User's Manual

AUTOSAR MCAL R4.0.3
User's Manual

MCU Driver Component Ver.1.0.2

Generation Tool User's Manual

Target Device:
RH850/P1x-C

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (http://www.renesas.com).

Renesas Electronics
www.renesas.com

Rev.1.00 Jan 2017

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other disputes involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawing, chart, program, algorithm, application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics products.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (space and undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. When using the Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat radiation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions or failure or accident arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please ensure to implement safety measures to guard them against the possibility of bodily injury, injury or damage caused by fire, and social damage in the event of failure or malfunction of Renesas Electronics products, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures by your own responsibility as warranty for your products/system. Because the evaluation of microcomputer software alone is very difficult and not practical, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please investigate applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive carefully and sufficiently and use Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall not use Renesas Electronics products or technologies for (1) any purpose relating to the development, design, manufacture, use, stockpiling, etc., of weapons of mass destruction, such as nuclear weapons, chemical weapons, or biological weapons, or missiles (including unmanned aerial vehicles (UAVs)) for delivering such weapons, (2) any purpose relating to the development, design, manufacture, or use of conventional weapons, or (3) any other purpose of disturbing international peace and security, and you shall not sell, export, lease, transfer, or release Renesas Electronics products or technologies to any third party whether directly or indirectly with knowledge or reason to know that the third party or any other party will engage in the activities described above. When exporting, selling, transferring, etc., Renesas Electronics products or technologies, you shall comply with any applicable export control laws and regulations promulgated and administered by the governments of the countries asserting jurisdiction over the parties or transactions.
10. Please acknowledge and agree that you shall bear all the losses and damages which are incurred from the misuse or violation of the terms and conditions described in this document, including this notice, and hold Renesas Electronics harmless, if such misuse or violation results from your resale or making Renesas Electronics products available any third party.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

Abbreviations and Acronyms

Abbreviation / Acronym	Description
AUTOSAR	AUTomotive Open System ARchitecture
ARXML	Autosar eXtensible Mark-up Language
BSWMDT	Basic Software Module Description Template
CFD	Control Flow Diagram
DFD	Data Flow Diagram
DEM	Diagnostic Event Manager
ECM	Error Control Module
ECU	Electronic Control Unit
Id	Identifier
MCAL	Micro Controller Abstraction Layer
MCU	Micro Controller Unit
SPAL	Standard Peripheral Abstraction Layer
XML	eXtensible Mark-up Language

Definitions

Terminology	Description
BSWMDT File	This file is the template for the Basic Software Module Description.
Configuration XML File	This file contains the setting of command line options.
ECU Configuration Description File	Input file to MCU Driver MCAL Code Generator Tool. It is generated by ECU Configuration Editor.
Sl.No	Serial Number.

Table of Contents

Chapter 1 Introduction 9
1.1. Document Overview 9
Chapter 2 Reference 11
2.1. Reference Documents 11
2.2. Trademark Notice 11
Chapter 3 Code Generation Overview 13
Chapter 4 Input Files 15
Chapter 5 Output Files 17
Chapter 6 Precautions 19
Chapter 7 User Configuration Validation 21
Chapter 8 Configuration Overview 23
8.1 Container Overview 23
8.1.1 Pre Compile Time Configurable Parameters 24
8.1.2 Post Build Time Configurable Parameters 30
Chapter 9 Messages 53
9.1 Error Messages 53
9.2 Warning Messages 61
9.3 Information Messages 61

List of Figures

Figure 3-1 Overview of Code Generation 13
Figure 3-2 Flow-Diagram of Code Generation 14
Figure 8-1 Configuration overview 23

List of Tables

Table 5-1 Output Files Description.......... 17
Table 8-1 Pre Compile Time Configurable Parameters 24
Table 8-2 Post Build Time Configurable Parameters.......... 30

Chapter 1 Introduction

The MCU Driver component provides services for basic microcontroller initialization, reset, standby modes, handles ECM and microcontroller specific functions required from other SPAL components.

The MCU Driver Component comprises of two sections i.e., Embedded Software and MCAL Code Generator Tool to achieve scalability and configurability.

The document describes the MCU module specific inputs and outputs of the MCAL Code Generator Tool that is the common code generator engine used for the generation of the configuration code for all MCAL modules. MCAL Code Generator Tool is a command line tool that extracts information from ECU Configuration Description File, BSWMDT File and generates MCU Driver Configuration source and header files (Mcu_PBcfg.c, Mcu_Hardware.c, Mcu_Hardware.h, Mcu_Cbk.h and Mcu_Cfg.h).

This document contains information on the options, input and output files of the MCAL Code Generator Tool. In addition, this manual covers a step-by-step procedure for the usage of MCAL Code Generator Tool.

1.1. Document Overview

This user manual is organized as given in the table below:

Section	Contents
Section 1 (Introduction)	Provides an introduction to the document and explains how information is organized in this manual.
Section 2 (Reference)	Provides a list of documents referred while developing this document.
Section 3 (Code Generation Overview)	Provides the overview of Code generation.
Section 4 (Input Files)	Provides information about ECU Configuration Description File.
Section 5 (Output Files)	Explains the output files that are generated by the MCAL Code Generator Tool
Section 6 (Precautions)	Contains precautions to be taken during configuration of ECU Configuration Description File.
Section 7 (User Configuration Validation)	Describes about user configuration validation done by the MCAL Code Generator Tool
Section 8 (Configuration overview)	Describes the overview of the configuration parameters.
Section 9 (Messages)	Describes all the Error/Warning/Information messages of R4.0.3 which helps the user to understand the probable reason for the same.

Chapter 2 Reference

2.1. Reference Documents

The following table lists the documents referred to develop this document:

Sl. No	Title	Version
1.	AUTOSAR_SWS_MCUDriver	3.2.0
2.	MCAL_CodeGenerator_Tool_UserManual.pdf	1.7
3.	R20UT3828EJ0100-AUTOSAR.pdf	1.0.2

2.2. Trademark Notice

Microsoft and Windows are trademarks/registered trademarks of Microsoft Corporation.

Chapter 3 Code Generation Overview

Overview of Code Generation is shown below

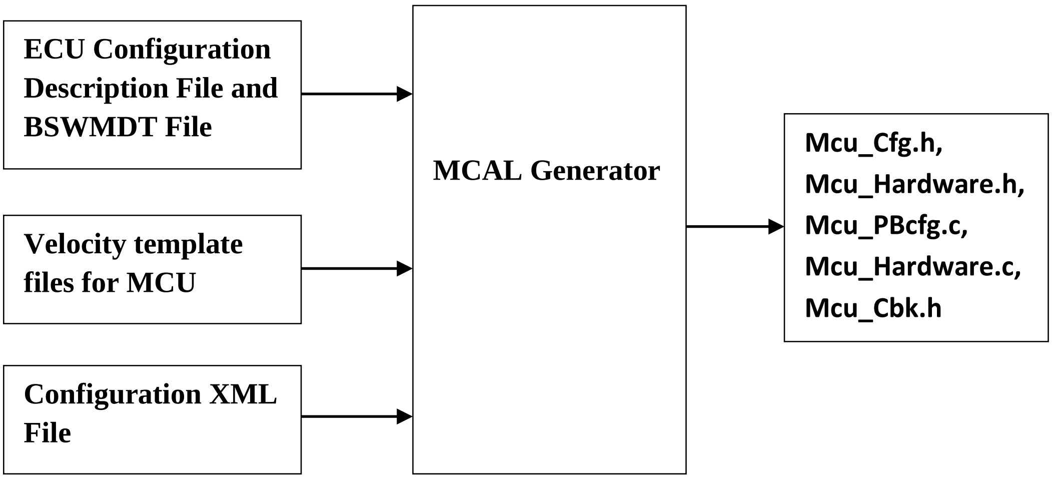
ECU Configuration Description File and BSWMDT File
Velocity template files for MCU
Configuration XML File
MCAL Generator
Mcu_Cfg.h, Mcu_Hardware.h, Mcu_PBcfg.c, Mcu_Hardware.c, Mcu_Cbk.h

Figure 3-1 Overview of Code Generation

- ECU Configuration Description File (.arxml):
  This file will contain MCU Driver specific configuration information. This file should be generated by AUTOSAR specified Configuration Editor.

- BSWMDT File (.arxml):
  MCAL Code Generator Tool uses “Common Published Information” from MCU module specific BSWMDT File. MCU module specific BSWMDT File should not be updated manually since it is “Static Configuration” file.

- Velocity template files:
  Mcu_PBcfg_c.vm, Mcu_Cfg_h.vm, Mcu_Hardware_h.vm, Mcu_Hardware_c.vm, Mcu_Validate.vm, Mcu_Cbk_h.vm and CommonHelper.vm
  They are interpreted by the MCAL Code Generator Tool in order to provide user input validation and generate the final output file needed by the AUTOSAR configuration chain.They are the "logic" of the Code Generator.

- Configuration XML File (.xml):
  This file is used to specify which velocity template to use and their location and the name of the output file generated.

For the error free input file, the MCAL Code Generator Tool generates the following output files: Mcu_Cfg.h, Mcu_Hardware.h, Mcu_PBcfg.c, Mcu_Cbk.h and Mcu_Hardware.c and displays appropriate context sensitive error messages for wrong input and exits.

