

T.O.M.A.S Team





Goal of the session

- □To be able to create an application using STM32CubeMX
- □ To be able to perform basic configuration, project processing and code writing within System Workbench For STM32 (SW4STM32) toolchain
- To understand better the concept of Hardware Abstraction Layer (HAL) and Low Layer (LL) libraries.
- □To be able to use some complementary debug tools



Workshop rules

- Just before the session we will provide you training slides, documentation and additional materials useful for exercises within **USER_MATERIAL.zip** file
- Most of the exercises will be done in parallel by trainer and participants.
- In case of any trouble with the exercise it is possible to reuse from USER_MATERIAL:
 - Existing ready STM32CubeMX project for particular exercise (_STM32CubeMX projects folder)
 - Existing starting SW4STM32 project:
 - ._SW4STM32 projects\STM32_Ecosystem_workshop_startups.zip
 - Source files modifications needed for each exercise: (Sources folder)
 - Already completed SW4STM32 project

._SW4STM32 projects\STM32_Ecosystem_workshop_solutions.zip



STM32CubeMX tuning

only for version 4.16.1

- Within STM32CubeMX v4.16.1 an issue with not correct linker file has been detected
- Issue will be solved within version 4.17 in coming weeks.
- In the meantime we suggest to install a patch which is located in USER_MATERIAL.zip in folder
 .\Patch for STM32CubeMX 4.16.1
- Installation guide:
 - Unzip it
 - Launch the .exe
 - Point on STM32Cube install folder
 - C:\Program Files\STMicroelectronics\STM32Cube (64 bit install)
 - C:\ Program Files (x86)\STMicroelectronics\STM32Cube (32 bit install)
 - Click ok (you should be prompted to overwrite a file)
 - Re open STM32CubeMX



Training material overview

Folder name	File name	description
_STM32CubeMX projects	.ioc	Configured STM32CubeMX project files for each exercise
_Sources	sine.h	Sine buffer example for all L4_DAC_ADC based exercises
	template_src.c	Parts of the source code needed for each exercise.
_SW4STM32_projects	STM32_Ecosystem_workshop_startups.zip	Initial projects for each exercise.
	STM32_Ecosystem_workshop_solutions.zip	Completed projects for each exercise
Documentation	*.pdf	Documentation referenced in the workshop (including board schematics)
Slides	*.pdf	Slides from the workshop







Agenda 7

08:30 – 09:00 AM	Registration and system check for pre-installed tools
09:00 – 09:30	Agenda and Marketing Introduction
09:30 – 10:00	Creation and tuning of a simple project
10:00 – 10:30	SW4STM32 tips and tricks
10:30 – 10:45	Coffee break
10:45 – 11:00	HAL presentation
11:00 – 12:00	Project migration from STM32L0 to STM32L4 platform
12:00 – 12:30	Complementary debug tools - <u>STMStudio</u>
12:30 – 13:30	Lunch
13:30 – 14:15	Complementary debug tools - <u>printf</u>
14:15 – 14:30	Quality and test of the libraries
14:30 – 14:45	Coffee break
14:45 – 15:15	Low Layer libraries presentation
15:15 – 16:30	Code optimization_using STM32Cube Low Layer library
16:30	<u>Q&A</u>

Goal of the session

- ✓ To be able to create an application using STM32CubeMX
- ✓ To be able to perform basic configuration, project processing and code writing within System Workbench For STM32 (SW4STM32) toolchain
- ✓ To understand better the concept of Hardware Abstraction Layer and Low Layer libraries.
- ✓ To be able to use some complementary debug tools



Materials on Google Drive 10

- All user materials can be downloaded from the Google Drive at address:
 - https://drive.google.com/open?id=0B5TcqDdbIL1IMG02bVIYQmN5MEE







Explanation of the acronyms 12

- ETM Enhanced Trace Macrocell
- □ ITM Instrumentation Trace Macrocell
- □ HAL Libraries Hardware Abstraction Layer Libraries
- LL Libraries Low Layer Libraries
- □ **MSP/msp** microcontroller support package
- **PPP** marking of any peripheral used to indicate name of the function, structure common for many peripherals
- **SVN** Subversion, Version Control System one of the tool for sharing the software projects within working groups
- SW4STM32 System Workbench for STM32 free toolchain for STM32 microcontrollers based on Eclipse editor, gcc C compiler and OpenOCD debugger, prepared by AC6 company
- SWD Single Wire Debug
- SWIM Single Wire Interface Module programming/debugging interface in STM8 microcontrollers
- SWO Single Wire Output one wire trace interface available in CortexM based microcontrollers (except CortexM0 and CortexM0+)
- **TRGO** Trigger Output internal output signal of the timer used for synchronization with other peripherals (timers, DAC, ADC)





May we ask you for your feedback, please ?



We need your feedback to do better sessions

- In case you would like to share with us your feedback, please use a survey, we have prepared for this session.
- Survey is anonymous
- You can find it under the link:

https://www.surveymonkey.com/r/handson_ecosystem



Let's start!



www.st.com/mcu

