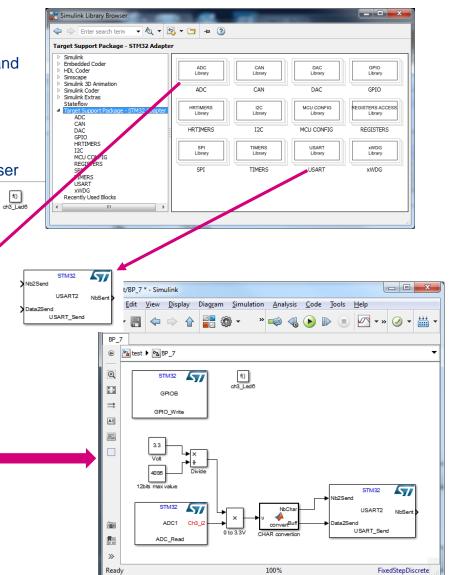
ADC1

ADC_Read

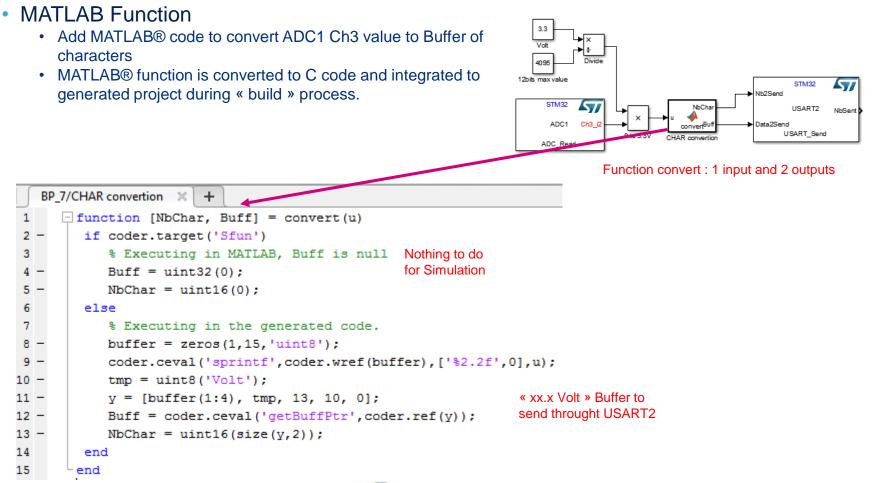
Send Buffer Size

- 16
- Add signal processing. ٠
- MATLAB® code can be added in « MATLAB Function » from Simulink Library Browser>User-Defined Functions>MATLAB Function





Push Button Action 2/2 71



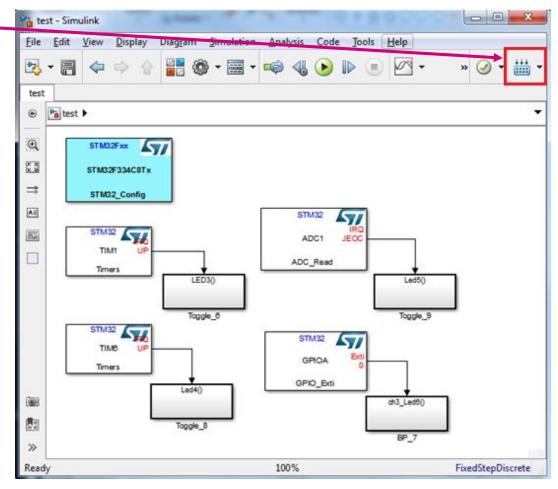


<code>"
setBuffPtr"</code> is a C function provided with STM32-MAT/TARGET that convert MATLAB® array to C pointer.