ECU Configuration Description File can be created or edited using ECU Configuration Editor.

Concept of execution for MCU Driver MCAL Code Generator Tool is as follows:

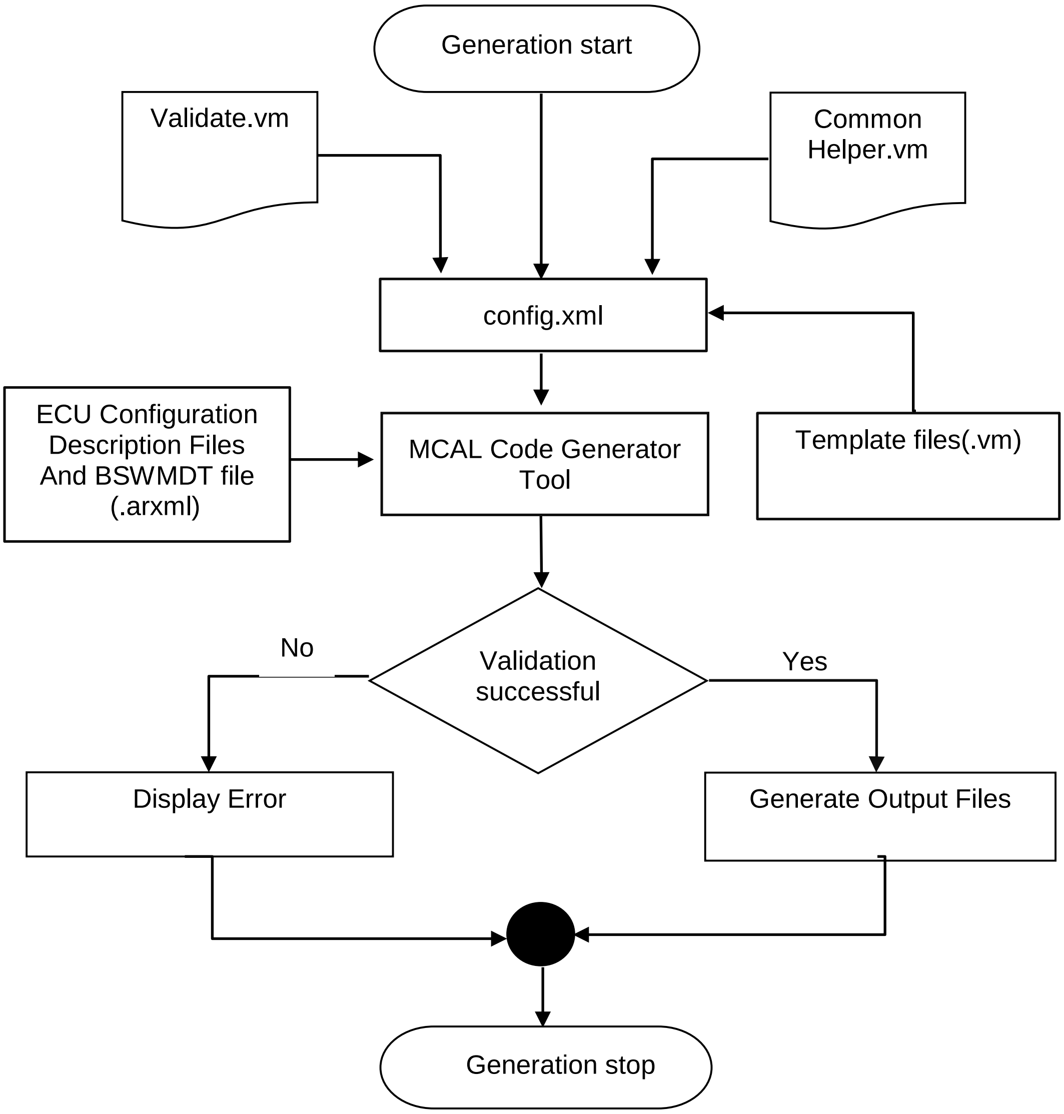

Figure 3-2 Flow-Diagram of Code Generation

The module “Validate” will validate the configuration (contents of ECU Configuration Description File(s) as input). If there are incorrect values or incorrect dependencies, the MCAL Code Generator Tool will display error, warning and information messages. In case of errors, the MCAL Code Generator Tool will abort the execution.

Mcu_Cfg_h.vm / Mcu_PBcfg_c.vm will generate compiler switch / structures necessary to the AUTOSAR Configuration chain and vendor specific parameters.

Mcu_Hardware_h.vm / Mcu_Hardware_c.vm will generate hardware related info (defines number of actual instances / channels used / structure to access to the I/O mapped peripheral).

Remark Please consult the general MCAL Code Generator Tool User Manual (MCAL_CodeGenerator_Tool_UserManual.pdf) and GettingStarted_MCAL_Drivers_X1x (R20UT3828EJ0100-AUTOSAR.pdf) for details about the MCAL Code Generator Tool command line options.

Chapter 4 Input Files

MCAL Code Generator Tool will accept the config.xml file which has paths to the Velocity template files for generating Mcu Driver files. MCAL Code Generator Tool need ECU Configuration Description File(s) and BSWMDT File as inputs to generate Mcu Driver specific source files. Hence ECU Configuration Description File should contain configuration of MCU Driver module. MCAL Code Generator Tool ignores any other AUTOSAR component configured in the ECU Configuration Description File. ECU Configuration Description File can be generated using configuration editor.

ECU Configuration Description File must comply with AUTOSAR standard ECU Configuration Description File format

Remark The detailed explanation about the parameters and containers are found in Parameter Definition File.

Chapter 5 Output Files

MCAL Code Generator Tool generates configuration details in C Header and C Source files Mcu_Cfg.h, Mcu_Hardware.c, Mcu_Hardware.h, Mcu_PBcfg.c and Mcu_Cbk.h.

The content of each output file is given in the table below:

Table 5-1 Output Files Description

Output File	Details
Mcu_Cfg.h	This file contains pre-compile time parameters.
Mcu_Hardware.c	This file contains the definitions for addresses of the hardware registers used in the MCU Driver Module.
Mcu_Hardware.h	This file contains the declarations for addresses of the hardware registers used in the MCU Driver Module.
Mcu_PBcfg.c	This file contains post-build configuration data.
Mcu_Cbk.h	This file contains call back function of Dem.

Remark Output files generated by MCAL Code Generator Tool should not be modified or edited manually.

Chapter 6 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R4.0.3 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain MCU Driver module.
- All the function names and the string values configured should follow C syntax for variables. It can only contain alphanumeric characters and "_". It should start with an alphabet.
- MCU Configuration is dependent on DEM module description file.
- Configuration xml file :config.xml should convey the velocity template file location and output file location.
- Configuration XML File should contain the file extension '.xml'.
- If the output files generated by MCAL Code Generator Tool are modified externally, then they may not produce the expected results or may lead to error/warning/Information messages.
- Short Name for a container should be unique within a name space.
- An error free ECU Configuration Description File generated from configuration editor has to be provided as input to the MCU Driver MCAL Code Generator Tool. Otherwise MCAL Code Generator Tool may not produce the expected results or may lead to errors/warnings/information messages.

Remarks: Please refer MCU Component User Manual (R20UT3651EJ0100-AUTOSAR.pdf) for deviations from AUTOSAR specifications.

Chapter 7 User Configuration Validation

This section provides help to analyze the error, warning and information messages displayed during the execution of MCU Driver MCAL code generator tool. It ensures conformance of input file with syntax and semantics. It also performs validation on the input file for correctness of the data.

For more details on list of Error/Warning/Information messages that are displayed as a result of input file(s) validation, refer Chapter 9 “*Messages*”.

The MCAL Code Generator Tool displays error or warning or information when the user has configured incorrect inputs. The format of Error/Warning/Information message is as shown below.

<message_type>_<vendor_id>_<module_id>_<message_id>:<message_content>.

where,

- <message_type> : ERR/WARNING/INFO
- < vendor_id > : vendor Id = 59
- < module_id > : 101- MCU Driver Module id (101) for user configuration checks.
- < Message_id.> : 001-999
- <message_content>: Message content provides information about error or warning or information displayed when the user has configured incorrect inputs.

  File Name’ and ‘Path’ need not be present for all Error/Warning/Information messages

  File Name: Name of the file in which the error has occurred.

