User's Manual

CONFIDENTIAL

R-Car V4M AUTOSAR R19-11 MCAL

User’s Manual

Driver Component

Generation Tool User's Manual

RTM8RC779HCMCL5QA0JCDRE
RTM8RC779HCMCL5DA0JCDRE

Target Device:
R-Car V4M

Renesas Electronics
www.renesas.com

Rev.0.12 Jan 2025

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 claims 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, drawings, charts, programs, algorithms, and 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 be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer 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, copying or reverse engineering.
6. 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; 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.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or 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 damage (space system; 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 any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using 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 dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. 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. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a 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. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using 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.
11. 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 comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

List of Abbreviations and Acronyms

Abbreviation	Full Form
API	Application Programming Interface
AUTOSAR	AUTomotive Open System ARchitecture
BSWMDT	Basic Software Module Description Template
CAN	Controller Area Networks
CC	Communication Controller
CDD	Complex Device Driver
DEM	Diagnostic Event Manager
Dio / DIO	Digital Input Output
DMA	Direct Memory Access
e.g.	Example
EB	External Buffer
ECM	Error Control Module
ECU	Electronic Control Unit
ETH	ETHernet
FDL	Data Flash Library
FLS	Flash driver
GPT	General Purpose Timer
HW	Hardware
Hz	Hertz
IB	Internal Buffer
ICCOM	Inter CPU communication
ID/Id	Identifier
IIC	Inter-Integrated Circuit
IPMMU	IP Memory Management Unit
CRC	Cyclic Redundancy Check
EMM	Error Management Module
RFSO	Failure Self-Detection Output
THS	THermal Sensor
IMR	Interrupt Mask Register
INTP	INTerruPt
ISR	Interrupt Service Routine
MBI	Message Buffer Unit
MCAL	Microcontroller Abstraction Layer
MCU	Micro Controller Unit
MHz	Mega Hertz
<Msn>	Module short name
NMI	Non Maskable Interrupt
PCLK	PCLK
PLL	Phase Locked Loop

Abbreviation	Full Form
Rx	Receive
SG	Scan Group
SPAL	Standard Peripheral Abstraction Layer
SPI	Serial Peripheral Interface
SW	Software
Tx	Transmit
TXM	Transmission Mode
<vi>	Vendor ID
VSMD / PDF	Vendor-Specific Module Definition / Parameter Definition File

Abbreviation	Full Form
WDG / Wdg	Watchdog Driver
WDT	Watchdog Timer
XML	eXtensible Mark-up Language
<MICRO_VARIANT>	RCAR
MSIOF	Clock-Synchronized Serial Interface with FIFO

Definitions

Terms	Represented by
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 <MSN> Driver Generation Tool. It is generated by ECU Configuration Editor.
Generation Tool / Tool	This tool is used interchangeably to refer to the <MSN> Driver Generation Tool.
Sl.No.	Serial Number.
Translation XML File	This file contains the translation and device specific header file path.

Table of Contents

List of Tables

1.Introduction

The purpose of this document is described the information related to the Generation Tool. Generation Tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File and generates <MSN> Driver C Source and C Header files.

The <MSN> Driver Component comprises two sections as Embedded Software and the Generation Tool to achieve scalability and configurability.

This document contains information on the options, input and output files of the <MSN> Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

ECU Configuration Description File contains information about <MSN> configuration.
See Chapter 3 to 19 for a more detailed description.

This program is built based on Microsoft .NET Framework.
Refer to the following sites about license in detail.
https://msdn.microsoft.com/en-us/library/ms994405.aspx

1.1 Supported MCAL Product Release Version

The document corresponds to the MCAL Product Release Version and Generation Tool Version shown in Table 1-1 and Table 1-2.

Table 1-1 Supported MCAL Product Release Version

Device	MCAL Product Release Version	No	Component Name	Version
R-Car V4M	Ver19.0.11	1	DIO	1.3.9
		2	PORT	1.1.14
		3	ICCOM	1.1.11
		4	IIC	1.0.13
		5	SPI	1.5.12
		6	FLS	1.1.11
		7	GPT	1.7.12
		8	MCU	1.1.19
		9	WDG	1.4.6
		10	CAN	1.1.17
		11	ETH	1.4.10
		12	THS	1.0.9
		13	IPMMU	1.0.9
		14	EMM	1.0.12
		15	RFSO	1.0.8
		16	CRC	1.0.10

Table 1-2 Supported Generation Tool Version

Device	MCAL Product Release Version	No	Component Name	Version
R-Car V4M	Ver19.0.11	1	DioRCAR.dll	1.0.8
		2	PortRCAR.dll	1.0.16
		3	CddIccomRCAR.dll	1.0.12
		4	CddIicRCAR.dll	1.0.10
		5	SpiRCAR.dll	1.0.9
		6	FlsRCAR.dll	1.0.12
		7	GptRCAR.dll	1.0.10
		8	McuRCAR.dll	1.1.18
		9	WdgRCAR.dll	1.0.6
		10	CanRCAR.dll	1.0.15
		11	EthRCAR.dll	1.1.2
		12	CddThsRCAR.dll	1.0.3
		13	CddIpmmuRCAR.dll	1.0.9
		14	CddEmmRCAR.dll	1.0.13
		15	CddRfsoRCAR.dll	1.0.7
		16	CddCrcRCAR.dll	1.0.6

1.2 Document Overview

The document has been segmented for easy reference. The Table 1-3 provides the user with an overview of the contents of each section:

Table 1-3 Document Overview

Chapter	Contents
Chapter 1 (Introduction)	Provides an introduction to the document and explains how information is organized in this manual.
Chapter 2 (Reference)	Provides a list of documents referred while developing this document.
Chapter 3 (User Configuration Validation)	Describes the user configuration validation performed by the <MSN> Driver Generation Tool.
Chapter 4 (CAN)	This chapter provides user's manual information for CAN driver components.
Chapter 5 (DIO)	This chapter provides user's manual information for DIO driver components.
Chapter 6 (PORT)	This chapter provides user's manual information for PORT driver components.
Chapter 7 (ETH)	This chapter provides user's manual information for ETH driver components.
Chapter 8 (FLS)	This chapter provides user's manual information for FLS driver components.
Chapter 9 (GPT)	This chapter provides user's manual information for GPT driver components.
Chapter 10 (ICCOM)	This chapter provides user's manual information for ICCOM driver components.
Chapter 11 (MCU)	This chapter provides user's manual information for MCU driver components.
Chapter 12 (IIC)	This chapter provides user's manual information for IIC driver components.
Chapter 13 (SPI)	This chapter provides user's manual information for SPI driver components.
Chapter 14 (WDG)	This chapter provides user's manual information for WDG driver components.
Chapter 15 (THS)	This chapter provides user's manual information for THS driver components.
Chapter 16 (IPMMU)	This chapter provides user's manual information for IPMMU driver components.
Chapter 17 (EMM)	This chapter provides user's manual information for EMM driver components.
Chapter 18 (RFSO)	This chapter provides user's manual information for RFSO driver components.
Chapter 19 (CRC)	This chapter provides user's manual information for CRC driver components.

2.Reference

2.1 Reference Documents

The documents referred to in this document is shown in Table 2-1 to Table 2-2.

Table 2-1 Reference Documents(1/2)

Sl. No.	Title	Version
[1]	R-Car Gen4 AUTOSAR R19-11 MCAL User's Manual Modules Overview R-CarS4_V4H_V4M_MCAL_ModuleOverview.pdf	2.17
[2]	R-Car V4M AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual r11uz0306ej0010-rcarv4m-mcal-eum.pdf	0.10
[3]	R-Car V4M Series User's Manual: Hardware r19uh0217ej0051-r-carv4m.pdf	0.51

Table 2-2 Reference Documents(2/2)

Sl. No.	Title	Version
[4]	Specification of CAN Driver AUTOSAR_SWS_CANDriver.pdf	R19-11
[5]	General Specification of Basic Software Modules AUTOSAR_EXP_CDDDesignAndIntegrationGuideline.pdf	R19-11
[6]	Specification of DIO Driver AUTOSAR_SWS_DIODriver.pdf	R19-11
[7]	Specification of Module Flash Driver AUTOSAR_SWS_FlashDriver.pdf	R19-11
[8]	Specification of GPT Driver AUTOSAR_SWS_GPTDriver.pdf	R19-11
[9]	Specification of MCU Driver AUTOSAR_SWS_MCUDriver.pdf	R19-11
[10]	Specification of PORT Driver AUTOSAR_SWS_PortDriver.pdf	R19-11
[11]	Specification of SPI Handler/Driver AUTOSAR_SWS_SPIHandlerDriver.pdf	R19-11
[12]	Specification of Watchdog Driver AUTOSAR_SWS_WatchdogDriver.pdf	R19-11
[13]	Specification of Ethernet Driver AUTOSAR_SWS_EthernetDriver.pdf	R19-11

3.User Configuration Validation

This chapter provides help to analyze the error, warning and information messages displayed during the execution of <MSN> Driver Generation 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 displayed as a result of input file(s) validation, refer to Section <n>.6 “Specific Messages”, Section <n>.7 “Common Messages” for <MSN> Driver module messages and Section 2.10 “Error Messages”, 2.11 “Warning Messages”, 2.12 “Information Messages” of “R-Car Gen4 AUTOSAR MCAL R19-11 User’s Manual Getting Started” for common messages.

