Elektrobit

MCAL Wrapper (McalExt) integration notes

for TRICORE TC38XQ

product release 8.8.0

Elektrobit Automotive GmbH
Am Wolfsmantel 46
91058 Erlangen, Germany
Phone: +49 9131 7701 0
Fax: +49 9131 7701 6333
Email: info.automotive@elektrobit.com

Technical support

https://www.elektrobit.com/support

Legal disclaimer

Confidential information.

Table of Contents

1. Overview of MCAL Integration release notes 4
    1.1. Location of MCAL documentation 4
2. Supported toolchain 5
    2.1. Toolchain options 5
3. Scope of this release 6
    3.1. Platforms Module 6
    3.2. Third-party MCAL version 6
    3.3. Third-party MCAL modules 6
    3.4. Third-party MCAL patches by Elektrobit Automotive GmbH 8
        3.4.1. Use of original or patched version in third-party MCAL modules 8
4. Overview of McalExt Integration notes 9
    4.1. Background information 9
5. MCAL Wrapper (McalExt) module description 10
    5.1. Vendor Delivery package 10
    5.2. Connection with Vendor delivery package 10
        5.2.1. `plugin.xml` connection description 10
        5.2.2. Makefile connection description 11
        5.2.3. anchors connection description 12
    5.3. MCAL Wrapper(McalExt) plugin Elektrobit Automotive GmbH file description 12
    5.4. Usage of McalExt wrapper module 15
        5.4.1. Elektrobit Automotive GmbH build environment 15
        5.4.2. User build environment 15
    5.5. Update MCAL version by the user 15

1. Overview of MCAL Integration release notes

Welcome to the MCAL release notes. These release notes are target and derivative-specific.

Chapter 2, “Supported toolchain“ provides information about the supported toolchain.

Chapter 3, “Scope of this release“ provides specific information about the hardware-dependent third-party modules contained in this EB tresos AutoCore release:

- AUTOSAR version and revision of your hardware-dependent modules
- SWS version and revision
- Module version
- Supplier of your hardware-dependent modules

1.1. Location of MCAL documentation

Depending on the platform release that you purchased, some of the modules may be supplied to Elektrobit Automotive GmbH by third-parties. All MCAL modules are documented outside of these release notes. This documentation contains additional information about the third-party MCAL modules and the patches that were made by Elektrobit Automotive GmbH.

You can find the MCAL module documentation in the following locations:

- `$TRESOS_BASE/doc/5.2_MCAL_Wrapper_Integration_release_notes`
- `$TRESOS_BASE/plugins/<McalExt plugin>/doc`
- In the online help of EB tresos Studio.

For information about the online help in EB tresos Studio, see the EB tresos Studio user documentation.

2. Supported toolchain

This release of EB tresos AutoCore supports TASKING_TriCore-VX_v6.2r2

2.1. Toolchain options

The toolchain options summarize under which conditions this release needs to be built. The release is tested using these toolchain options. If you change the compiler options, consider this release *untested*.

Compiler	**Options**
\ctc\bin\ctc.exe	`--core=tc1.6.2 -Wa-gAHLs -Wa-Ogs -Wa--error-limit=42 --emit-locals=-equ,-symbols --iso=99 --eabi-compliant --integer-enumeration --language=-comments,-gcc,+volatile,-strings --switch=auto --align=0 --default-near-size=0 --default-a0-size=0 --default-a1-size=0 -O2ROPYG --tradeoff=4 -g --source -DOS_TRICOREARCH=OS_TRICOREARCH_162 -DCOMPILERCFG_EXTENSION_MCAL_FILE -DUSE_TASKING_INIT=0 -DOS_TOOL=OS_tasking -DOS_ARCH=OS_TRICORE`
Assembler	**Options**
\ctc\bin\astc.exe	`--core=tc1.6.2 -gAHLs -Ogs --error-limit=42 --emit-locals=-equ,-symbols -mt`
Linker	**Options**
\ctc\bin\ltc.exe	`-I$(BOARD_PROJECT_PATH) -M$(MAP_FILE) --user-provided-initialization-code --library=fp_fpu --library=rt`

3. Scope of this release

3.1. Platforms Module

This release of the `Platforms` module contains the mandatory and derivative-specific implementation part of the `Base` module.