  Path: Absolute path of the container in which the parameter is present

Chapter 8 Configuration Overview

8.1 Container Overview

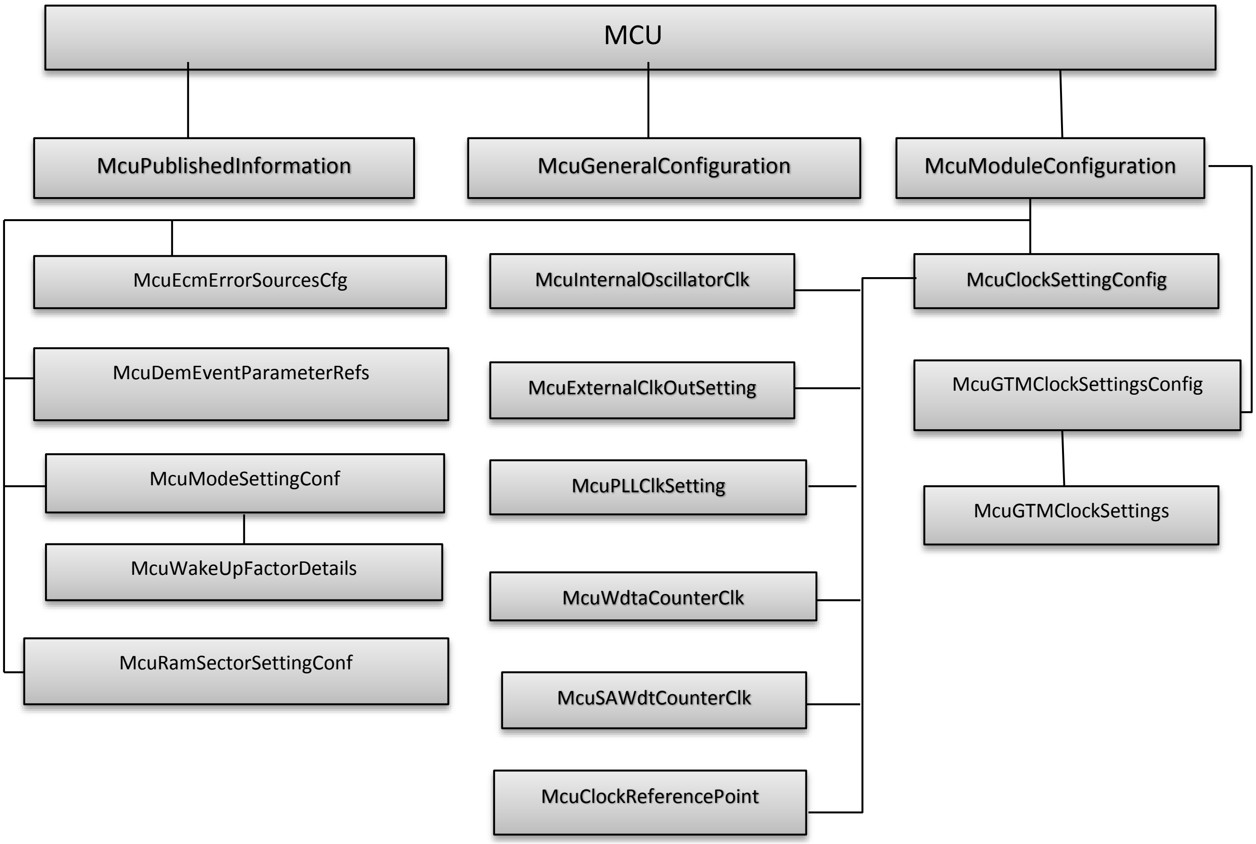

Figure 8-1 Configuration overview

8.1.1 Pre Compile Time Configurable Parameters

Table 8-1 Pre Compile Time Configurable Parameters

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuGeneral	McuVersionCheckExternalModules	Boolean	TRUE / FALSE	Enable / disable AUTOSAR Version check for inter-module dependencies
	McuDevErrorDetect	Boolean	TRUE / FALSE	Pre-processor switch for enabling the development error detection and reporting
	McuGetRamStateApi	Boolean	TRUE / FALSE	Pre-processor switch to enable/disable the API Mcu_GetRamState .
	McuInitClock	Boolean	TRUE / FALSE	Pre-processor switch to enable/disable the API Mcu_InitClock to initialize the clock settings.
	McuNoPll	Boolean	TRUE / FALSE	This parameter shall be set true, if the H/W does not have a PLL or the PLL circuitry is enabled after the power on without S/W intervention. In this case MCU_DistributePllClock has to be disabled and MCU_GetPllStatus has to return MCU_PLL_STATUS_UNDEFINED. Otherwise this parameters has to be set False.
	McuPerformResetApi	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the API to read out the modules version information
	McuVersionInfoApi	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the API to read out the

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
				modules version information
	McuCriticalSectionProtection	Boolean	TRUE / FALSE	This parameter specifies if the MCU driver CPU load can be reduced by disabling the enter/exit critical section functionality by adding a precompiled configuration parameter to the MCU driver configuration.
	McuSwResetCallApi	Boolean	TRUE / FALSE	This parameter shall decide if MCU_RESET_CALLOUT API will be used for to reset software
	McuEcmDelayTimerOverflowValue	Integer	0-65535	This parameter specifies to configure the overflow value for the ECM delay timer
	McuEcmErrorOutputMode	Enum	NON_DYNAMIC_MODE DYNAMIC_MODE	This parameter used to configure the error output either in Dynamic or Non-Dynamic mode
	McuClm0Operation	Boolean	TRUE / FALSE	This parameter enables or disables operation of clock monitor function for CLMA0
	McuClm1Operation	Boolean	TRUE / FALSE	This parameter enables or disables operation of clock monitor function for CLMA1
	McuClm2Operation	Boolean	TRUE / FALSE	This parameter enables or disables operation of clock monitor function for CLMA2
	McuClm3Operation	Boolean	TRUE / FALSE	This parameter enables or disables operation of clock monitor function for CLMA3

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
	McuClm4Operation	Boolean	TRUE / FALSE	This parameter enables or disables operation of clock monitor function for CLMA4
	McuClm0MonitoringClock Accuracy	Float	0-20	This parameter specifies the monitoring clock accuracy of CLMA0 in percentage.
	McuClm1MonitoringClock Accuracy	Float	1-20	This parameter specifies the monitoring clock accuracy of CLMA1 in percentage.
	McuClm2MonitoringClock Accuracy	Float	2-20	This parameter specifies the monitoring clock accuracy of CLMA2 in percentage.
	McuClm3MonitoringClock Accuracy	Float	0-20	This parameter specifies the monitoring clock accuracy of CLMA3 in percentage.
	McuClm4MonitoringClock Accuracy	Float	0-20	This parameter specifies the monitoring clock accuracy of CLMA4 in percentage.
	McuClm0SamplingClock Accuracy	Float	6-26	This parameter specifies the sampling clock accuracy of CLMA0 in percentage
	McuClm1SamplingClock Accuracy	Float	3-20	This parameter specifies the sampling clock accuracy of CLMA1 in percentage
	McuClm2SamplingClock Accuracy	Float	5-20	This parameter specifies the sampling clock accuracy of CLMA2 in percentage
	McuClm3SamplingClock Accuracy	Float	2-20	This parameter specifies the sampling clock

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
				accuracy of CLMA3 in percentage
	McuClm4SamplingClockAccuracy	Float	2-20	This parameter specifies the sampling clock accuracy of CLMA4 in percentage
	McuLoopCount	Integer	1-255	The count value for the feed-back loop in Mcu module.
	McuEcmRstConfigure	Enum	SYSTEM_RESE APPLICATION_RESET	This parameter is used to configure the type of Ecm Reset. SYSTEM_RESET: ECM Module will generate System Reset. APPLICATION_RESET: ECM Module will generate Application Reset.
	McuDeviceName	Enum	R7F701370A, R7F701371, R7F701372, R7F701373, R7F701374	This parameter contains the supported device name.
	McuGTMConfigurationSupport	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the complete support of all GTM related functionality.
	McuInterruptConsistencyCheck	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the Interrupt Consistency Check functionality
	McuUseWriteVerifyErrorInterface	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the user can configure error notification interface instead of DEM error for Write-Verify functionality.
	McuWriteVerifyErrorInterface	function	Null/WriteVerify	Pre-processor switch to enable or disable the parameter is used to configure the name of the error

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
				notification which shall be called with the ApiID and ErrorId when there is a Write Verify error.
	McuWriteVerify	Enum	WV_DISABLE WV_INIT_ONLY WV_INIT_RUNTIME	This parameter enables selection and support of WV_INIT_ONLY or WV_INIT_RUNTIME register write-verify functionality of registers.
	McuClma0SelfDiagnosticTest	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the user can configure CLMA0 Self Diagnosis Support.
	McuClma1SelfDiagnosticTest	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the user can configure CLMA1Self DiagnosisSupport.
	McuClma2SelfDiagnosticTest	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the user can configure CLMA2 Self DiagnosisSupport.
	McuClma3SelfDiagnosticTest	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the user can configure CLMA3 Self DiagnosisSupport.
	McuClma4SelfDiagnosticTest	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the user can configure CLMA4 Self DiagnosisSupport.
	McuCvmSelfDiagnosticTest	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the user can configure CVM Self Diagnosis Support

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
	McuEcmSelfDiagnosticTest	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable the user can configure ECM Self Diagnosis Support
	McuLockStepSelfDiagnosticTest	Boolean	TRUE / FALSE	Pre-processor switch to enable or disable Lockstep Self Diagnosis Support

8.1.2 Post Build Time Configurable Parameters

Table 8-2 Post Build Time Configurable Parameters

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuModuleConfiguration	McuClockSrcFailureNotification	Enum	DISABLED ENABLED	This parameter enables or disables clock failure notification. In case this feature is not supported by HW the setting should be disabled. This parameter is not used for implementation
	McuNumberOfMcuModes	Integer	1-255	This parameter shall represent the number of Modes available for the MCU. This parameter is not used for implementation
	McuRamSectors	Integer	1-4294967295	This parameter shall represent the number of RAM sectors available for the MCU. This parameter is not used for implementation.
	McuResetSetting	Integer	1-255	This parameter relates to the MCU specific reset configuration. This applies to the function Mcu_PerformReset, which performs a microcontroller reset using the hardware feature of the microcontroller. This parameter is not used for implementation
	McuCvmOutMaskFbist	Boolean	TRUE / FALSE	This parameter enables or disables the CVMOUT Output Mask for FBIST Operation
	McuCvmOutMaskDiag	Boolean	TRUE / FALSE	This parameter enables or disables the CVMOUT Output Mask for DIAG Operation
	McuCvmOutputFilter	Boolean	TRUE / FALSE	This parameter enables or disables the CVMOUT Output Filter Control
	McuCvmDiagLockBit	Boolean	TRUE / FALSE	This parameter lock or unlock the CVM Diagnosis during normal operation.
	McuCvmResetEnable	Boolean	TRUE / FALSE	This parameter enable or disable a reset of core voltage operation area upon detection of abnormal in core power supply.