Note <n>: is a number from 4 to 17.

The Generation Tool displays error or warning or information when the user configures incorrect inputs. The format of Error/Warning/Information message is as shown below.

ERR/WRN/INF<mid><xxx>: <Error/Warning/Information Message>.

Note <mid>: Module Id for user configuration checks.
000 - for general.
001~123.
Error/Warning/Information Message of MCALConfGen executable.
Note <xxx>: 001-999 - Message Id.
001~999

4.CAN

4.1 Overview

Modern vehicles are becoming increasingly dependent on electronic modules. Most electrical or mechanical functions, which are critical for the normal operations of the vehicles, are controlled by embedded computer systems. CAN Driver performs the hardware access and offers a hardware-independent API to the upper layer. The only upper layer accessible to the CAN Driver is the CAN Interface. Several CAN Controllers can be controlled by the CAN Driver as long as they belong to the same CAN Hardware Unit.

The CAN Driver Component comprises two sections, Embedded Software and the Generation Tool, to achieve scalability and configurability.

This chapter describes features of the CAN Driver Generation Tool. CAN Driver Generation Tool is a command line tool that accepts ECU Configuration Description file(s) as input, and generates C Header and C Source files. This tool generates Can_Cfg.h, Can_PBcfg.c and Can_Lcfg.c files.

This chapter contains information on the options, input and output files of the CAN Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

ECU Configuration Description File contains information about CAN configuration.

4.2 CAN Driver Generation Tool Overview

Can Driver Generation Tool overview is shown below.

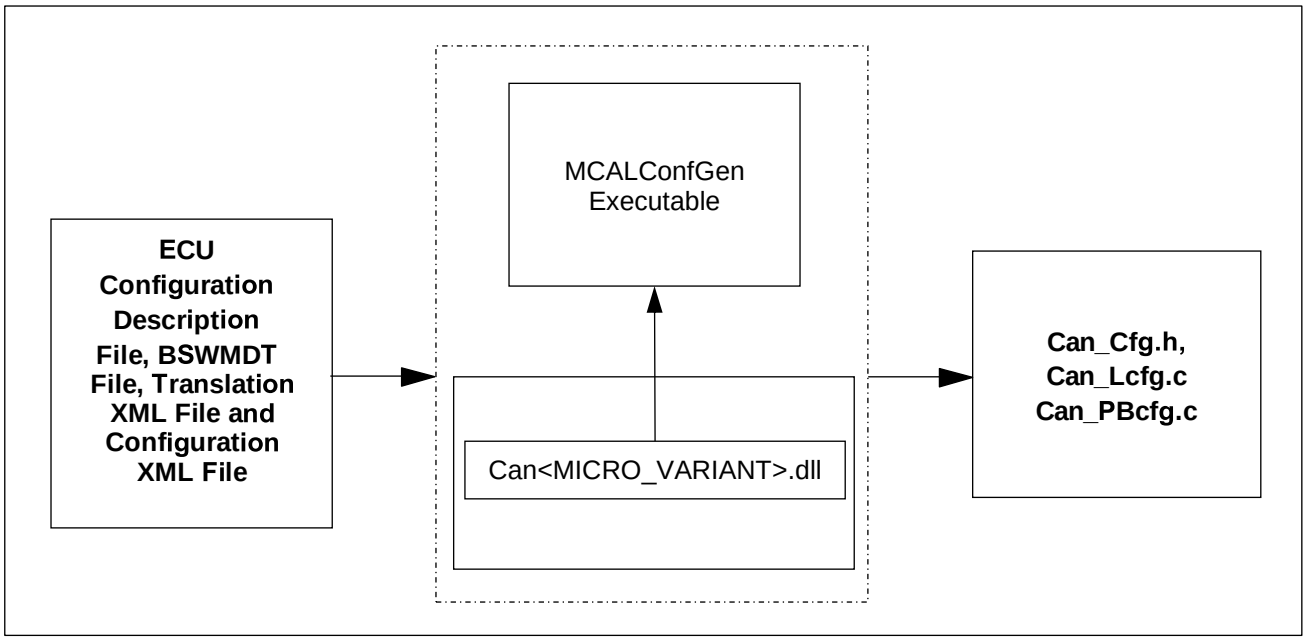


Figure 4-1 Overview of CAN Driver Generation Tool

CAN Driver Generation Tool is a command line tool that extracts and analyzes the configuration details provided in the input file, validates correctness of the data, and provides scalability and configurability for CAN Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input and displays appropriate context-sensitive error messages for wrong input and exits. Tool creates the Log file (Can.log) that contains the list of Error/Warning/Information messages in the output directory.

CAN Driver Generation Tool comprises 2 components, MCALConfGen executable and CanRCAR.dll. At runtime, the executable loads the dll to generate output files.
For the error-free input file, the tool generates the following output files: Can_Cfg.h, Can_PBcfg.c and Can_Lcfg.c.

Can_Cfg.h will be compiled and linked with CAN Driver Component. Can_PBcfg.c and Can_Lcfg.c will be compiled and linked separately from the other C Source files and placed in flash.

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

Note:

- The generation tool returns 1 when error, and return 0 when no error.
- CAN Driver Generation Tool uses “Common Published Information” from CAN module-specific BSWMDT File. This file should not be updated manually since it is “Static Configuration” file.

4.3 Input Files

CAN Driver Generation Tool accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input. This tool needs information about CAN Driver module. Hence ECU Configuration Description File should contain configuration of CAN Driver module. Generation Tool ignores any other AUTOSAR component configured in the ECU Configuration Description File. ECU Configuration Description File can be generated using configuration editor.

This file must comply with AUTOSAR standard ECU Configuration Description File format.

Note The detailed explanations about the parameters and containers are found in Parameter Definition File.

4.4 Output Files

CAN Driver Generation Tool generates configuration details in C Header and C Source files (Can_Cfg.h, Can_Lcfg.c and Can_PBcfg.c)

The content of each output file is given in the Table 4-1:

Table 4-1 Output Files Description

Output File	Details
Can_Cfg.h	This file contains pre-compile time parameters.
Can_Lcfg.c	This file contains link-time parameters.
Can_PBcfg.c	This file contains post-build time parameters.

Note Output files generated by CAN Driver Generation Tool should not be modified or edited manually.

4.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain CAN Driver module.
- If a Translation XML File is provided on command line, the Generation Tool will get this file as input.
- Otherwise, the Generation Tool will get the file Can.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input.
- Otherwise, the Generation Tool will get the file Can.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension '.trxml'.
- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by CAN Driver Generation 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 CAN Driver Generation Tool. Otherwise, the tool may not produce the expected results or may lead to errors/warnings/information messages.

Note For more information, please refer to the following appendix file: "V4M_Deviation_List.xlsx" in R-CAR V4M AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual.

4.6 Specific 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 errors, warning and information messages displayed by the Generation Tool.

4.6.1 Specific Error Messages

ERR080004: The parameter 'parameter name' in the container 'container name' should be configured.
This error occurs, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in the ECU Configuration Description File. The list of mandatory parameters with respect to container is listed in Table **4-2** to Table **4-4**.

Table 4-2 Parameter List for ERR080004 (1/3)

Parameter Name	Container Name
CanAlreadyInitDetCheck	CanGeneral
CanCriticalSectionProtection	
CanDevErrorDetect	
CanDeviceName	
CanMainFunctionModePeriod	
CanOsCounterRef	
CanPublicIcomSupport	
CanTimeoutDuration	
CanVersionCheckExternalModules	
CanVersionInfoApi	
CanEnableClkImmediateValue	
CanUnintendedInterruptCheck	
CanIsrCategory	
CanSelfTestApi	
CanRamTestApi	
CanEccErrorCorrect	
CanMainFunctionPeriod	CanMainFunctionRWPeriods
CanHardwareUnitSelect	CanGlobalConfiguration
CanClockSourceSelect	
CanEnableDLCCheck	
CanIntervalTimerPrescalerSet	
CanMirrorFunctionSupport	
CanPayloadOverflowModeSelect	
CanRxBufferPayloadLength	
CanTransmitPrioritySelect	
CanIcomConfigId	CanIcomConfig
CanIcomWakeOnBusOff	
CanIcomWakeupCauses	
CanIcomMessageId	CanIcomRxMessage
CanIcomMessageIdMask	

CanIcomCounterValue	
CanIcomMissingMessageTimerValue	
CanIcomPayloadLengthError	
CanIcomPayloadLengthValue	
CanIcomSignalMask	CanIcomRxMessageSignalConfig
CanIcomSignalValue	
CanIcomSignalOperation	
CanHardwareUnitSelect	CanGlobalConfiguration
CanClockSourceSelect	
CanEnableDLCCheck	
CanIntervalTimerPrescalerSet	
CanMirrorFunctionSupport	
CanPayloadOverflowModeSelect	
CanRxBufferPayloadLength	
CanTransmitPrioritySelect	