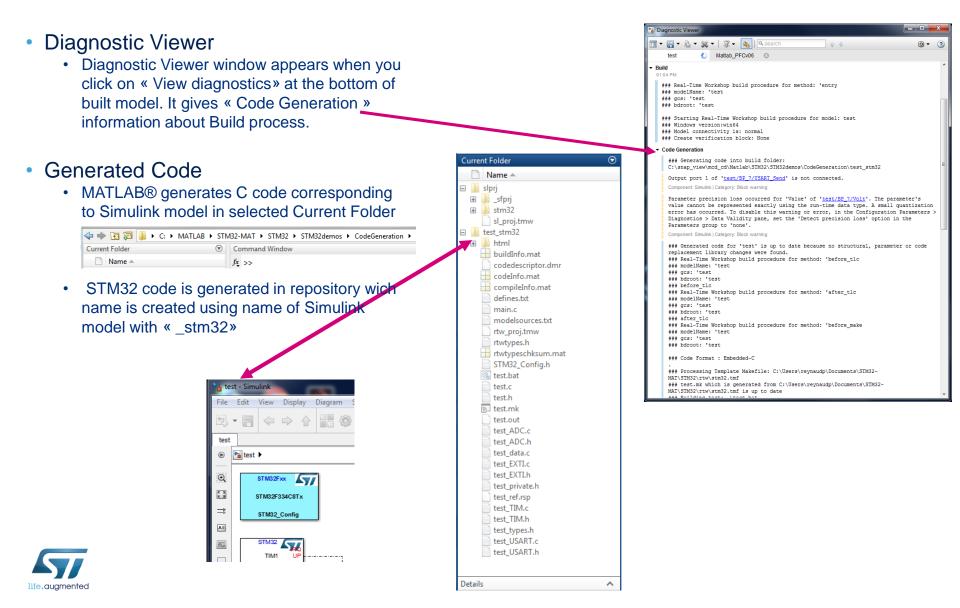
Build Application 72

- Generate code for created application
 - · Press « Build Model » to automatically generate C code and toolchain project.





MATLAB® Code Generation 1/2 73

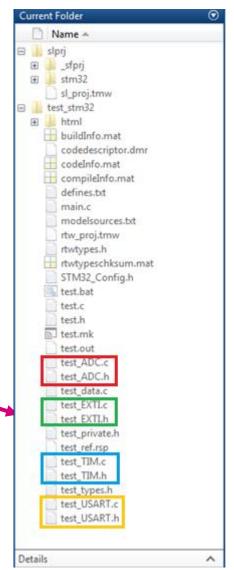


MATLAB® Code Generation 2/2

of Simulink model and peripheral name. (F) stm32 sl_proj.tmw test_stm32 htest - Simulink html Eile Edit View Display Diagram Simulation Analysis Code Jools Help • 4 🖉 • » 🥥 • 🛗 • test 🐵 🏊 test 🕨 defines.txt 0 STMOJEX main.c 13 STM02F334C8Tx \Rightarrow STM32_Config 57 STM32 rtwtypes.h STM32 84 ADC1 JECC. TIME ADC_Read Timera test.bat LED3/ test.c loggie_6 Topple_9 test.h STM22 STM32 57 b] test.mk TIME OPICA test.out Tenars test_ADC.c GPIO_Edi Led4/ test ADC.h 80 dh3_Led5() test_data.c 薗 loggie_8 test_EXTLc >> 25 test EXTLh Ready 3.3 test_private.h test_ref.rsp 40.95 57 test_TIM.c STM02 test TIM.h 57 4004 test_types,h test_USART.c 60 test_USART.h 81 View 2 warnings 100% **FixedStepDiscrete**

STM32 peripherals driver code is generated

in .c/.h files wich name is created using name





STM32CubeMX Code Generation 1/2 75

Download Application

STM32CubeMX process

- STM32CubeMX is automatically called from MATLAB® when « Download Application » has been selected from Model Configuration window.
- STM32CubeMX generates configuration code.

Generating user source code	

• STM32CubeMX adds necessary library files. .c/.h library files from HAL STM32 libraries.

Copying libraries files...

STM32CubeMX generates toolchain project
including files generated from MATLAB®

Generating EWARM project...

STM32CubeMX generated project can be open

 \Rightarrow

Click « Open Project » to automatically open project using selected toolchain.