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
	McuSwResetTrigger	Enum	SW_SYSTEM_RESET SW_APPLICATION_RESET NO_RESET	This parameter will decide the type of software reset.
McuClockSetting Config	McuClockSettingId	Integer	0-255	The Id of this McuClockSettingConfig to be used as argument for the API call "Mcu_InitClock".It should be unique and consecutive over multiple ClockSetting configurations.
	McuMainOsciFrequency	Enum	FREQ_16000000 FREQ_20000000 FREQ_24000000	This parameter selects the main oscillator frequency in Hz
	McuCpuMainSysClk	Integer	120000000-240000000	This parameter contains the CPU main system clock frequency. This parameter does not affect the behaviour of the MCU driver. It is a frequency that will be available to other modules for reference. It either needs to be calculated manually or by the configuration editor
	McuUnitName	Enum	CLK_CPUM	This parameter contains the functional unit name for the CPU.
McuInternalOscillatorClk	McuPeripheralClock	Integer	16000000	This parameter contains the Internal Oscillator Clock value.
McuExternalClkOutSetting	McuExternalClk0SourceSel	Enum	MCU_MAIN_OSCILLATOR MCU_CLK_LSB MCU_CLK_CPU MCU_BACKUP_CLOCK	This parameter is to select the external clock 0 source. MCU_MAIN_OSCILLATOR: 16 to 24 MHz MainOSC. MCU_CLK_LSB (Low Speed System Clock): 30 to 40 MHz MCU_CLK_CPU: 120 to 240 MHz clock. MCU_BACKUP_CLOCK: 8 MHz.
	McuExternalClk1SourceSel	Enum	MCU_MAIN_OSCILLATOR MCU_CLK_LSB MCU_CLK_CPU MCU_BACKUP_CLOCK	This parameter is to select the external clock 1 source. MCU_MAIN_OSCILLATOR: 16 to 24 MHz MainOSC. MCU_CLK_LSB: Low Speed System Clock. If MCU_CLK_CPU is 240 MHz or 160 MHz, MCU_CLK_LSB is 40 MHz.

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
				If MCU_CLK_CPU is 120 MHz, MCU_CLK_LSB is 30 MHz MCU_CLK_CPU: 120 to 240 MHz clock. MCU_BACKUP_CLOCK: Internal oscillator clock * 1/2.
	McuExternalClk0DividerSel	Integer	1-1023	This parameter is to select the divider for external clock 0 source
	McuExternalClk1DividerSel	Integer	1-1023	This parameter is to select the divider for external clock 1 source
	McuExternalClock0	Integer	250000- 20000000	This parameter contains the output of External Clock 0 value.
	McuExternalClock1	Integer	250000- 20000000	This parameter contains the output of External Clock 1 value.
McuPLLClkSetting	McuSystemSourceSel	Enum	MCU_PLL MCU_INTERNAL_OSCILLATOR	This parameter is to select the System Clock source 0. MCU_PLL: 320 to 480 MHz Clock. MCU_INTERNAL_OSCILLATOR: 16 MHz IntOSC.
	McuPLLClk0DividerSel	Integer	1-7	This parameter is to select the divider for PLL0 source.
	McuPLLClk1DividerSel	Integer	1-7	This parameter is to select the divider for PLL1 source.
	McuPLLClock0	Integer	320000000-480000000	This parameter contains the PLL0 Clock value.
	McuOPBT1Sel	Integer	0-4294967295	This parameter contains the value that is to be written in OPBT1.
McuSAWdtCounterClk	McuSAWdtCounterDivider	Enum	DIVBY_ID_8 DIVBY_ID_9	This parameter contains the divider for the Secure Application WDTA counter clock DIVBY_ID_8: Secure Application WdtacounterClk/80000 (or) WdtacounterClk/250 is selected as clock divider.Provided SAWDTMD=0/1. DIVBY_ID_9: Secure Application WdtacounterClk/80000 (or) WdtacounterClk/250 is selected as clock divider.Provided SAWDTMD=1/0.

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
	McuUnitName	Enum	WDTCLKI	This parameter contains the functional unit name for the Secure Application Wdta counter Clock.
	McuPeripheralClock	Integer	2000- 640000	This parameter contains the Wdta counter Clock.
McuWdtaCounterClk	McuWdtCounterDivider	Enum	DIVBY_1_ID_1 DIVBY_32_ID_1	This parameter contains the divider for the WDTA counter clock DIVBY_1_ID_1: WdtacounterClk/1 is selected as clock divider. DIVBY_32_ID_1: WdtacounterClk/32 is selected as clock divider
	McuUnitName	Enum	WDTCLKI	This parameter contains the functional unit name for the Wdta counter Clock.
	McuPeripheralClock	Integer	250000- 8000000	This parameter contains the Wdta counter Clock.
McuClockReferencePoint	McuClockReferencePointFrequency	Float	0- 4294967295	This is the frequency for the specific instance of the McuClockReferencePoint container. It shall be given in Hz. This parameter is not used for implementation. So if it is used for reference frequency then user has to take care for configuration of valid clock domain associated with it
McuGTMClockSettings	McuGTMClockSelection	Enum	CMU_CLK0 CMU_CLK1 CMU_CLK2 CMU_CLK3 CMU_CLK4 CMU_CLK5 CMU_CLK6 CMU_CLK7	This parameter used to configure the CMU clocks. CMU_CLK0: CMU clock 0. CMU_CLK1: CMU clock 1. CMU_CLK2: CMU clock 2. CMU_CLK3: CMU clock 3. CMU_CLK4: CMU clock 4. CMU_CLK5: CMU clock 5. CMU_CLK6: CMU clock 6. CMU_CLK7: CMU clock 7.
	McuGTMChannelClkSrcDivider	Integer	1-16777215	This parameter holds the Channel Clock Source Divider.
McuGTMClockSettingsConfig	McuGTMCMUGCLKNumerator	Integer	1-16777215	This parameter holds the Numerator for the CMU Global Clock Divider.
	McuGTMCMUGCLKDenominator	Integer	1-16777215	This parameter holds the Denominator for the CMU Global Clock Divider.

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuEcmErrorSourcesCfg0 to McuEcmErrorSourcesCfg92	McuEcmErrorMaskableInterrupt	Boolean	TRUE / FALSE	This parameter enables or disables generation of a maskable interrupt when the error occurs from each source
	McuEcmErrorNonMaskableInterrupt	Boolean	TRUE / FALSE	This parameter enables or disables generation of a non maskable interrupt when the error occurs from each error source
	McuEcmErrorInternalReset	Boolean	TRUE / FALSE	This parameter enables or disables generation of an internal reset when the error occurs from each error source
	McuEcmErrorNMIDelayTimer	Boolean	TRUE / FALSE	This parameter enables or disables delay timer start caused by non-maskable interrupts in response to errors from each error source
	McuEcmErrorMIDelayTimer	Boolean	TRUE / FALSE	This parameter enables or disables delay timer start caused by maskable interrupts in response to errors from each error source
	McuEcmErrorOutputMask	Boolean	TRUE / FALSE	This parameter mask or unmask the individual error sources of the error pin output from each error source
	McuMiNotification	Function	NA	This parameter contains notification function for MI notification
	McuNmiNotification	Function	NA	This parameter contains notification function for NMI notification.
	McuEcmInitialNotification	function	NA	This parameter is used for notify the upper layer,if any ECM error has been already detected at system start-up.
McuEcmErrorSourcesCfg93	McuEcmErrorInternalReset	Boolean	TRUE / FALSE	This parameter enables or disables generation of an internal reset when the ecm delay timer overflow error occurs.
	McuEcmErrorOutputMask	Boolean	TRUE / FALSE	This parameter mask or unmask the individual error sources of the error pin output when the ecm delay timer overflow error occurs.
	McuEcmInitialNotification	function	NA	This parameter is used for notify the upper layer,if any ECM error has been