Table 4-3 Parameter List for ERR080004 (2/3)

Parameter Name	Container Name
CAN_E_TIMEOUT_FAILURE	CanDemEventParameterRefs
CanBusoffProcessing	CanController
CanControllerActivation	
CanControllerDefaultBaudrate	
CanControllerId	
CanControllerSelection	
CanControllerInterfaceMode	
CanEnableTransmitHistoryInterrupt	
CanRxProcessing	
CanTxProcessing	
CanEnableCanCanFDGateway	
CanFDGatewayForwardingFormat	
CanFDGatewayBRSBit	
CanControllerBaudRate	CanControllerBaudrateConfig
CanControllerBaudRateConfigID	
CanControllerSeg1	
CanControllerSeg2	
CanControllerSyncJumpWidth	
CanControllerFdBaudRate	CanControllerFdBaudrateConfig
CanControllerSeg1	
CanControllerSeg2	
CanControllerSyncJumpWidth	
CanControllerTxBitRateSwitch	
CanControllerRef	CanHardwareObject
CanHandleType	
CanIdType	
CanMemoryMode	
CanObjectId	
CanObjectType	

Table 4-4 Parameter List for ERR080004 (3/3)

Parameter Name	Container Name
CanHwFilterCode	CanHwFilter
CanHwFilterDLCCheckValue	
CanHwFilterMask	
CanHwFilterReceiveIdType	
CanHwFilterSourceNode	
CanEnableReceiveFIFOInterrupt	CanReceiveFIFOConfiguration
CanReceiveFIFOBufferDepth	
CanReceiveFIFOPayloadLength	
CanEnableTxRxFIFOInterrupt	CanTxRxFIFOConfiguration
CanTxRxFIFOBufferDepth	
CanTxRxFIFOPayloadLength	
CanEnableTransmitQueueInterrupt	CanTransmitQueueConfiguration

CanTransmitQueueBufferDepth	
CanTransmitQueueSelection	

Note

CanReceiveFIFOConfiguration, CanTxRxFIFOConfiguration and CanTransmitQueueConfiguration container and the parameters are mandatory only if CanMemoryMode is configured as RECEIVE_FIFO, TRANSMIT_RECEIVE_FIFO or TRANSMIT_QUEUE_MODE respectively.

CanHwFilter container and the parameters are mandatory only if CanObjectType in the same CanHardwareObject is configured 'RECEIVE'.

ERR080034: The calculated NBT (Nominal Bit Time) value should be in the range of <8-385>.

This error occurs, if the calculated NBT (Nominal Bit Time) value is not in the range of 8 to 385 for the particular controller. The calculation of NBT is done as follows:

NBT value = [CanControllerSeg1 + CanControllerSeg2 + 1] in CanControllerBaudrateConfig container.

ERR080061: The parameter 'CanMemoryMode' present in the container 'CanHardwareObject' should not be configured as <BUFFER_MODE>, since the parameter 'CanObjectType' is configured as <GATEWAY> object. GATEWAY object is applicable only for <TRANSMIT_RECEIVE_FIFO_MODE> or <TRANSMIT_QUEUE_MODE> mode.

This error occurs, if the parameter 'CanMemoryMode' present in the container 'CanHardwareObject' is configured as BUFFER_MODE, and the parameter 'CanObjectType' as GATEWAY mode.

ERR080066: The parameter 'CanMemoryMode' present in the container 'CanHardwareObject' should not be configured as <TRANSMIT_QUEUE_MODE>, since the parameter 'CanObjectType' is configured as <RECEIVE> object. TRANSMIT_QUEUE_MODE mode is applicable only for TRANSMIT or GATEWAY object.

This error occurs, if the parameter 'CanMemoryMode' present in the container 'CanHardwareObject' is configured as TRANSMIT_QUEUE_MODE, and the parameter 'CanObjectType' as RECEIVE object.

ERR080042: The maximum number of Transmit and Gateway objects configured for the controller < value of CanControllerSelection> should not be more than <value of MaxTransBuffer[<hardware unit number>]>.

Total number of Transmit and Gateway Object = Transmit Buffer + Transmit/ Gateway TxRxFIFO + Transmit/ Gateway Queue * (Queue Depth + 1).

This error will occur, if the number of Transmit Object for the CanController is more than the value of MaxTransBuffer[<hardware unit number>] (device-specific value) .

Note

If HTH is GATEWAY

HTH queue depth = CanTransmitQueueBufferDepth + 1 + 3

Else

HTH queue depth = CanTransmitQueueBufferDepth + 1

MaxTransBuffer = 32

ERR080013: The reference path <value of reference path> provided for the parameter 'Parameter name' in the container 'Container name' having short name <short name of container> is incorrect.

This error occurs, if the reference path is incorrect for the parameter 'parameter name' in the container 'container name'.

CanControllerPplClock or CanControllerPclkClock shall be checked when CanEnableClkImmediateValue is configured as <false>. Refer to Table 4-5 Parameter List for ERR080013.

Table 4-5 Parameter List for ERR080013

Parameter Name	Container Name
CanControllerPclkClock	CanConfigSet
CanControllerPplClock	
CanOsCounterRef	CanGeneral
CanControllerDefaultBaudrate	CanController
CanControllerRef	CanHardwareObject
CanMainFunctionRWPeriodRef	
CAN_E_INTERRUPT_CONTROLLER_FAILURE	CanDemEventParameterRefs
CAN_E_TIMEOUT_FAILURE	
CanGatewayCopyObjectRef	CanHwFilter
CanGatewaySourceControllerRef	

ERR080052: The parameter 'CanControllerPplClock' in the container 'CanConfigSet' should be configured, since the parameter 'CanClockSourceSelect' in the container 'CanGlobalConfiguration' having short name <short name of container> is configured as <CLKC> and the parameter 'CanEnableClkImmediateValue' in the container 'CanGeneral' is configured as <false>.

This error occurs, if the parameter 'CanControllerPplClock' is not configured, and the parameter 'CanClockSourceSelect' is configured as CLKC, and 'CanEnableClkImmediateValue' as false.

ERR080114: The configured value of parameter 'immediate clock value' in container 'CanConfigSet' should be consistent with clock setting on MCU configured in 'clock reference' parameter. Since the parameter 'CanEnableClkImmediateValue' in the container 'CanGeneral' is configured as <true>.

This error will occur, if 'CanEnableClkImmediateValue' in the container 'CanGeneral' is configured as true, and the value of 'immediate clock value' parameter is not consistent with the clock setting on the MCU.

The mapping of 'clock reference' and 'immediate clock value' parameters is provided in Table 4-6:

Table 4-6 Parameter List for ERR080114

Clock reference parameters	Immediate clock value parameters
CanControllerPplClock	CanControllerPplClockImmediateValue
CanControllerPclkClock	CanControllerPclkClockImmediateValue

ERR080116: The peripheral clock value in parameter <'CanControllerPclkClock/ CanControllerPclkClockImmediateValue' or 'CanControllerMainClock/ CanControllerMainClockImmediateValue' > should not be less than fCAN clock value 'fCAN clock value'.

This error occurs, if the clock value provided in the parameter <CanControllerPclkClock/ CanControllerPclkClockImmediateValue or CanControllerMainClock/ CanControllerMainClockImmediateValue> is less than fCAN clock value.

ERR080119: The parameter 'CanMainFunctionRWPeriodRef' in container 'CanHardwareObject' having short name <short name of CanHardwareObject> should be configured, since this RECEIVE hardware object is referring to a controller having 'CanRxProcessing' parameter is configured as <POLLING>, or <MIXED> and the parameter 'CanHardwareObjectUsesPolling' is configured as <true>.

This error will occur, if the parameter ‘CanMainFunctionRWPeriodRef’ is not configured for a RECEIVE hardware object, then this object is referring to a controller having CanRxProcessing parameter configured as POLLING or MIXED, and the parameter ‘CanHardwareObjectUsesPolling’ as true.

ERR080122: The value <configured value> configured for the parameter 'CanTransmitQueueSelection' present in the 'CanTransmitQueueConfiguration' container should be unique among hardware objects referring to the controller <CanControllerId value>.

This error will occur, if the value of the parameter ‘CanTransmitQueueSelection’ present in the 'CanTransmitQueueConfiguration' container is not unique among the hardware objects referring to the same controller.

ERR080123: 'The parameter 'CanHardwareObjectUsesPolling' in container 'CanHardwareObject' having short name <short name of CanHardwareObject container> should be configured since this RECEIVE hardware object is refering to a CanController container having short name <short name of CanController container> with 'CanRxProcessing' parameters are configured as <MIXED>.

This error occurs, if 'CanHardwareObjectUsesPolling' parameter in CanHardwareObject container is not configured and the hardware object is referring to a controller having CanTxProcessing or CanRxProcessing parameters configured as MIXED value.

ERR080134: 'The parameter ‘immediate clock value' in the container 'CanConfigSet' should be configured, since the parameter ‘CanEnableClkImmediateValue’ in the container 'CanGeneral' is configured as <true>.