STM32CubeMX Code Generation 2/2

STM32CubeMX project generation

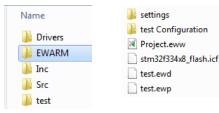
· Project is generated in same repository as ioc file. (Example with Simulink « test » project)

STM32CubeMX project contains

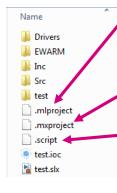
· Drivers : Contains STM32 selected library and **CMSI** files



EWARM: Contains toolchain project files (IAR for example)





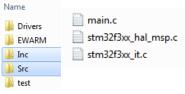


.mlproject : Generated from MATLAB®. Contains information about .c/.h files to add to project from MATLAB®

.mxproject : Generated from STM32CubeMX. Contains information about .c/.h files generated from STM32CubeMX

.script : Generated from MATLAB®. Contains STM32CubeMX command to generate project.

Inc & Src: Contains STM32CubeMX generated ٠ or modified files.



main.c : Generated from MATLAB®. It has been modified by STM32CubeMX to add project configuration.

hal msp.c: Peripherals configuration

_it.c : Interrupt handlers for configured interrupt only.

test (Simulink project name): Contains all .c/h files generated from MATLAB®

Name	test.c
Drivers	test_ADC.c
EWARM	📄 test_data.c
inc 🔐	test_EXTI.c
鷆 Src	test_TIM.c
퉬 test	test_USART.c

Toolchain Project

Toolchain settings

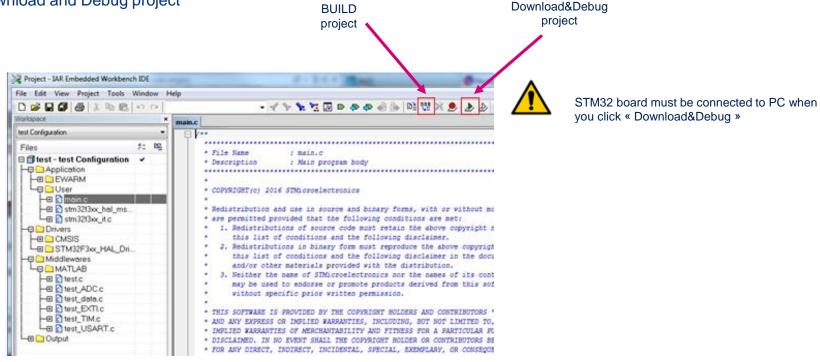
 STM32CubeMX has automatically generated project including mandatory settings. It is exactly same project at it should be generated « by hand ». Possibility to tune all settings.



Example using EWARM (IAR) toolchain for Simulink test project

Toolchain Actions

- Build project
- Download and Debug project



Run Project 78

- Simulink « test» example results •
 - · Project is started and waits at 1st main instruction.
 - Click «Go»

life.auamentea

- LD3/LD4/LD5 are blinking
- LD6 is alternatively ON and OFF when you press User button. ADC value set on PA2 (ADC1 Ch3) is sent to PC through USART.

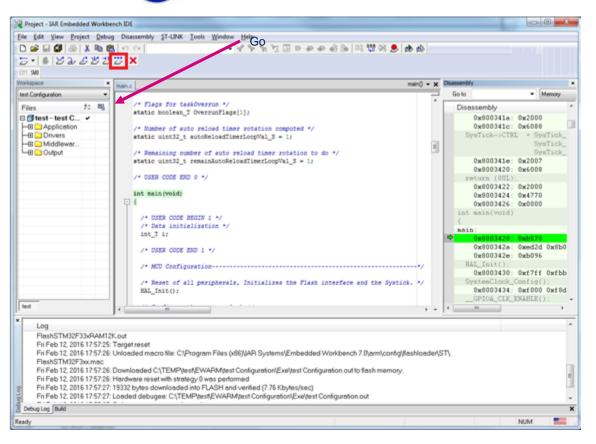
You can see ADC value on PC using PuTTY for example.

0.60Volt 1.02Volt 2.28Volt .30Volt .00Volt





Example using EWARM (IAR) toolchain for Simulink test project



Releasing your creativity with the STM32