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
				already detected at system start-up.
McuDemEventParameterRefs	MCU_E_CLOCK_FAILURE	Reference	Reference to DemEventId of Dem module.	Reference to configured DEM event to report "Clock source failure".
	MCU_E_WRITE_TIMEOUT_FAILURE	Reference	Reference to DemEventId of Dem module.	Reference to configured DEM event to report "write failure".
	MCU_E_POWERDOWN_MODE_FAILURE	Reference	Reference to DemEventId of Dem module.	Reference to configured DEM event to report "Power Down Mode failure".
	MCU_E_INT_INCONSISTENT	Reference	Reference to DemEventId of Dem module.	Reference to the DemEventParameter which shall be issued when interrupt inconsistency is detected.
	MCU_E_REG_WRITE_VERIFY	Reference	Reference to DemEventId of Dem module.	Reference to the DemEventParameter which shall be issued when a register write-verify failure was detected. If the reference is not configured the error shall not be reported.
	MCU_E_CLM_SELFDIAG_FAILURE	Reference	Reference to DemEventId of Dem module.	Reference to the configured DEM event to report “CLM self diagnostic failure”
	MCU_E_ECM_SELFDIAG_FAILURE	Reference	Reference to DemEventId of Dem module.	Reference to the configured DEM event to report “ECM self diagnostic failure”
	MCU_E_CVM_SELFDIAG_FAILURE	Reference	Reference to DemEventId of Dem module.	Reference to the configured DEM event to report “CVM self diagnostic failure”
	MCU_E_LOCKSTEP_SELFDIAG_FAILURE	Reference	Reference to DemEventId of Dem module.	Reference to the configured DEM event to report “LockStep self diagnostic failure”
McuModeSettingConf	McuMode	Integer	1-255	The parameter represents the MCU Mode settings. This parameter is not used for implementation.
	McuModeType	Enum	MCU_HALT_MODE MCU_MODULE_STANDBY_MODE	This parameter contains the power save mode type
	McuMcanStopTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the stop mode triggering of MCAN domain

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
	McuFlexrayStopTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the stop mode triggering of FLEXRAY domain
	McuGtmStopTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the stop mode triggering of GTM domain
	McuEthernetStopTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the stop mode triggering of ETHERNET domain
	McuRsentStopTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the stop mode triggering of RSENT domain
	McuHsUsrtStopTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the stop mode triggering of HS_USRT domain
	McuCsihStopTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the stop mode triggering of CSIH domain
	McuRlin3StopTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the stop mode triggering of RLIN3 domain
	McuAdcStopTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the stop mode triggering of ADC domain
	McuMcanWakeUpTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the Wakeup triggering of MCAN domain
	McuFlexrayWakeupTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the Wakeup triggering of FLEXRAY domain
	McuGtmWakeupTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the Wakeup triggering of GTM domain
	McuEthernetWakeupTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the Wakeup triggering of ETHERNET domain
	McuRsentWakeupTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the Wakeup triggering of RSENT domain
	McuHsUsrtWakeupTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the Wakeup triggering of HS_USRT domain
	McuCsihWakeupTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the Wakeup triggering of CSIH domain

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
	McuRlin3WakeupTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the Wakeup triggering of RLIN3 domain
	McuAdcWakeupTrigger	Boolean	TRUE / FALSE	This parameter will enables or disables the Wakeup triggering of ADC domain
McuWakeUpFactorDetails	McuWakeUpFactorName	Enum	INTECMMI INTIPIR0 INTIPIR1 INTICUP INTWDTA INTSW0 INTP0 INTP1 INTP2 INTP3 INTP4 INTOTS0OTI INTOTS0OTULI INTSTM00 INTSTM01 INTGTM0TIM00 INTGTM0TIM02 INTGTM0MCS00 INTGTM0MCS02 INTADCF012 INTCSIH0TIR INTCSIH1TIR INTRLIN33UR2 INTP5 INTP6 INTP7 INTP8 INTP9 INTGTM0TIM01 INTGTM0TIM03 INTMTTCANFE INTMCAN0FE INTETNA0	This parameter contains wake up factor name for PE/Peripherals

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
			INTFLX0LINE0 INTFLX1LINE0 INTSENT0RI INTSENT1RI INTDNFA2WUF0 INTDNFA2WUF1 INTDNFA3WUF0 INTDNFA3WUF1 INTDNFA4WUF0 INTDNFA4WUF1 FEINT FENMISWDTA	
McuRamSectorSettingConf	McuRamDefaultValue	Integer	0-255	This parameter shall represent the Data pre-setting to be initialized
	McuRamSectionBaseAddress	Integer	4275961856-4277829631	This parameter shall represent the MCU RAM section base address. For LOCAL RAM 128KB having range 1 to 131072. For VARIABLE RAM 192KB having range 1 to 196608. For GLOBAL RAM BANK A 480KB having range 1 to 491520. For GLOBAL RAM BANK B 480KB having range 1 to 491520. For VARIABLE RAM 192KB having range 1 to 196608
	McuRamSectionSize	Integer	1-65536	This parameter shall represent the MCU RAM Section size in bytes. For LOCAL RAM 128KB having range 1 to 131072. For VARIABLE RAM 192KB having range 1 to 196608. For GLOBAL RAM BANK A 480KB having range 1 to 491520. For GLOBAL RAM BANK B 480KB having range 1 to 491520. For VARIABLE RAM 192KB having range 1 to 196608.

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuPublishedInformation	McuResetReason	Integer	0	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module. This parameter is not used for implementation.
McuRstRsnConfPowOnRst	McuResetReason	Integer	0	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_POWER_ON_RESET
McuRstRsnConfTerminalRst	McuResetReason	Integer	1	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_TERMINAL_RESET
McuRstRsnConfCvmRst	McuResetReason	Integer	2	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CVM_RESET
McuRstRsnConfSwSysRst	McuResetReason	Integer	3	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_SW_SYS_RESET
McuRstRsnConfWdtRst	McuResetReason	Integer	4	This container contains the configuration for the different type of reset reason that can be retrieved from Mcu_GetResetReason API.

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuRstRsnConfLockStepCoreRst	McuResetReason	Integer	5	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_LOCK_STEP_CORE_RST
McuRstRsnConfPbusFssRst	McuResetReason	Integer	6	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_PBUS_FSS_RST
McuRstRsnConfBusBridgeErrorRst	McuResetReason	Integer	7	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_BUS_BRIDGE_ERROR_RST
McuRstRsnConfSafetyMechCompRst	McuResetReason	Integer	8	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_SAFETY_MECH_COMP_RST
McuRstRsnConfTempSensorRst	McuResetReason	Integer	9	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_TEMPERATURE_SENSOR_RST
McuRstRsnConfClma0Rst	McuResetReason	Integer	10	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
				module, MCU_CLMA0_RST
McuRstRsnConfClma2Rst	McuResetReason	Integer	11	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CLMA2_RST
McuRstRsnConfClma3Rst	McuResetReason	Integer	12	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CLMA3_RST
McuRstRsnConfClma1Rst	McuResetReason	Integer	14	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CLMA1_RST
McuRstRsnConfLRamEccDedRst	McuResetReason	Integer	15	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_LRAM_ECC_DED_RST
McuRstRsnConfGRamEccDedRst	McuResetReason	Integer	16	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_GRAM_ECC_DED_RST

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuRstRsnConfCacheRamEdcRst	McuResetReason	Integer	17	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CACHE_RAM_EDC_RST
McuRstRsnConfCodeFlsEccDedRst	McuResetReason	Integer	18	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CODE_FLS_ECC_DED_RST
McuRstRsnConfDataFlsEccDedRst	McuResetReason	Integer	19	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_DATA_FLS_ECC_DED_RST
McuRstRsnConfCsihRamEccDedRst	McuResetReason	Integer	20	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CSIH_RAM_ECC_DED_RST
McuRstRsnConfCanRamEccDedRst	McuResetReason	Integer	21	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CAN_RAM_ECC_DED_RST

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuRstRsnConfEthRamDedRst	McuResetReason	Integer	22	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_ETH_RAM_ECC_DED_RST
McuRstRsnConfFrRamEccDedRst	McuResetReason	Integer	23	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_FR_RAM_ECC_DED_RST
McuRstRsnConfGtmRamEccDedRst	McuResetReason	Integer	24	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_GTM_RAM_ECC_DED_RST
McuRstRsnConfBusEccDedRst	McuResetReason	Integer	25	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_BUS_ECC_DED_RST
McuRstRsnConfBusEccSedRst	McuResetReason	Integer	26	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_BUS_ECC_SED_RST

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuRstRsnConfLRamAddrOvfRst	McuResetReason	Integer	27	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_LRAM_ADDR_OVF_RST
McuRstRsnConfGRamAddrOvfRst	McuResetReason	Integer	28	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_GRAM_ADDR_OVF_RST
McuRstRsnConfCodeFlsAddrOvfRst	McuResetReason	Integer	29	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CODE_FLS_ADDR_OVF_RST
McuRstRsnConfDataFlsAddrOvfRst	McuResetReason	Integer	30	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_DATA_FLS_ADDR_OVF_RST
McuRstRsnConfPeriRamEccAddrOvfRst	McuResetReason	Integer	31	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_PERI_RAM_ECC_ADDR_OVF_RST

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuRstRsnConfDtsRamEccDedRst	McuResetReason	Integer	32	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_DTS_RAM_ECC_DED_RST
McuRstRsnConfDtsRamEccSedRst	McuResetReason	Integer	33	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_DTS_RAM_ECC_SED_RST
McuRstRsnConfLRamEccSedRst	McuResetReason	Integer	34	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_LRAM_ECC_SED_RST
McuRstRsnConfGRamEccSedRst	McuResetReason	Integer	35	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_GRAM_ECC_SED_RST
McuRstRsnConfCodeFlsEccSedRst	McuResetReason	Integer	36	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CODE_FLS_ECC_SED_RST