This error occurs, if the parameter ‘Immediate clock value parameter’ is not configured, and the parameter CanEnableClkImmediateValue is configured as true.

The ‘Immediate clock value parameter’ is provided in Table 4-7.

Table 4-7 Parameter List for ERR080134

Immediate clock value parameters
CanControllerPclkClockImmediateValue
CanControllerPplClockImmediateValue
CanControllerMainClockImmediateValue

ERR080135: The configured value of parameter 'CanTransmitQueueSelection' in container having short name <short name of CanTransmitQueueConfiguration> should not be <TXQ3> since the type of its parent hardware object is configured as <GATEWAY>.

This error will occur, if the value of the parameter ‘CanTransmitQueueSelection’ is TXQ3, and the parent hardware object has CanObjectType as GATEWAY.

ERR080136: The configured value of 'CanTransmitQueueBufferDepth' parameter in container having short name <short name of CanTransmitQueueConfiguration> should not exceed <28> since the type of its parent hardware object is configured as <GATEWAY>.

This error will occur, if the value of the parameter ‘CanTransmitQueueBufferDepth’ exceeds <28> and the parent hardware object has CanObjectType as GATEWAY.

ERR080137: The total of transmit buffers allocated for (TXQ0 + TXQ1) or (TXQ2 + TXQ3 + number of <TRANSMIT_RECEIVE_FIFO_MODE> hardware objects) of controller <CanControllerId value> exceed the maximum number of transmit buffer.

This error will occur, if two transmit queue hardware objects refer to the same controller and:

Use TXQ0, TXQ1 (with the total of allocated transmit buffers) exceeding 16 OR use TXQ2, TXQ3 (with the total of allocated transmit buffers and the number of Transmit/Receive FIFO hardware objects of the controller) exceeding 16.

ERR080124: The parameter 'PARAM1' in the container 'PARAM2' should be configured as <false> since 'PARAM3' is configured as <MIXED> and 'CanHardwareObjectUsesPolling' is configured as <true> in the corresponding container 'CanHardwareObject'.

This error will occur, if PARAM1 to enable interrupt is configured as true when the relevant processing mode is configured as mixed, and the hardware object uses the polling mode. The combination of PARAM1, PARAM2 and PARAM3 is in Table 4-8:

Table 4-8 Parameter List for ERR080124

Object Type	Memory Mode	PARAM1	PARAM2	PARAM3
RECEIVE	RECEIVE_FIFO_MODE	CanEnableReceiveFIFOInterrupt	CanReceiveFIFOConfiguration	CanRxProcessing
RECEIVE	TRANSMIT_RECEIVE_FIFO_MODE	CanEnableTxRxFIFOInterrupt	CanTxRxFIFOConfiguration	CanRxProcessing

ERR080125: 'The parameter '<PARAM>' in the container '<CONTAINER>' which is a sub-container of 'CanHardwareObject' container having short name '<short name of CanHardwareObject>' should be configured as <false> since the parameter 'CanObjectType' of this container is not configured as <GATEWAY>.

This error occurs, if CanEnableTxRxFIFOOverwriteMode or CanEnableTransmitQueueOverwriteMode is configured as true, while the parent 'CanHardwareObject' has CanMemoryMode not configured as GATEWAY. The combination of PARAM and CONTAINER is in Table 4-9:

Table 4-9 Parameter List for ERR080125

CONTAINER	PARAM
CanTxRxFIFOConfiguration	CanEnableTxRxFIFOOverwriteMode
CanTransmitQueueConfiguration	CanEnableTransmitQueueOverwriteMode

ERR080044: The maximum number of Receive buffer configured for the controller <value of CanControllerSelection> should not be more than <value of MaxRecvBuffer>.

This error will occur, if the maximum number of Receive buffer configured for a CanController is more than the value of MaxRecvBuffer.

MaxRecvBuffer = 16

n: hardware unit number

ERR080140: The parameter 'PARAM1' in the container 'PARAM2' should be configured as <true> since 'PARAM3' is configured as <MIXED> and 'CanHardwareObjectUsesPolling' is configured as <false> in the corresponding container 'CanHardwareObject'.

This error will occur, if PARAM1 to enable interrupt is configured as false when the relevant processing mode is configured as mixed and parameter CanHardwareObjectUsesPolling configured as false.
The combination of PARAM1, PARAM2 and PARAM3 is in Table 4-10:

Table 4-10 Parameter List for ERR080140

Object Type	Memory Mode	PARAM1	PARAM2	PARAM3
RECEIVE	RECEIVE_FIFO_MODE	CanEnableReceiveFIFO Interrupt	CanReceiveFIFOConfiguration	CanRxProcessing
RECEIVE	TRANSMIT_RECEIVE_FIFO_MODE	CanEnableTxRxFIFOInterrupt	CanTxRxFIFOConfiguration	CanRxProcessing

ERR080039: Parameter 'CanIcomMessageIdMask' and 'CanIcomRxMessageSignalConfig' should not be defined together in the container.
The error occurs if CanIcomMessageIdMask and CanIcomRxMessageSignalConfig are defined together in the CanIcomRxMessage container.

ERR080040: All 'CanIcomRxMessage' instances the Message IDs which are defined in 'CanIcomMessageId' and in 'CanIcomRxMessageIdMask' should not overlap.
The error occurs if CanIcomRxMessage instances the Message IDs which are defined in CanIcomMessageId and in CanIcomRxMessageIdMask are overlap.

ERR080141: The parameter 'CanEnableClkImmediateValue' in the container'CanGeneral' should not be configured as <false> when the parameter 'CanClockSourceSelect' in the container'CanGlobalConfiguration' is configured as <CLK_XINCAN> since the reference parameter to MCU clock having short name 'CanControllerMainClock' is not available.
This error occurs, if the parameter ‘CanEnableClkImmediateValue’ is configured as <false> and the parameter ‘CanClockSourceSelect’ is configured as CLK_XINCAN.

ERR080014: The parameter 'CanUnintendedInterruptCheck' is configured as true but 'CAN_E_INTERRUPT_CONTROLLER_FAILURE' parameter is not configured.
This error occurs, if the parameter CanUnintendedInterruptCheck is configured as true but 'CAN_E_INTERRUPT_CONTROLLER_FAILURE' parameter is not configured.

ERR080146: The parameter ‘CanIcomCalloutFunction’ in the container ‘CanIcomGeneral’ should be configured, since ‘CanIcomPayloadLengthError’ parameter is configured as <true>.
This error occurs, if CanIcomCalloutFunction is not configured in the CanIcomGeneral container when CanIcomPayloadLengthError parameter in CanIcomRxMessage container is configured as <true>.
Note: If the CanIcomCalloutFunction is set as “NULL” “NULL_PTR”, it is considered as not configure.

ERR080151: The reference path provided for parameter <reference parameter name> in container ‘container name’ should not be duplicated with the reference path of other parameters.
This error will occur, if the reference path of <reference parameter name> in container <container name> is duplicated with other parameters. The reference parameter and container name as Table **4-11** Parameter List for ERR080151:

Table 4-11 Parameter List for ERR080151

Container name	Reference Parameter name
CanDemEventParameterRefs	CAN_E_TIMEOUT_FAILURE
	CAN_E_INTERRUPT_CONTROLLER_FAILURE

4.6.2 Specific Warning Messages

WRN080001: The value <value of CanHwFilterMask> configured for the parameter 'CanHwFilterMask' of the container 'CanHwFilter' having short name <short name> is not considered for the implementation since the parameter 'CanHwFilterReceiveIdType' in the same container is configured as <value of CanHwFilterReceiveIdType> and the parameter 'CanHandleType' in the 'CanHardwareObject' container is configured as <FULL>.

This warning occurs, if the value configured for the parameter 'CanHwFilterMask' of the container 'CanHwFilter' is other than 0x1FFFFFFF and 0x7FF, when the parameter 'CanHwFilterReceiveIdType' in the same container is configured as EXTENDED and STANDARD respectively, and CanHandleType in the CanHardwareObject container is configured as FULL.

WRN080008: Valid values of the parameter 'CanControllerFdBaudRate' in CanControllerFdBaudrateConfig container for the controller <value of CanControllerSelection> are <33/83/125/250/500/1000/2000/5000/8000>.

This warning occurs, if the value of the parameter 'CanControllerBaudRate' in CanControllerBaudrateConfig container is other than 33, 83, 125, 250, 500, 1000, 2000, 5000 or 8000Kbps for the particular controller.

WRN080009: The immediate value is configured in 'immediate clock value' parameter will be used without verification, because 'CanEnableClkImmediateValue' is configured as <true> and reference path in 'clock reference' parameter is incorrect.

This warning will occur, if CanEnableClkImmediateValue is true, and reference path of the clock reference parameter is incorrect.