This `Platforms` module shall be used only for TC38XQ derivatives.

This module is tested only on hardware with the same sub-derivative as the third-party MCAL version. Other sub-derivatives are not tested.

3.2. Third-party MCAL version

This release contains the MCAL release MC-ISAR_AS422_TC3xx_1.40.0_PR from Infineon.

This release of EB tresos AutoCore is tested only with sub-derivative TC387.

3.3. Third-party MCAL modules

This release includes the hardware-dependent third-party MCAL modules listed in the table below.

Module name	AUTOSAR version and revision	SWS version and revision	Module version	Supplier
Adc	4.2.2	4.2.2	10.40.0	Infineon
Can	4.2.2	4.2.2	10.40.2	Infineon
CanTrcv	4.2.2	4.2.2	10.40.1	Infineon
CanTrcv	4.2.2	4.2.2	10.40.1	Infineon
Crc	4.2.2	4.2.2	10.40.0	Infineon
Dio	4.2.2	4.2.2	10.40.0	Infineon
Dma	4.2.2	4.2.2	10.40.1	Infineon
Dsadc	4.2.2	4.2.2	10.40.1	Infineon

Module name	AUTOSAR version and revision	SWS version and revision	Module version	Supplier
Eth	4.2.2	4.2.2	10.40.1	Infineon
Fee	4.2.2	4.2.2	10.40.2	Infineon
Fls	4.2.2	4.2.2	10.40.1	Infineon
FlsLoader	4.2.2	4.2.2	10.40.1	Infineon
Fr	4.2.2	4.2.2	10.40.1	Infineon
Gpt	4.2.2	4.2.2	10.40.0	Infineon
Hssl	4.2.2	4.2.2	10.40.2	Infineon
I2c	4.2.2	4.2.2	10.40.1	Infineon
Icu	4.2.2	4.2.2	10.40.1	Infineon
Iom	4.2.2	4.2.2	10.40.0	Infineon
Lin	4.2.2	4.2.2	10.40.1	Infineon
McalLib	4.2.2	4.2.2	10.40.1	Infineon
Mcu	4.2.2	4.2.2	10.40.2	Infineon
Ocu	4.2.2	4.2.2	10.40.0	Infineon
Port	4.2.2	4.2.2	10.40.1	Infineon
Pwm	4.2.2	4.2.2	10.40.2	Infineon
Sent	4.2.2	4.2.2	10.40.1	Infineon
Smu	4.2.2	4.2.2	10.40.0	Infineon
Spi	4.2.2	4.2.2	10.40.2	Infineon
Stm	4.2.2	4.2.2	10.40.0	Infineon
Uart	4.2.2	4.2.2	10.40.2	Infineon
Wdg	4.2.2	4.2.2	10.40.1	Infineon
ResourceM	4.2.2	4.2.2	10.40.0	Infineon

Table 3.1. Third-party hardware-dependent modules

3.4. Third-party MCAL patches by Elektrobit Automotive GmbH

3.4.1. Use of original or patched version in third-party MCAL modules

The EB tresos Installer is the installation tool for the third-party MCAL modules. For more information on the EB tresos Installer, see 1.1_EB_tresos_installation_guide.pdf. In the EB tresos Installer, you can choose one of the following options:

- During installation it is possible to disable the update package. Afterwards, only original MCAL modules are part of your installation. For more information, see Chapter 5, “MCAL Wrapper (McalExt) module description“.
- If you also install the update package, all changes by Elektrobit Automotive GmbH that are described in Section 5.3, “MCAL Wrapper(McalExt) plugin Elektrobit Automotive GmbH file description” are part of your EB tresos AutoCore installation.

  Additionally, with this update package you can switch between the original version and the patched version of each MCAL module.

The update package includes the `perform_MCAL_change.bat` batch file in McalExt plugin. Use this batch file to switch from one version to another.