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuRstRsnConfDataFlsEccSedRst	McuResetReason	Integer	37	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_DATA_FLS_ECC_SED_RST
McuRstRsnConfCsihRamEccSedRst	McuResetReason	Integer	38	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CSIH_RAM_ECC_SED_RST
McuRstRsnConfCanRamEccSedRst	McuResetReason	Integer	39	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_CAN_RAM_ECC_SED_RST
McuRstRsnConfEthRamEccSedRst	McuResetReason	Integer	40	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_ETH_RAM_ECC_SED_RST
McuRstRsnConfFrRamEccSedRst	McuResetReason	Integer	41	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_FR_RAM_ECC_SED_RST

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuRstRsnConfGtmRamEccSedRst	McuResetReason	Integer	42	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_GTM_RAM_ECC_SED_RST
McuRstRsnConfPEGuardRst	McuResetReason	Integer	43	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_PE_GUARD_RST
McuRstRsnConfGramGuardRst	McuResetReason	Integer	44	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_GRAM_GUARD_RST
McuRstRsnConfMemcGuardRst	McuResetReason	Integer	45	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_MEMC_GUARD_RST
McuRstRsnConfSlaveGuardRst	McuResetReason	Integer	46	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_SLAVE_GUARD_RST
McuRstRsnConfCodeFlsPEUnmapAccessRst	McuResetReason	Integer	47	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
				module, MCU_CODE_FLS_PE_UNMAP_ACCESS_RST
McuRstRsnConfGramPEUnmapAccessRst	McuResetReason	Integer	48	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_GRAM_PE_UNMAP_ACCESS_RST
McuRstRsnConfLpbPEUnmapAccessRst	McuResetReason	Integer	49	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_LPB_PE_UNMAP_ACCESS_RST
McuRstRsnConfPBusUnmapAccessRst	McuResetReason	Integer	50	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_PBUS_UNMAP_ACCESS_RST
McuRstRsnConfHBusUnmapAccessRst	McuResetReason	Integer	51	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_HBUS_UNMAP_ACCESS_RST
McuRstRsnConfCodeFlsGvciUnmapAccessRst	McuResetReason	Integer	52	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module,

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
				MCU_CODE_FLS_GVCI_UNMAP_ACCESS_RST
McuRstRsnConfGramFlsGvciUnmapAccessRst	McuResetReason	Integer	53	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_GRAM_FLS_GVCI_UNMAP_ACCESS_RST
McuRstRsnConfResHbusUnmapAccessRst	McuResetReason	Integer	54	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_RES_HBUS_UNMAP_ACCESS_RST
McuRstRsnConfDmaTransferRst	McuResetReason	Integer	55	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_DMA_TRANSFER_RST
McuRstRsnConfDmaUnmappedRst	McuResetReason	Integer	56	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_DMA_UNMAPPED_RST
McuRstRsnConfFlsSequenceRst	McuResetReason	Integer	57	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module,

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
				MCU_FLS_SEQUENCE_RST
McuRstRsnConfFlsFaciRst	McuResetReason	Integer	58	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_FLS_FACI_RST
McuRstRsnConfAdcParityRst	McuResetReason	Integer	59	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_ADC_PARITY_RST
McuRstRsnConfPEUnintenEnableDisableRst	McuResetReason	Integer	60	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_PE_UNINTEN_EN_DIS_RST
McuRstRsnConfUnintenDeactUserRst	McuResetReason	Integer	61	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_UNINTEN_DEACT_USR_RST
McuRstRsnConfUnintenActCfpModeRst	McuResetReason	Integer	62	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_UNINTEN_ACT_CFP_MODE_RST

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
McuRstRsnConfUnintenDebugEnableDetectRst	McuResetReason	Integer	63	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_UNINTEN_DEBUG_EN_DET_RST
McuRstRsnConfUnintenActTestModeRst	McuResetReason	Integer	64	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_UNINTEN_ACT_TESTMODE_RST
McuRstRsnConfEcmCompRst	McuResetReason	Integer	65	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_ECM_COMP_RST
McuRstRsnConfDebuggerRst	McuResetReason	Integer	66	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_DEBUGGER_RESET
McuRstRsnConfSwApplRst	McuResetReason	Integer	67	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_SW_APPL_RESET
McuRstRsnConfBistRst	McuResetReason	Integer	68	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager

Container Name	Parameter Name	Parameter Type	Parameter Range	Parameter Description
				module, MCU_BIST_RESET
McuRstRsnConfRstUndefined	McuResetReason	Integer	69	The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_RESET_UNDEFINED
McuRstRsnConfRstUnknown	McuResetReason	Integer	70	The parameter represents the different type of reset that a Microcontroller supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module, MCU_RESET_UNKNOWN

Chapter 9 Messages

The messages help to identify the syntax or semantic errors in the ECU Configuration Description File. Hence it ensures validity and correctness of the information available in the ECU Configuration Description File.

The following section gives the list of error, warning and information messages displayed by the MCAL Code Generator Tool.

9.1 Error Messages

ERR_59_101_002: McuClockSettingConfig container short name between configsets should be same for each Clocksetting ID in path Mcu/McuModuleConfiguration/McuClockSettingConfig. For ClocksettingId $Id, the container short name is different between configset $ConfigSetCount and $MaxConfigSets.

This error will occur when McuClockSettingConfig container short name between configsets is not same for each Clocksetting ID in path Mcu/McuModuleConfiguration/McuClockSettingConfig.

ERR_59_101_003: McuClockSettingId value in the container path Mcu/McuModuleConfiguration/McuClockSettingConfig is out of range in configset $ConfigSetCount. It should be configured less than $McuInstance."

This error will occur when McuClockSettingId in the container path Mcu/McuModuleConfiguration/McuClockSettingConfig is out of range in a configset

ERR_59_101_004: McuClockSettingId value = $Id in the container path Mcu//McuModuleConfiguration/McuClockSettingConfig is repeating in configset $ConfigSetCount**.**

This error will occur when McuClockSettingId in the container path Mcu/McuModuleConfiguration/McuClockSettingConfig is repeating in a configset

ERR_59_101_005: McuClockSettingId value = $Id in the container path Mcu/McuModuleConfiguration/McuClockSettingConfig is not configured in configset $ConfigSetCount

This error will occur when McuClockSettingId in the container path Mcu/McuModuleConfiguration/McuClockSettingConfig is not configured in a configset

ERR_59_101_006: Reference path configured for the parameter MCU_E_CLOCK_FAILURE is not correct in the path <Mcu/McuModuleConfiguration/McuDemEventPrameterRefs

This error will occur when reference path configured for the parameter MCU_E_CLOCK_FAILURE is not correct in the path

ERR_59_101_007: Reference path configured for the parameter MCU_E_WRITE_TIMEOUT_FAILURE is not correct in the path <Mcu/McuModuleConfiguration/McuDemEventPrameterRefs.

This error will occur when reference path configured for the parameter MCU_E_WRITE_TIMEOUT_FAILURE is not correct in the path

ERR_59_101_008: Value configured for the parameter McuModeType is repeating in the path Mcu/McuModuleConfiguration/McuModeSettingConf. It should be unique within configset.

This error will occur when value configured for the parameter McuModeType is repeating in the path Mcu/McuModuleConfiguration/McuModeSettingConf

ERR_59_101_009: Value configured for the parameter McuWakeUpFactorName in the path Mcu/McuModuleConfiguration/ McuModeSettingConf0/ McuWakeUpFactorDetails should not be repeated within the McuWakeUpFactorDetails container

This error will occur when value configured for the parameter McuWakeUpFactorName in the path Mcu/McuModuleConfiguration/ McuModeSettingConf0/ McuWakeUpFactorDetails is repeated within the McuWakeUpFactorDetails container

ERR_59_101_010: Reference path configured for the parameter MCU_E_POWERDOWN_MODE_FAILURE is not correct in the path <Mcu/McuModuleConfiguration/McuDemEventPrameterRefs

This error will occur when reference path configured for the parameter MCU_E_POWERDOWN_MODE_FAILURE is not correct in the path

ERR_59_101_012: WakeUpTrigger of a particular domain cannot be enabled when StopTrigger of the same domain is disabled in the path <Mcu/McuModuleConfiguration/McuModeSettingConf>.

This error will occur when WakeUpTrigger is expected from a particular domain when StopTrigger of the same domain is disabled in the path

ERR_59_101_013: McuGTMCMUClock = $McuCMUClk is repeating in McuGTMClockSettings container in configset $ConfigSetCount.

This error will occur when the McuGTMCMUClock is repeating in McuGTMClockSettings container in configsets

ERR_59_101_014: This value of OPBT1.PLL0FREQ = $McuPLLFreq is not supported for P1H-C devices.

This error will occur when the OPBT1.PLL0FREQ is not supported for P1H-C devices.

ERR_59_101_015: The configured value of PLL clock should be equal to $McuReqPLLClock MHz.

This error will occur when the configured value of PLL clock is not equal to $McuReqPLLClock MHz.

ERR_59_101_016: The configured value of McuOPBT1Sel = $McuOPBT1Sel is not supported.

This error will occur when the configured value of McuOPBT1Sel is not supported.

ERR_59_101_017: The value of the parameter McuEcmDelayTimerOverflowValue = $McuEcmDelTimOverVal Hz is out of range.

This error will occur when the value of McuEcmDelayTimerOverflowValue is not in range.

ERR_59_101_018: The value of the parameter McuClm0MonitoringClockAccuracy = $McuClm0MonClockAcc Hz is out of range.