The mapping of 'clock reference' and 'immediate clock value' parameters is provided in Table **4-12**:

Table 4-12 Parameter List for WRN080009

Clock reference parameters	Immediate clock value parameters
CanControllerPclkClock	CanControllerPclkClockImmediateValue
CanControllerPplClock	CanControllerPplClockImmediateValue

WRN080002: The configured value for the parameter '<CanDemEventParameter Name>' in the container 'CanDemEventParameterRefs' should not be configured since the value of the parameter '<Denpendent parameter>' of 'CanGeneral' container is configured as <false or disable>. This warning occurs, if the DemEvenParameter CAN_E_INTERRUPT_CONTROLLER_FAILURE of CanDemEventParameterRefs container is configured when its dependent parameter CanUnintendedInterruptCheck in container CanGeneral is configured as false or disable.

4.6.3 Specific Information Messages

INF080011: The reference clock in ‘clock reference’ parameter will be used, the configured value in ‘immediate clock value’ parameter will not be used.

This information will occur, if the value of CanEnableClkImmediateValue parameter is false, and the reference path given in ‘clock reference’ parameter is correct.

The mapping of ‘clock reference’ and ‘immediate clock value’ parameters is provided in Table 4-13:

Table 4-13 Parameter List for INF080011

Clock reference parameters	Immediate clock value parameters
CanControllerPclkClock	CanControllerPclkClockImmediateValue
CanControllerPplClock	CanControllerPplClockImmediateValue

INF080012: The value of ‘immediate clock value’ parameter will be used without verification, since the parameter ‘clock reference’ in ‘CanConfigSet’ container is not configured.

This information will occur, if CanEnableClkImmediateValue parameter is true, and ‘clock reference’ parameter is not configured.

The mapping of ‘clock reference’ and ‘immediate clock value’ parameters is provided in Table 4-14:

Table 4-14 Parameter List for INF080012

Clock reference parameters	Immediate clock value parameters
CanControllerPclkClock	CanControllerPclkClockImmediateValue
CanControllerPplClock	CanControllerPplClockImmediateValue
CanControllerMainClock	CanControllerMainClockImmediateValue

INF080014: Configuration of the below 'sub-container name' sub-container is ignored, since the parameter 'CanMemoryMode' present in the 'CanHardwareObject' container is not configured as <configured value of CanMemoryMode>.

This information occurs, if the respective sub-container is configured, and the parameter CanMemoryMode is not configured as the following value. Refer to Table 4-15:

Table 4-15 Parameter List for INF080014

Configured value of CanMemoryMode	Sub-Container name
RECEIVE_FIFO_MODE	CanReceiveFIFOConfiguration
TRANSMIT_RECEIVE_FIFO_MODE	CanTxRxFIFOConfiguration
TRANSMIT_QUEUE_MODE	CanTransmitQueueConfiguration

INF080015: Configuration of the parameters 'CanGatewayCopyObjectRef' and 'CanGatewaySourceControllerRef' in the below 'CanHwFilter' sub-container are ignored since the parameter 'CanObjectType' in the 'CanHardwareObject' container is not configured as <GATEWAY>.

This information occurs, if the parameter ‘CanGatewayCopyObjectRef’ or ‘CanGatewaySourceControllerRef’ in any CanHwFilter sub-container is configured for the hardware object not configured as GATEWAY.

INF080016: The parameter 'CanHardwareObjectUsesPolling' in the RECEIVE 'CanHardwareObject' container having short name '<short name of CanHardwareObject container>' will be ignored since none of the parameter 'CanRxProcessing' is configured as <MIXED> in the container CanController referred by this hardware object.

This information occurs, if neither of the parameters ‘CanTxProcessing’ and ‘CanRxProcessing’ is configured as MIXED in CanController container referred by a CanHardwareObject with CanHardwareObjectUsesPolling configured.

4.6.4 Specific Deviation List

None.

4.7 Common 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 errors, warning and information messages displayed by the Generation Tool.

4.7.1 Common Error Messages

ERR080003: 'CAN Driver, MCU Driver', 'OS Service Layer' or 'DEM' component is not present in the input file(s).
This error occurs, if CAN Driver, MCU Driver, OS Service Layer or DEM component is not present in the input ECU Configuration Description File(s).

ERR080007: The value <configured value> configured for the parameter 'CanControllerSelection' present in the container 'CanController' should be unique within a configuration set 'CanConfigSet'.
This error occurs, if the value of the parameter 'CanControllerSelection' is not unique within configuration sets.

ERR080038: The value of the parameter 'parameter name' configured in container 'container name' should be <actual value>. In general per configuration set, the value of 'parameter name' parameter should start with <min value> and should be sequential without any gaps.
This error occurs, if the value of the following parameters is not unique or not started from min value with sequential order in the respective container per configuration set. Refer to Table 4-16:

Table 4-16 Parameter List for ERR080038

Parameter Name	Container Name
CanControllerId	CanController
CanObjectId	CanHardwareObject
CanIcomConfigId	CanIcomConfig

ERR080089: The value of 'CanControllerBaudrateConfigID' should be unique within a controller.
This error occurs, if the value of CanControllerBaudRateConfigID is duplicated within the same controller.

ERR080045: The maximum number of CanHwFilter configured for the controller <value of CanControllerSelection> should not be more than <value of MaxRecvRule>.
This error occurs, if the maximum number of CanHwFilter configured for the controller is more than the value of MaxRecvRule.
MaxRecvRule = 384

ERR080111: The maximum number of CanHwFilter configured for all controllers of hardware unit <hardware unit number> should not be more than <value of MaxSharedRecvRule>.
This error occurs, if the maximum number of CanHwFilter configured for all controllers of a hardware unit is more than the value of MaxSharedRecvRule.
MaxSharedRecvRule = 768 (n = 0)
n: hardware unit number

ERR080064: The maximum number of Receive FIFO Object configured for all the controllers of hardware unit <hardware unit number> should not be more than <8>.
This error occurs, if the maximum number of Receive FIFO Object configured for all the controllers of a hardware unit is more than <8>.

ERR080065: The maximum number of Transmit/Receive FIFO configured for the controller <value of CanControllerSelection> should not be more than <3>.
This error occurs, if the number of Transmit/Receive FIFO Object for the CanController is more than 3.

ERR080068: The maximum number of Transmit Queue configured for the controller <value of CanControllerSelection> should not be more than <value of MaxTransQueue>.
This error occurs, if the number of Transmit Queue for the CanController is more than the value of MaxTransQueue.
MaxTransQueue = 4

ERR080005: Value of the parameter 'CanControllerSeg1' in below 'container name' container should be greater than or equal to the value of the parameter 'CanControllerSeg2' in same container for the controller <value of CanControllerSelection>.
This error occurs, if the value of the parameter 'CanControllerSeg1' in CanControllerBaudrateConfig or CanControllerFdBaudrateConfig container is less than that of the parameter 'CanControllerSeg2' in the same container for the particular controller.

ERR080006: Value of the parameter 'CanControllerSyncJumpWidth' in below 'container name' container should be less than or equal to the value of the parameter 'CanControllerSeg1' in same container for the controller <value of CanControllerSelection>.
This error occurs, if the value of the parameter 'CanControllerSyncJumpWidth' in CanControllerBaudrateConfig or CanControllerFdBaudrateConfig container is greater than that of the parameter 'CanControllerSeg1' in the same container for the particular controller.

ERR080033: Value of the parameter 'CanControllerSyncJumpWidth' in below 'container name' container should be less than or equal to value of the parameter 'CanControllerSeg2' in same container for the controller <value of CanControllerSelection>.
This error occurs, if the value of the parameter 'CanControllerSyncJumpWidth' in CanControllerBaudrateConfig or CanControllerFdBaudrateConfig container is greater than that of the parameter 'CanControllerSeg2' in the same container for the particular controller.

ERR080060: The parameter 'CanMemoryMode' present in the container 'CanHardwareObject' having short name <short name of CanHardwareObject> should not be configured as <RECEIVE_FIFO_MODE>, since the parameter 'CanObjectType' is configured as <configured value of CanObjectType> object.
This error occurs, if the parameter 'CanMemoryMode' present in the container 'CanHardwareObject' is configured as RECEIVE_FIFO_MODE, and the parameter 'CanObjectType' as TRANSMIT or GATEWAY object.

ERR080017: Value of the parameter 'Parameter Name' of the container 'Container name' should be within the range of <0 to 2047> for <STANDARD> CanHwFilterReceivedType and <0 to 536870911> for <EXTENDED> CanHwFilterReceivedType.

This error occurs, if CanHwFilterReceivedType is STANDARD and the values for the parameter 'CanHwFilterCode' or 'CanHwFilterMask' are other than 0x00 - 0x7FF (0 - 2047), and are other than 0x00 - 0x1FFFFFFF (0 - 536870911) when CanHwFilterReceivedType is EXTENDED. Refer to Table 4-17:

Table 4-17 Parameter List for ERR080017

Parameter Name	Container Name
CanHwFilterCode	CanHwFilter
CanHwFilterMask	

ERR080054: The CanHwFilter should be configured in the container 'CanHardwareObject' having short name <short name of CanHardwareObject>, since 'CanObjectType' is configured as <RECEIVE> or <GATEWAY>.

This error occurs, if CanHwFilter is not configured for any of the receive object.