In the `perform_MCAL_change.bat`, you need to specify one of the following parameters:

- `EB_update`: Update the files in the plug-in with content of Elektrobit Automotive GmbH.
- `origin`: Reset the files in the plug-in with Vendor content.

WARNING ⚠

Changes due to execution of the batch file affects all projects

The changes affect all projects that use the changed plug-in. Therefore, execute the batch-file before code generation for your project.

4. Overview of McalExt Integration notes

These Integration notes are target and derivative-specific.

This user guide describes the concepts and the configuration of the plugin:

- `McalExt`

4.1. Background information

`McalExt` is a wrapper plugin that makes the connection between the Vendor MCAL delivery, EB tresos Studio, Elektrobit Automotive GmbH build environment and will allow the user to use the Vendor MCAL with EB tresos Studio with as less modifications as possible.

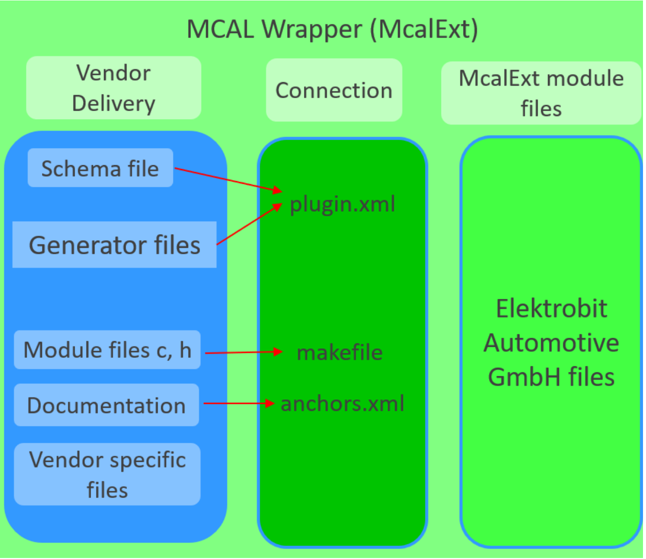


Figure 4.1. MCAL Wrapper (McalExt) overview

MCAL Wrapper (McalExt) integration notes also:

- provides information about the Vendor Delivery package, see Section 5.1, “Vendor Delivery package”
- provides information about the connection with Vendor delivery package, see Section 5.2, “Connection with Vendor delivery package”
- provides information about Elektrobit Automotive GmbH files,see Section 5.3, “MCAL Wrapper(McalExt) plugin Elektrobit Automotive GmbH file description”

5. MCAL Wrapper (McalExt) module description

5.1. Vendor Delivery package

Represents the package that is delivered by different MCAL Vendors (e.g. Infineon, Renesas, NXP, etc.) and contains different MCAL modules. For the MCAL modules and version that are integrated in this release please see Section 3.3, "Third-party MCAL modules".

5.2. Connection with Vendor delivery package

The Vendor delivery has a structure that can't be used directly in Elektrobit Automotive GmbH environment. The `McalExt` wrapper module is introduced to make the connections between Elektrobit Automotive GmbH environment and the Vendor delivery. The connection is made in different files from `McalExt` wrapper plugin:

`plugin.xml` connection, see Section 5.2.1, "`plugin.xml` connection description"

`makefile` connection, see Section 5.2.2, "Makefile connection description"

`anchors.xml` connection, see Section 5.2.3, "anchors connection description"

5.2.1. `plugin.xml` connection description

- Schema file e.g.:

  ```
  <schema>
  <manager class="dreisoft.tresos.autosar2.resourcehandling.AutosarSchemaManager"/>
  <!-- Define the file(s) from which to load the schemas -->
  <resource value="MCAL_Delivery/PathToSchemaFile/ModuleName.xdm"  type="xdm"/>
  </schema>
  ```

  PathToSchemaFile - represents the path to where the module schema file is located in the MCAL Vendor delivery.

  ModuleName - represents the name of the schema file that should be used.