This error will occur when the value of the McuClm0MonitoringClockAccuracy is not in range.

ERR_59_101_019: The value of the parameter McuClm0SamplingClockAccuracy = $McuClm0SamClockAcc Hz is out of range.

This error will occur when the value of the McuClm0SamplingClockAccuracy is not in range.

ERR_59_101_020: The value of the parameter McuClm1MonitoringClockAccuracy = $McuClm1MonClockAcc Hz is out of range.

This error will occur when the value of the McuClm1MonitoringClockAccuracy is not in range.

ERR_59_101_021: The value of the parameter McuClm1SamplingClockAccuracy = $McuClm1SamClockAcc Hz is out of range.

This error will occur when the value of the McuClm1SamplingClockAccuracy is not in range.

ERR_59_101_022: The value of the parameter McuClm2MonitoringClockAccuracy = $McuClm2MonClockAcc Hz is out of range.

This error will occur when the value of the McuClm2MonitoringClockAccuracy is not in range.

ERR_59_101_023: The value of the parameter McuClm2SamplingClockAccuracy = $McuClm2SamClockAcc Hz is out of range.

This error will occur when the value of the McuClm2SamplingClockAccuracy is not in range.

ERR_59_101_024: The value of the parameter McuClm3MonitoringClockAccuracy = $McuClm3MonClockAcc Hz is out of range.

This error will occur when the value of the McuClm3MonitoringClockAccuracy is not in range

ERR_59_101_025: The value of the parameter McuClm3SamplingClockAccuracy = $McuClm3SamClockAcc Hz is out of range.

This error will occur when the value of the McuClm3SamplingClockAccuracy is not in range

ERR_59_101_028: The value of the parameter McuLoopCount = $McuLoopCt Hz is out of range.

This error will occur when the value of the McuLoopCount is not in range.

ERR_59_101_029: The value of the parameter McuMainOsciFrequency is invalid in McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuMainOsciFrequency is not in range.

ERR_59_101_030: The value of the parameter McuCpuMainSysClk = $McuCpuClk Hz is out of range in McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuCpuMainSysClk is not in range.

ERR_59_101_031: The value of the parameter McuPeripheralClock = $McuPeriClock Hz is out of range in the container McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuPeripheralClock is not in range.

ERR_59_101_032: The value of the parameter McuExternalClk0DividerSel = $McuExtClk0DivSel Hz is out of range in McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuExternalClk0DividerSel is not in range.

ERR_59_101_033: The value of the parameter McuExternalClock0 = $McuExtClk0 Hz is out of range in McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuExternalClock0 is not in range.

ERR_59_101_034: The value of the parameter McuExternalClk1DividerSel = $McuExtClk1DivSel Hz is out of range McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuExternalClk1DividerSel is not in range.

ERR_59_101_035: The value of the parameter McuExternalClock1 = $McuExtClk1 Hz is out of range in McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuExternalClock1 is not in range.

ERR_59_101_036: The value of the parameter McuPLLClk0DividerSel = $McuPLLClk0DivSel Hz is out of range in McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuPLLClk0DividerSel is not in range.

ERR_59_101_037: The value of the parameter McuPLLClock0 = $McuPLLClock0 Hz is out of range in McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuPLLClock0 is not in range.

ERR_59_101_038: The value of the parameter McuPLLClk1DividerSel = $McuPLLClk1DivSel Hz is out of range in McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuPLLClk1DividerSel is not in range.

ERR_59_101_039: The value of the parameter McuOPBT1Sel = $McuOPBT1Se Hz is out of range in McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuOPBT1Sel is not in range.

ERR_59_101_040: The value of the parameter McuPeripheralClock = $McuPeripClock Hz is out of range in the container McuSAWdtCounterClk in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuPeripheralClock is not in range.

ERR_59_101_041: The value of the parameter McuGTMCMUGCLKNumerator = $McuGTMCMUGCLKNum Hz is out of range in McuGTMClockSettingsConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuGTMCMUGCLKNumerator is not in range.

ERR_59_101_042: The value of the parameter McuGTMCMUGCLKDenominator = $McuGTMCMUGCLKDen Hz is out of range in McuGTMClockSettingsConfig$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuGTMCMUGCLKDenominator is not in range.

ERR_59_101_043: The value of the parameter McuGTMChannelClkSrcDivider = $McuGTMChClkSrcDiv Hz is out of range in McuGTMClockSettings$ClockCount in McuModuleConfiguration$ConfigSetCount.

ERR_59_101_044: The value of the parameter McuRamDefaultValue = $McuRamDefVal Hz is out of range in McuRamSectorSettingConf$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuRamDefaultValue is not in range.

ERR_59_101_045: The value of the parameter McuRamSectionBaseAddress = $McuRamSecBaseAdd Hz is out of range in McuRamSectorSettingConf$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuRamSectionBaseAddress is not in range.

ERR_59_101_046: The value of the parameter McuRamSectionSize = $McuRamSecSize Hz is out of range in McuRamSectorSettingConf$ClockCount in McuModuleConfiguration$ConfigSetCount.

This error will occur when the value of the McuRamSectionSize is not in range.

ERR_59_101_047: The calculated output frequency for the McuExternalClock0 exceeds 20000000 Hz in the container McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount

This error will occur when the value of the McuExternalClock0 exceeds 20000000 Hz.

ERR_59_101_048: The calculated output frequency for the McuExternalClock1 exceeds 20000000 Hz in the container McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount

This error will occur when the value of the McuExternalClock1 exceeds 20000000 Hz.

ERR_59_101_049: The configured value of McuExternalClock0 should be equal to $McuClkVal0 Hz in the container McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount

This error will occur when the value of the McuExternalClock0 is not matching with McuClkVal0.

ERR_59_101_050: The configured value of McuExternalClock1 should be equal to $McuClkVal1 Hz in the container McuClockSettingConfig$ClockCount in McuModuleConfiguration$ConfigSetCount

This error will occur when the value of the McuExternalClock1 is not matching with McuClkVal1.

ERR_59_101_051: The value of the parameter McuClm4MonitoringClockAccuracy = $McuClm4MonClockAcc Hz is out of range.

This error will occur when the value of the McuClm4MonitoringClockAccuracy is not in range.

ERR_59_101_052: The value of the parameter McuClm4SamplingClockAccuracy = $McuClm4SamClockAcc Hz is out of range.

This error will occur when the value of the McuClm4SamplingClockAccuracy is not in range.

ERR_59_101_053: The parameter 'MCU_E_INT_INCONSISTENT' in the McuDemEventParameterRefs Container should be configured, because the 'McuInterruptConsistencyCheck' in $McuGeneral container is true

This error occur when ''McuInterruptConsistencyCheck ' in McuGeneralConfiguration container is configured as true and MCU_E_INT_INCONSISTENT in McuDemEventParameterRefs is not configured.

ERR_59_101_054: Reference path configured for the parameter MCU_E_INT_INCONSISTENT in McuModuleConfiguration$ConfigSetCount is not correct in the container McuDemEventParameterRefs

This error occur when ''McuInterruptConsistencyCheck ' in McuGeneralConfiguration container is configured as true and MCU_E_INT_INCONSISTENT in McuDemEventParameterRefs is not configured correctly.

ERR_59_101_055: The value configured for the parameter McuWriteVerifyErrorInterface in the container $McuGeneral shall not be Empty or NULL,since McuUseWriteVerifyErrorInterface is configured as True.

This error occur when '' McuUseWriteVerifyErrorInterface ' in McuGeneralConfiguration container is configured as true and McuWriteVerifyErrorInterface is not configured.

ERR_59_101_056: The parameter 'MCU_E_REG_WRITE_VERIFY' in the McuDemEventParameterRefs Container should be configured, because the 'McuWriteVerify' in $McuGeneral container is true.

This error occur when '' McuWriteVerify' in McuGeneralConfiguration container is configured as true and 'MCU_E_REG_WRITE_VERIFY' in McuDemEventParameterRefs is not configured.

ERR_59_101_057: Reference path configured for the parameter MCU_E_REG_WRITE_VERIFY in McuModuleConfiguration$ConfigSetCount is not correct in the container McuDemEventParameterRefs

This error occur when '' McuWriteVerify' in McuGeneralConfiguration container is configured as true and 'MCU_E_REG_WRITE_VERIFY' in McuDemEventParameterRefs is not configured correctly.

ERR_59_101_058: The value of the parameter McuEcmErrorMaskableInterrupt or McuEcmErrorNonMaskableInterrupt should be true for McuEcmErrorSource<16, 17, 48 and 49> since the parameter McuGetRamStateApi is configured as true in the McuGeneralConfiguration container.

This error will occur when the parameters McuEcmErrorMaskableInterrupt or McuEcmErrorNonMaskableInterrupt are not configured as true when the the parameter McuGetRamStateApi is configured as true in the McuGeneralConfiguration container.

Remark As issue raised in Bugzilla: 54536 Autosar parameter McuClockSettingId in McuClockSettingConfig container range is changed to 0 to 255 instead of 1 to 255.

ERR_59_101_059: The value of the parameter McuClmaXOperation should be configured as true since the parameter McuClmaxSelfDiagnostic Test is configured as true in the McuGeneralConfiguration container**.**

This error occur when McuClmaxSelfDiagnosticTest in McuGeneralConfiguration container is configured as true and McuClmaXOperation is configured as false.