ERR080055: The sub-container 'sub-container name' should be configured, IF the parameter 'CanMemoryMode' present in the container 'CanHardwareObject' is configured as <configured value of CanMemoryMode>.

This error occurs, if the respective sub-container is not configured for the following value of CanMemoryMode. Refer to Table 4-18:

Table 4-18 Parameter List for ERR080055

Configured Value of CanMemoryMode	Sub-Container
RECEIVE_FIFO_MODE	CanReceiveFIFOConfiguration
TRANSMIT_RECEIVE_FIFO_MODE	CanTxRxFIFOConfiguration
TRANSMIT_QUEUE_MODE	CanTransmitQueueConfiguration

ERR080056: The parameter 'CanReceiveFIFOInterruptSrcSel' in the container 'CanReceiveFIFOConfiguration' should be configured, since the parameter 'CanEnableReceiveFIFOInterrupt' in the same container is configured as <true>.

This error occurs, if the parameter 'CanReceiveFIFOInterruptSrcSel' in the container 'CanReceiveFIFOConfiguration' is not configured, and the parameter 'CanEnableReceiveFIFOInterrupt' in the same container is configured as true.

ERR080057: The parameter ' CanReceiveFIFOInterruptRatioSel' in the container 'CanReceiveFIFOConfiguration' should be configured, since the parameters 'CanEnableReceiveFIFOInterrupt' is configured as <true> and 'CanReceiveFIFOInterruptSrcSel' is configured as <SPECIFIED_FIFO_RATIO> in the same container.

This error will occur, if the parameter 'CanReceiveFIFOInterruptRatioSel' in the container 'CanReceiveFIFOConfiguration' is not configured, and the parameter 'CanEnableReceiveFIFOInterrupt' is configured as true, and CanReceiveFIFOInterruptSrcSel as SPECIFIED_FIFO_RATIO in the same container.

ERR080058: The parameter 'CanTxRxFIFOInterruptSrcSel' in the container 'CanTxRxFIFOConfiguration' should be configured, since the parameter 'CanEnableTxRxFIFOInterrupt' in the same container is configured as <true>.

This error occurs, if the parameter ‘CanTxRxFIFOInterruptSrcSel’ in the container ‘CanTxRxFIFOConfiguration’ is not configured, and the parameter ‘CanEnableTxRxFIFOInterrupt’ in the same container is configured as true.

ERR080059: The parameter 'CanTxRxFIFOInterruptRatioSel' in the container 'CanTxRxFIFOConfiguration' should be configured, since the parameters 'CanEnableTxRxFIFOInterrupt' is configured as <true> and 'CanTxRxFIFOInterruptSrcSel' is configured as <SPECIFIED_FIFO_RATIO_OR_EMPTY> in the same container.

This error will occur, if the parameter ‘CanTxRxFIFOInterruptRatioSel’ in the container ‘CanTxRxFIFOConfiguration’ is not configured, and the parameter ‘CanEnableTxRxFIFOInterrupt’ is configured as true, and CanTxRxFIFOInterruptSrcSel as SPECIFIED_FIFO_RATIO_OR_EMPTY in the same container.

ERR080067: The parameter 'CanTransmitQueueInterruptSrcSel' in the container 'CanTransmitQueueConfiguration' should be configured, since the parameter 'CanEnableTransmitQueueInterrupt' in the same container is configured as <true>.

This error occurs, if the parameter ‘CanTransmitQueueInterruptSrcSel’ in the container ‘CanTransmitQueueConfiguration’ is not configured, and the parameter ‘CanEnableTransmitQueueInterrupt’ in the same container is configured as true.

ERR080070: The maximum number of shared Rx/Tx Object configured for all the controllers of hardware unit <hardware unit number> should not be more than value <value of MaxSharedBuffer>.

This error will occur, if the maximum number of the shared Rx/Tx Object configured for all the controllers of a hardware unit is more than the value of MaxSharedBuffer.

MaxSharedBuffer = 77824 (n = 0)

n: hardware unit number

ERR080074: The parameter 'CanTransmitPrioritySelect' in the container 'CanGlobalConfiguration' having short name <short name of container> should not be configured as DEPEND_ON_NUMBER_OF_BUFFER since the parameter 'CanObjectType' in the container 'CanHardwareObject' having short name <short name of container> is configured as ‘TRANSMIT’ or ‘GATEWAY’ and the parameter CanMemoryMode in the same container is configured as TRANSMIT_QUEUE_MODE.

This error occurs, if the container ‘CanHardwareObject’ has the parameter ‘CanObjectType’ configured as ‘TRANSMIT’ or ‘GATEWAY’, the parameter ‘CanMemoryMode’ as TRANSMIT_QUEUE_MODE, and the parameter ‘CanTransmitPrioritySelect’ in the container CanGlobalConfiguration as DEPEND_ON_NUMBER_OF_BUFFER.

ERR080036: The value of parameter "CanTimeoutDuration" should be larger than or equal to "OsSecondsPerTick".

This error will occur, if the value of the parameter ‘CanTimeoutDuration’ is less than the OsSecondsPerTick.

ERR080082: In CAN-FD mode, Nominal Baud Rate Prescaler (NBRP bits) and Data Baud Rate Prescaler (DBRP bits) should be same to avoid loss of synchronization.
This error occurs, if for the same controller the Nominal brb and Data brb differs when CanFdSupport is on.

ERR080085: Calculated brp_value is out of range for the controller <controller id>. The calculated brp (baud rate prescaler division value) value should be in the range of <0- 1023 / 0- 255>.
This error occurs, when the brp_value is out of the range of <0-1023> for NBRP, <0-255> for DBRP.

ERR080079: The value configured for the parameter CanLPduReceiveCalloutFunction or 'CanIcomCalloutFunction ' which is present in the container 'CanGeneral' and 'CanIcomGeneral' should follow C syntax '< [a-zA-Z] [a-zA-Z0-9_] >'.
This error occurs, if the value configured for the CanLPduReceiveCalloutFunction or CanIcomCalloutFunction parameter does not adhere to C syntax (characters other than (a-z, A-Z, 0-9 or "_").

ERR080106: The parameter 'PARAM1' in the container 'PARAM2' should be configured as <false> since 'PARAM3' is configured as <POLLING>.
This error occurs, if PARAM1 to enable interrupt is configured as true, when the relevant processing mode is configured as the polling mode. The combination of PARAM1, PARAM2 and PARAM3 is in Table 4-19:

Table 4-19 Parameter List for ERR080106

Object Type	PARAM1	PARAM2	PARAM3
RECEIVE	CanEnableReceiveFIFOInterrupt	CanReceiveFIFOConfiguration	CanRxProcessing
RECEIVE	CanEnableTxRxFIFOInterrupt	CanTxRxFIFOConfiguration	CanRxProcessing
TRANSMIT	CanEnableTxRxFIFOInterrupt	CanTxRxFIFOConfiguration	CanTxProcessing
TRANSMIT	CanEnableTransmitQueueInterrupt	CanTransmitQueueConfiguration	CanTxProcessing

ERR080107: Valid values of the parameter 'CanHwFilterDLCCheckValue' in the container 'CanHwFilter' having short name <short name of CanHwFilter> are <0/1/2/3/4/5/6/7/8/12/16/20/24/32/48/64>.
This error occurs, if CanHwFilterDLCCheckValues is configured as the invalid number for CFDGAFLP0_j.GAFLDLC.

ERR080108: The parameter 'CanTxRxFIFOInterruptRatioSel' in the container 'CanTxRxFIFOConfiguration' should be FIFO_2_BY_8 or FIFO_4_BY_8 or FIFO_6_BY_8 or FIFO_8_BY_8, since this object is configured as <RECEIVE> and 'CanTxRxBufferDepth' is configured as <BUFFER_4> and 'CanEnableTxRxFIFOInterrupt' is configured as <true> and 'CanTxRxFIFOInterruptSrcSel' is configured as <SPECIFIED_FIFO_RATIO_OR_EMPTY>.
This error occurs, if CanTxRxFIFOInterruptRatioSel is configured as odd ratio, when the interruption is enabled and CanTxRxFIFOBufferDepth is configured as 4.

ERR080109: The parameter 'CanReceiveFIFOInterruptRatioSel' in the container 'CanReceiveFIFOConfiguration' should be FIFO_2_BY_8 or FIFO_4_BY_8 or FIFO_6_BY_8 or FIFO_8_BY_8, since 'CanReceiveFIFOBufferDepth' is configured as <BUFFER_4> and 'CanEnableReceiveFIFOInterrupt' is configured as <true> and 'CanReceiveFIFOInterruptSrcSel' is configured as <SPECIFIED_FIFO_RATIO>.

This error will occur, if CanReceiveFIFOInterruptRatioSel is configured as odd ratio, when the interruption is enabled and CanReceiveFIFOBufferDepth is configured as 4.

ERR080112: The parameter 'CanGatewaySourceControllerRef' in the container 'CanHwFilter' should be configured, since 'CanObjectType' is configured as <GATEWAY>.