- Code Generator e.g.:

  ```
  <!-- common template path parameters -->
  <parameter name="templates"
  ```

```
mode="generate,verify" value="MCAL_Delivery/PathToGenerator"/>
```

PathToGenerator - represents the path where the generator is present in the Vendor delivery.

- Ant Code Generator e.g.:

```
<generator moduleId="ModuleId"
class="dreisoft.tresos.generator.ng.api.NGGenerator"
id="ModuleId_UniqueNGGeneratorId"
step="post"> <!-- run after code-generation -->
<parameter name="buildfile" value="MCAL_Delivery/PathToAntGenerator/AntGeneratorFile.xml"/>
</generator>
```

PathToAntGenerator - represents the path where the ant generator file is present in the Vendor delivery.

AntGeneratorFile - represents the name of the ant generator file delivered by the Vendor.

5.2.2. Makefile connection description

For each MCAL Module that is integrated in `McalExt` wrapper plugin, a make folder is present which contains the makefiles for the respective MCAL module:

- `Module_defs.mak` file - registers the file(s) that are present in Vendor delivery and the files that are generated for this module.

```
McalExt_GEN_FILES += $(McalExt_OUTPUT_PATH)\inc\ModuleName_Cfg.h
McalExt_GEN_FILES += $(McalExt_OUTPUT_PATH)\inc\ModuleName_PBcfg.h
McalExt_GEN_FILES += $(McalExt_OUTPUT_PATH)\src\ModuleName_PBcfg.c
CC_INCLUDE_PATH   += $(McalExt_CORE_PATH)\MCAL_Delivery\PathToHeaderFiles
```

ModuleName - represents the name of the header files that are generated.

PathToHeaderFiles - represents the path from the Vendor delivery where the static header files are located.

- `Module_rules.mak` file - registers the specific module file(s) that are needed for compilation.

```
McalExt_src_FILES += $(McalExt_CORE_PATH)\MCAL_Delivery\PathToSourceFile\Module_Name.c
McalExt_src_FILES += $(McalExt_OUTPUT_PATH)\src\ModuleName_PBcfg.c
```

ModuleName - represents the name of c files that are generated or are present in the Vendor delivery.

PathToSourceFile - represents the path from the Vendor delivery where c static files are present.

All the MCAL modules makefiles will be included in `McalExt_defs.mak` and `McalExt_rules.mak` files only if the respective module is used in the tresos Studio project:

- `McalExt_defs.mak` file

  ```
  ifeq ($(Can_VARIANT),ModuleNameVariant)
  include $(McalExt_CORE_PATH)\make\make_Can\Can_defs.mak
  endif
  ```

- `McalExt_rules.mak` file

  ```
  ifeq ($(Can_VARIANT),ModuleNameVariant)
  include $(McalExt_CORE_PATH)\make\make_Can\Can_rules.mak
  endif
  ```

  ModuleNameVariant - represents the MCAL module name that is used in the tresos Studio project.

5.2.3. anchors connection description

- `anchors.xml` file registers the MCAL documentation which is shown in the EB tresos Studio help window, e.g:

  ```
  <topic label="DocName" href="PathToDoc/DocName"/>
  <topic label="DocName" href="PathToDoc/DocName"/>
  <topic label="DocName" href="PathToDoc/DocName"/>
  ```

  PathToDoc - represents the path where the MCAL documents are located.

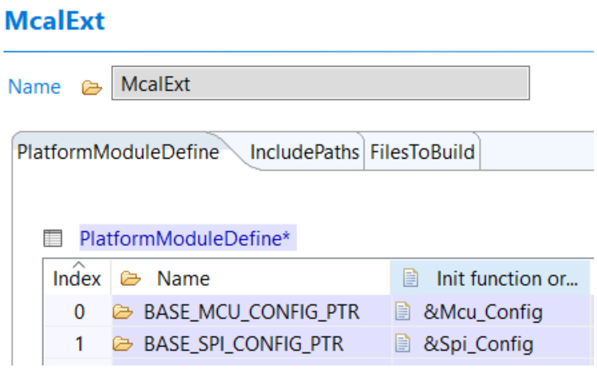
  DocName - represents the name of the MCAL documents.