ERR_59_101_060: The value configured for the parameter MCU_E_CLM_SELFDIAG_FAILURE in the McuDemEventParameterRefs Container in McuModuleConfiguration$ConfigSetCount shall not be empty or NULL when any of the parameters 'McuClma0SelfDiagnosticTest' or 'McuClma1SelfDiagnosticTest' or 'McuClma2SelfDiagnosticTest' or 'McuClma3SelfDiagnosticTest' or 'McuClma4SelfDiagnosticTest' in 'McuGeneralConfiguration' is configured as true.

For device R7F701373 and R7F701374

The value configured for the parameter MCU_E_CLM_SELFDIAG_FAILURE in the McuDemEventParameterRefs Container in McuModuleConfiguration$ConfigSetCount shall not be empty or NULL when any of the parameters 'McuClma0SelfDiagnosticTest' or 'McuClma0SelfDiagnosticTest' or 'McuClma2SelfDiagnosticTest' or 'McuClma3SelfDiagnosticTest'' in 'McuGeneralConfiguration' is configured as true for P1M-C devices.

This error occur when McuClma<0,1,2,3,4>SelfDiagnosticTest in McuGeneralConfiguration container is configured as true and MCU_E_CLM_SELFDIAG_FAILURE in McuDemEventParameterRefs is not configured.

ERR_59_101_061: Reference path configured for the parameter MCU_E_CLM_SELFDIAG_FAILURE in McuModuleConfiguration$ConfigSetCount is not correct in the container McuDemEventParameterRefs

This error occur when McuClma<0,1,2,3,4>SelfDiagnosticTest in McuGeneralConfiguration container is configured as true and MCU_E_CLM_SELFDIAG_FAILURE in McuDemEventParameterRefs is not configured correctly.

ERR_59_101_062: The parameter MCU_E_CLM_SELFDIAG_FAILURE in the McuDemEventParameterRefs Container in McuModuleConfiguration$ConfigSetCount should be configured, because the parameter McuEcmSelfDiagnosticTest in McuGeneralConfiguration container is configured as true".

This error occur when 'McuEcmSelfDiagnosticTest' in McuGeneralConfiguration container is configured as true and MCU MCU_E_ECM_SELFDIAG_FAILURE in McuDemEventParameterRefs is not configured.

ERR_59_101_063: Reference path configured for the parameter MCU_E_ECM_SELFDIAG_FAILURE in McuModuleConfiguration$ConfigSetCount is not correct in the container McuDemEventParameterRefs

This error occur when 'McuEcmSelfDiagnosticTes'' in McuGeneralConfiguration container is configured and MCU_E_ECM_SELFDIAG_FAILURE in McuDemEventParameterRefs is not configured correctly.

ERR_59_101_064: The parameter MCU_E_CVM_SELFDIAG_FAILURE in the McuDemEventParameterRefs Container in McuModuleConfiguration$ConfigSetCount should be configured, because the parameter 'McuCvmSelfDiagnosticTest' in McuGeneralConfiguration container is configured as true"

This error occur when 'McuCvmSelfDiagnosticTest' in McuGeneralConfiguration container is configured as true and MCU_E_CLM_SELFDIAG_FAILURE in McuDemEventParameterRefs is not configured.

ERR_59_101_065: Reference path configured for the parameter MCU_E_CVM_SELFDIAG_FAILURE in McuModuleConfiguration$ConfigSetCount is not correct in the container McuDemEventParameterRefs

This error occur when 'McuCvmSelfDiagnosticTest' in McuGeneralConfiguration container is configured and MCU_E_CLM_SELFDIAG_FAILURE in McuDemEventParameterRefs is not configured correctly.

ERR_59_101_066: The parameter MCU_E_LOCKSTEP_SELFDIAG_FAILURE in the McuDemEventParameterRefs Container in McuModuleConfiguration$ConfigSetCount should be configured, because the parameter ' McuLockStepSelfDiagnosticTest' in McuGeneralConfiguration container is configured as true"

This error occur when ' McuLockStepSelfDiagnosticTest' in McuGeneralConfiguration container is configured as true and MCU_E_LOCKSTEP_SELFDIAG_FAILURE in McuDemEventParameterRefs is not configured.

ERR_591_01_067: Reference path configured for the parameter MCU_E_LOCKSTEP_SELFDIAG_FAILURE in McuModuleConfiguration$ConfigSetCount is not correct in the container McuDemEventParameterRefs

This error occur when ' McuLockStepSelfDiagnosticTest ' in McuGeneralConfiguration container is configured and MCU_E_LOCKSTEP_SELFDIAG_FAILURE in McuDemEventParameterRefs is not configured correctly.

ERR_59_101_068: The parameters 'McuEcmErrorMaskableInterrupt' and 'McuEcmErrorNonMaskableInterrupt' are configured as true for the error source McuEcmErrorContainer.

This error occur when McuEcmErrorMaskableInterrupt and McuEcmErrorNonMaskableInterrupt is configured for same error source.

ERR_59_101_069: The parameter 'McuEcmErrorMIDelayTimer' cannot be true since the interrupt 'McuEcmErrorMaskableInterrupt' is configured as false for the same error source McuEcmErrorContainer.

This error occur when McuEcmErrorNMIDelayTimer configured without configuring McuEcmErrorMaskableInterrupt of same error source.

ERR_59_101_070: The parameter 'McuEcmErrorNMIDelayTimer' cannot be true since the interrupt 'McuEcmErrorNonMaskableInterrupt' is configured as false for the same error source McuEcmErrorContainer.

This error occur when McuEcmErrorNMIDelayTimer configured without configuring McuEcmErrorNonMaskableInterrupt of same error source.

9.2 Warning Messages

WARNING_59_101_001: Since the McuModeType parameter is configured as MCU_MODULE_STANBY_MODE, stop trigger of any of the Target domain in the path Mcu/McuModuleConfiguration / McuModeSettingConf need to be enabled.

9.3 Information Messages

INFO_59_101_001: The Time Period of the each GTM CMU clock TCMU_CLK[x] is calculated as,

TCMU_CLK[x] = ((CMUGlobalClockNumerator/CMUGlobalClockDenominator) * TSYS_CLK) * (ChannelClkSrcDivider + 1).

Revision History

Sl.No	Description	Version	Date
1	Initial Version	1.0.0	14-Aug-2015
2	The following changes are made: 1. R number is added in the last page 2. Added McuGTMConfigurationSupport in 8.1.1 3. Added McuGTMClockSettings and McuOPBT1Sel in 8.1.2 4. Updated McuExternalClkOutSetting and McuPLLClkSetting in 8.1.2 5. Added parameter for GTM support in Figure 8-1 Configuration overview. 6. Added error message ERR_59_101_013 , ERR_59_101_014 , ERR_59_101_015, ERR_59_101_016 in 10.1.1 Error Messages 7. Added Information message in Section 10.3 8. Description added for error messages in 10.1.1 Error Messages 9. Compiler version updated in 9.3 User Environment Settings 10. Added Information message INFO_59_101_001 in Section 10.3.	1.0.1	15-Apr-2016
3	Following changes are made 1. The type of the parameters McuClm0MonitoringClockAccuracy,McuClm1MonitoringClockAccuracy, McuClm2MonitoringClockAccuracy, McuClm3MonitoringClockAccuracy, McuClm0SamplingClockAccuracy, McuClm1SamplingClockAccuracy, McuClm2SamplingClockAccuracy, McuClm3SamplingClockAccuracy, are changed from integer to float in section 8.1.1 2. In section 10.1, error messages ERR_59_101_017 to ERR_59_101_067 are added and ERR_59_101_029 is updated. 3. In section 8.1.1, parameters McuClm4MonitoringClockAccuracy, McuClm4SamplingClockAccuracy, McuInterruptConsistencyCheck, McuUseWriteVerifyErrorInterface, McuWriteVerifyErrorInterface, McuWriteVerify, Clma Self Diagnosis, McuCvmSelfDiagnosticTest, McuEcmSelfDiagnosticTest, McuLockStepSelfDiagnosticTest are added. 4. Updated section 8.1.2 post build configuration parameters. 5. Removed Translation XML File from Definition. 6. Updated Chapters 1,3,4,5,6,7 by rephrasing Tool and MCU Driver Generation Tool with MCAL Code Generator Tool 7. Removed Chapter 9 Generation Tool Options, Chapter-10 Notes. 8. Updated Chapter 3 with a remark for common MCAL Code Generator Tool user manual 9. Renamed the Chapter 3 heading as Code Generation Overview. 10. Chapter-4 description rephrased and Flow chart in chapter 3 updated. 11. Updated parameter McuMainOsciFrequency in table 8-2 as enumeration type. 12. ERR_59_101_026 and ERR_59_101_027 are removed.	1.0.2	27-Jan-2017

AUTOSAR MCAL R4.0.3 User's Manual
MCU Driver Component Ver.1.0.2
Generation Tool User's Manual

Publication Date: Rev.1.00, January 27, 2017

Published by: Renesas Electronics Corporation

SALES OFFICES

Renesas Electronics Corporation

http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 4.1

AUTOSAR MCAL R4.0.3

User’s Manual

RENESAS

Renesas Electronics Corporation