This error occurs, if CanGatewaySourceControllerRef is not configured in the CanHwFilter container that belongs to gateway object.

ERR080113: The parameter 'CanControllerPclkClock' in the container 'CanConfigSet' should be configured, since the parameter 'CanEnableClkImmediateValue' in the container 'CanGeneral' is configured as <false>.

This error occurs, if the parameter 'CanControllerPclkClock' in the container 'CanConfigSet' is not configured, and the parameter 'CanEnableClkImmediateValue' in the container 'CanGeneral' is configured as false.

ERR080120: The value <configured value> configured for the parameter 'CanHardwareUnitSelect' present in the container 'CanGlobalConfiguration' should be unique.

This error occurs, if the value of the parameter 'CanHardwareUnitSelect' is not unique.

ERR080121: The value of parameter 'CanHardwareUnitSelect' in a container 'CanGlobalConfiguration' should be configured as <value of CanHardwareUnitSelect> since the parameter 'CanControllerSelection' in the container 'CanController' having short name <short name of container> is configured as <value of CanControllerSelection>.

This error occurs, if a controller channel is used while the hardware unit is not configured.

ERR080133: The value configured of 'CanControllerIdOffset' parameter in 'CanGeneral' container should be lower than or equal to <255 - total number of controller within a configset + 1>.

This error occurs, if (the value of 'CanControllerIdOffset' parameter + number of controller - 1) exceeds 255.

ERR080115: Parameter 'CanGatewayCopyObjectRef' in 'CanHwFilter' container should not refer to a TRANSMIT hardware object.

This error occurs, if the parameter 'CanGatewayCopyObjectRef' refers to a TRANSMIT hardware object container.

ERR080139: The parameter 'PARAM1' in the container 'PARAM2' should be configured as <true> since 'PARAM3' is configured as <INTERRUPT>.

This error will occur, if PARAM1 to enable interrupt is configured as false when the relevant processing mode is configured as interrupt mode. The combination of PARAM1, PARAM2 and PARAM3 as Table 4-20:

Table 4-20 Parameter list for ERR080139

Object Type	Memory Mode	PARAM1	PARAM2	PARAM3
RECEIVE	RECEIVE_FIFO_MODE	CanEnableReceiveFIFO	CanReceiveFIFOConfigu	CanRxPro

		Interrupt	ration	cessing
RECEIVE	TRANSMIT_RECEIVE_FIFO_MODE	CanEnableTxRxFIFOInterrupt	CanTxRxFIFOConfiguration	CanRxProcessing
TRANSMIT	TRANSMIT_RECEIVE_FIFO_MODE	CanEnableTxRxFIFOInterrupt	CanTxRxFIFOConfiguration	CanTxProcessing
TRANSMIT	TRANSMIT_QUEUE_MODE	CanEnableTransmitQueueInterrupt	CanTransmitQueueConfiguration	CanTxProcessing

ERR080117: The parameter 'CanMainFunctionRWPeriodRef' in container 'CanHardwareObject' should be configured since this TRANSMIT hardware object is referring to a controller having 'CanTxProcessing' parameter is configured as <POLLING>.

This error will occur, if the parameter CanMainFunctionRWPeriodRef is not configured for any TRANSMIT hardware object which is referring to a controller having parameter CanTxProcessing is configured as POLLING.

ERR080118: The configured value of 'CanMainFunctionRWPeriodRef' parameter is not the same among all TRANSMIT hardware objects which refer to 'CanController' container having 'CanTxProcessing' parameter is configured as <POLLING>.

This error will occur, if configured value of CanMainFunctionRWPeriodRef parameter is not the same among all TRANSMIT hardware objects which refer to the same Controller having CanTxProcessing parameter is configured as POLLING.

ERR080152: The CanTransmitHistoryInterruptSrcSel parameter in the CanController container should be configured since the CanEnableTransmitHistoryInterrupt parameter is configured as true.

This error will occur, if the parameter 'CanTransmitHistoryInterruptSrcSel' in the container 'CanController' is not configured since the CanEnableTransmitHistoryInterrupt parameter is configured as true, the error will be raised.

ERR080153: The combination of <'PARAM1' parameter in the 'PARAM2' container, 'PARAM3' parameter and PARAM4 parameter> should be configured as <true, false, any> or <true, true, FILLED_THREE_QUARTERS>, or <false, true, EACH_MESSAGE> since 'PARAM5' is configured as <INTERRUPT>.

This error will occur, if the 'PARAM1' parameter of the 'PARAM2' container don't configure as 'true' since the combination of 'PARAM3' parameter and 'PARAM4' parameter is <'false', 'any'> or <'true', 'FILLED_THREE_QUARTERS'> while the 'PARAM5' parameter is configured as INTERRUPT the error will be raised. Refer Table 4-21.

Table 4-21 Parameter list for ERR080153

Object Type	Memory Mode	PARAM1	PARAM2	PARAM3	PARAM4	PARAM5
TRANSMIT	TRANSMIT_RECEIVE_FIFO_MODE	CanEnableTxRxFIFOInterrupt	CanTxRxFIFOConfiguration	CanEnableTransmitHistoryInterrupt	CanTransmitHistoryInterruptSrcSel	CanTxProcessing

This error will occur, if the 'PARAM1' parameter of the 'PARAM2' container don't configure as 'false' since the combination of 'PARAM3' parameter and 'PARAM4' parameter is <'true', 'EACH_MESSAGE'> while the 'PARAM5' parameter is configured as INTERRUPT, the error will be raised. Refer Table 4-22.

Table 4-22 Parameter list for ERR080153

Object Type	Memory Mode	PARAM1	PARAM2	PARAM3	PARAM4	PARAM5
TRANSMIT	TRANSMIT_QUEUE_MODE	CanEnableTransmitQueueInterrupt	CanTransmitQueueConfiguration	CanEnableTransmitHistoryInterrupt	CanTransmitHistoryInterruptSrcSel	CanTxProcessing

ERR080154: The CanTransmitHistoryInterruptSrcSel parameter in the CanController container should not be configured as EACH_MESSAGE since the CanEnableTransmitHistoryInterrupt parameter is configured as true and the CanTxProcessing parameter is configured as POLLING mode.

This error will occur, if the CanTransmitHistoryInterruptSrcSel parameter in the CanController container is configured as EACH_MESSAGE since the CanEnableTransmitHistoryInterrupt parameter is configured as true and the CanTxProcessing parameter is configured as POLLING mode.

4.7.2 Common Warning Messages

WRN080007: parameter ‘CanEnableDLCCheck’ in the container 'CanGlobalConfiguration' having short name <short name of container> is configured as <true> and the parameter ‘CanHwFilterDLCCheckValue’ in the container ‘CanHwFilter’ having short name <short name of container> is configured as larger value than the payload size of the target buffer. Entire or a part of messages will be discarded.

This warning occurs, if the parameter ‘CanEnableDLCCheck’ is configured as true, and the parameter ‘CanHwFilterDLCCheckValue’ is configured as the value larger than the payload size of the buffer to which this filter is assigned.

WRN080013: The parameter 'CanGatewayCopyObjectRef' in 'CanHwFilter' container whose CanHardwareObject parent container having the short name "<short name of CanHardwareObject>" shall not be used since it was configured as a reference to itself.

This warning occurs, if the parameter 'CanGatewayCopyObjectRef' configured to reference itself in the ‘CanHardwareObject’.

4.7.3 Common Information Messages

INF080006: Calculated Baud rate prescaler is a float value for the controller <value of CanControllerId>. It should be an integer value. Generation Tool is considering Baud rate prescaler value as <value of baud rate>. Hence actual Baud rate <value of new baud rate> is different from configured Baud rate <value of CanControllerBaudRate>.
This information will be generated, when the calculated baud rate prescaler is a float value for the particular controller instead of the integer value.

INF080009: CanHwFilter containers in the container 'CanHardwareObject' having short name <short name of CanHardwareObject> are ignored, since this hardware object is configured as <TRANSMIT>.
This information will be generated, when the parameter 'CanHwFilter' is configured, and the hardware object as TRANSMIT.

INF080010: The parameter 'CanHwFilterSourceNode' in the container 'CanHwFilter' having short name <short name of container> should be configured as <FROM_OTHER_NODE>, since the parameter 'CanMirrorFunctionSupport' in the container 'CanGlobalConfiguration' having short name <short name of container> is configured as <false>. Parameter 'CanHwFilterSourceNode' should be configured as <FROM_OWN_NODE> only when the parameter ' CanMirrorFunctionSupport ' is configured as <true>.
This information will be generated, when the parameter 'CanHwFilterSourceNode' is configured as <FROM_OWN_NODE> and the parameter 'CanMirrorFunctionSupport' as false.

INF080001: The value of parameter 'CanTimeoutDuration' is rounded and configured as <value>.
This information will be generated, when the division of the parameter 'CanTimeoutDuration' by the parameter OsSecondsPerTick results into non-integer value.

INF080013: The container 'CanGlobalConfiguration' having short name <short name of container> should not be configured since there is no controller configured for hardware unit <value of hardware unit>.
This information will occur, if a hardware unit is configured while the controllers are not used.