5.3. MCAL Wrapper(McalExt) plugin Elektrobit Automotive GmbH file description

Some patches made by Elektrobit Automotive GmbH are due to missing or incomplete files in the MCAL Vendor delivery. Additionally, some EB tresos Studio features are enabled. These patches are separated from the original installation files.

- `McalExt` wrapper schema file `McalExt.xdm` located in `config` folder allows the user to configure several parameters that can be used in the project:

- **PlatformModuleDefine** – the user can configure defines that will be generate in Platforms_Modules.h file and that can be used in the project, e.g.: configure Mcu configuration pointer that will be used in Mcu_Init() function.



Name: McalExt

Index	Name	Init function or...
0	BASE_MCU_CONFIG_PTR	&Mcu_Config
1	BASE_SPI_CONFIG_PTR	&Spi_Config

Figure 5.1. Platform Module Define Parameter configuration

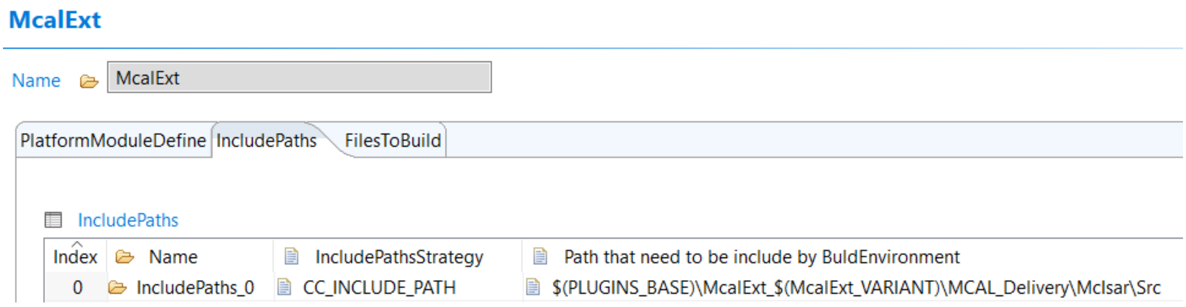
Generated file content example:

```
#define BASE_MCU_CONFIG_PTR              &Mcu_Config
#define BASE_SPI_CONFIG_PTR              &Spi_Config
```

Usage of defines(EcuM_DriverInitListOne()) e.g.:

```
/* *** Call service Init of module Mcu *** */
Mcu_Init(BASE_MCU_CONFIG_PTR);
```

- **IncludePaths** – allows the user to configure different paths that need to be included by the build environment. This will be generated in the McalExtWrapper.mak file:



Name: McalExt

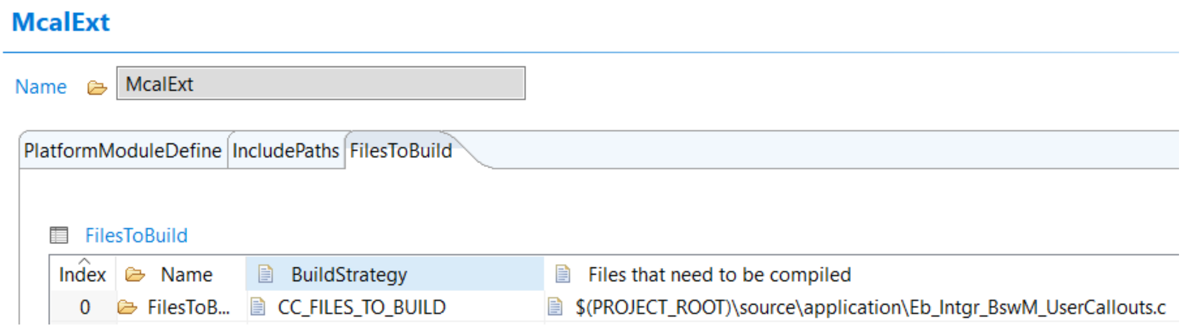
Index	Name	IncludePathsStrategy	Path that need to be include by BuldEnvironment
0	IncludePaths_0	CC_INCLUDE_PATH	$(PLUGINS_BASE)\McalExt_$(McalExt_VARIANT)\MCAL_Delivery\McIsar\Src

Figure 5.2. Path that need to be include by BuldEnvironment

Generated content in McalExtWrapper.mak file:

```
CC_INCLUDE_PATH += $(PLUGINS_BASE)\McalExt_$(McalExt_VARIANT)\MCAL_Delivery\McIsar\Src
```

- **FilesToBuild** - allows the user to configure different files that need to be compiled. This will be generated in the McalExtWrapper.mak file:



McalExt

Name: McalExt

FilesToBuild

Index	Name	BuildStrategy	Files that need to be compiled
0	FilesToB...	CC_FILES_TO_BUILD	$(PROJECT_ROOT)\source\application\Eb_Intgr_BswM_UserCallouts.c

Figure 5.3. Files that need to be compiled

Generated content in McalExtWrapper.mak file:

```
CC_FILES_TO_BUILD  += $(PROJECT_ROOT)\source\application\Eb_Intgr_BswM_UserCallouts.c
```

- Preconfiguration and Recommended configuration – In `config_ext` folder there can be present files for different MCAL modules, e.g:
  - Preconfiguration files which contain the parameter configuration value that should not be modified by the user e.g. McuResetReasonConf.
  - Recommended configuration – the configuration which was validated by Elektrobit Automotive GmbH while performing IP3/QP2(if this was ordered), e.g.: Mcu recommended configuration that contain the clock configuration and other Mcu related parameter configuration. The user can decide if the Recomended configuration is used or not.
- `swcd` - Includes the BSWMD files which are mandatory since AUTOSAR 4.0. Those files are used by BSW modules and EB tresos Studio wizards provided by Elektrobit Automotive GmbH.
  - Generation of Exclusive areas in EB tresos AutoCore Generic `Rte` module.
  - Mapping of Main function in EB tresos AutoCore Generic `Rte` module.
  - Generation of MemMap header file(s) in EB tresos AutoCore Generic `MemMap` module.
- `include` - contains header file(s) that are created/patched by Elektrobit Automotive GmbH.
- `src` - contains source file(s) that should be compiled, created by Elektrobit Automotive GmbH.
- `resources` - Includes several xml-based service needs assistant or properties files which are provided by Elektrobit Automotive GmbH. These files support the customer to faster complete a valid configuration.
  - Dem_Events.xml – Dem event generation in the EB tresos AutoCore Generic `Dem` module.

- EcuM initialization in the EB tresos AutoCore Generic `EcuM` module.
- SchM Main function handling in the EB tresos AutoCore Generic `Rte` module.

5.4. Usage of McalExt wrapper module

5.4.1. Elektrobit Automotive GmbH build environment

In order to use the `McalExt` wrapper module in a project that is using Elektrobit Automotive GmbH environment the user needs to:

- Add `McalExt` plugin to the project configuration.
- Add the needed MCAL plugins to the project configuration.

5.4.2. User build environment

When a different build enviroment(other then Elektrobit Automotive GmbH) is used in the project, the user needs to:

- Add `McalExt` plugin to the project configuration.
- Add the needed MCAL plugins to the project configuration
- Add all the files that are configured in the module makefiles and all file configured in the "IncludePaths" and "FilesToBuild" from McalExt module configuration, see IncludePaths and FilesToBuild description from Section 5.3, “MCAL Wrapper(McalExt) plugin Elektrobit Automotive GmbH file description”

5.5. Update MCAL version by the user

When then MCAL version is updated the user needs to make some verification:

- Check if all the used files are at the same location. If the the Vendor doesn’t modify the MCAL installed structure then the used path should be the same.
- During integration, Elektrobit Automotive GmbH applies patches on the Vendor files due to some bugs or due to some incompatibly. These patches are present in the files with the .EB_update extension. When the user installs the new MCAL version, he needs to verify if those patches need to be applied on the new MCAL version files.