4.7.4 Common Deviation List

None.

5.DIO

5.1 Overview

The DIO Driver component provides the service for initializing the whole DIO structure of the microcontroller.
The DIO Driver module comprises two sections, Embedded Software and the Generation Tool, to achieve scalability and configurability.
This chapter describes the features of the DIO Driver Generation Tool. This tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File and generates DIO Driver C Source and C Header files.
DIO Driver Generation Tool will generate Dio_Cfg.h and Dio_Lcfg.c files.

This chapter contains information on the options, input and output files of the DIO Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.
ECU Configuration Description File contains information about DIO configuration
This program is built based on Microsoft .NET Framework.
Refer to the following sites about the license in detail.
https://msdn.microsoft.com/en-us/library/ms994405.aspx

5.2 DIO Driver Generations Tool Overview

DIO Driver Generation Tool overview is shown in *Figure 5-1*.

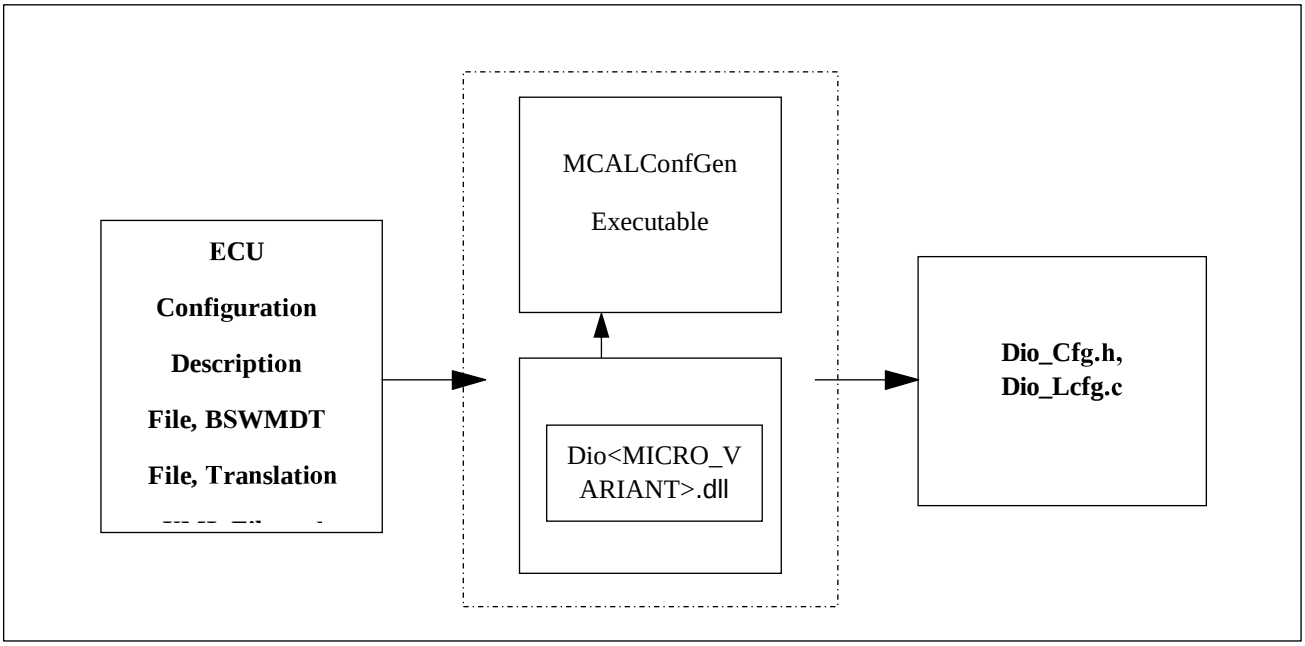


Figure 5-1 Overview of DIO Driver Generation Tool

DIO Driver Generation Tool is a command line tool that extracts and, analyzes the configuration details provided in the input file, validates correctness of the data, and provides scalability and configurability for DIO Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input and displays appropriate context sensitive error messages for wrong input and

exits. Tool creates the Log file (Dio.log) that contains the list of Error/Warning/Information messages in the output directory.
DIO Driver Generation Tool comprises two components, MCALConfGen executable and Dio<MICRO_VARIANT>.dll. At runtime, the executable component loads the dll to generate output files.
For the error-free input file, the tool generates the following output files:
In R19-11, DIO Driver Generation Tool will generate Dio_Cfg.h and Dio_Lcfg.c files.
Dio_Cfg.h will be compiled and linked with DIO Driver Component. Dio_Lcfg.c will be compiled and linked separately from the other C Source files and placed in flash.
ECU Configuration Description File can be created or edited using ECU Configuration Editor.

- The generation tool returns 1 when error, and 0 when no errors.
- DIO Driver Generation Tool uses “Common Published Information” from DIO module-specific BSWMDT File. DIO module-specific BSWMDT File should not be updated manually since it is “Static Configuration” file.

5.3 Input Files

DIO Driver Generation Tool accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input. DIO Driver Generation Tool needs information about DIO Driver module. Hence ECU Configuration Description File should contain configuration of DIO Driver module. Generation 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.

Note: The detailed explanations about the parameters and containers are found in Parameter Definition File.

5.4 Output Files

DIO Driver Generation Tool generates configuration details in C Header and C Source files.
(In R19-11, DIO Driver Generation Tool will generate Dio_Cfg.h and Dio_Lcfg.c files).
The content of each output file is given in the *Table 5-1*.

Table 5-1 Output Files Description

Output File	Details
Dio_Cfg.h	This file contains the macro definitions for development error detects, version info API and channel group. It also contains DIO Channel Configuration Handles, DIO Port Configuration Handles and DIO Channel Group Configuration Handles.
Dio_Lcfg.c	This file contains Data Structures for DIO Port Group Configuration, DIO Port Channel Configuration and DIO Port Channel Group Configuration. It also contains information on Number of ports and Channels configured.

Note: Output files generated by DIO Driver Generation Tool should not be modified or edited manually.

5.5 Precautions

ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.

- The input file must contain DIO Driver module.
- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Dio.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Dio.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension '.trxml'.
- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by DIO Driver Generation 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 DIO Driver Generation Tool. Otherwise, Tool may not produce the expected results or may lead to errors/warnings/information messages.

Note For more information, please refer to the following appendix file: *"V4M_Deviation_List.xlsx"* in *R-Car Gen4 AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual*.

5.6 Specific 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 device-specific error, warning and information messages displayed by the Generation Tool.

5.6.1 Specific Error Messages

None.

5.6.2 Specific Warning Messages

None.

5.6.3 Specific Information Messages

None.

5.6.4 Specific Deviation List

None.

5.7 Common 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 common error, warning and information messages displayed by the Generation Tool.

5.7.1 Common Error Messages

ERR120003: 'DIO Driver' Component is not present in the input file(s).
This error will occur, if DIO Driver Component is not present in the input ECU Configuration Description File(s).

ERR120004: The parameter 'parameter name' in the container 'container name' should be configured.
This error will occur, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File.
The list of mandatory parameters with respect to container is listed in *Table 5-2*:

Table 5-2 Parameter List for ERR120004

Parameter Name	Container Name
DioDevErrorDetect	DioGeneral
DioMaskedWritePortApi	
DioReadChannelOutputValueApi	
DioFlipChannelApi	
DioVersionInfoApi	
DioVersionCheckExternalModules	
DioCriticalSectionProtection	
DioReadChannelGroupOutputValueApi	
DioExclusiveControl	
DioExclusiveTimeout	
DioExclusiveSelection	
DioPortName	DioPort
DioChannelBitPosition	DioChannel
DioPortMask	DioChannelGroup
DioPortOffset	

ERR120005: The value <value for DioPortName> configured for the parameter 'DioPortName' present in the container 'DioPort' should be unique within a configuration set.
This error will occur, if the value for parameter DioPortName present in the container DioPort is not unique.

ERR120006: The value for parameter 'DioChannelBitPosition' present in the container 'DioChannel' of the DIO port group 'value for DioPortName parameter' is invalid.

This error will occur, if the value of the parameter ‘DioChannelBitPosition’ present in the container ‘DioChannel’ for the respective DIO port group is not valid as the configured channel bit position must belong to the respective DIO port group.
Example:
- GPIO
Suppose in GPIO_0_BITS_00_TO_18, only 0 to 18 channels are available. If user configures channels, which does not belong to GPIO_0_BITS_00_TO_18 then it is invalid configuration.

ERR120007: The value <value for DioChannelBitPosition> configured for the parameter 'DioChannelBitPosition' present in the container 'DioChannel' of the DIO port group <value for DioPortName parameter> should be unique.
This error will occur, if the value for parameter DioChannelBitPosition present in the container DioChannel is not unique for respective DIO port group configured for the parameter DioPortName.

ERR120008: The value for parameter 'DioPortMask' <value for the DioPortMask> present in the container 'DioChannelGroup' of the DIO port group <value for DioPortName parameter> is invalid.
This error will occur, if value for parameter DioPortMask present in the container DioChannelGroup for respective DIO port group is not valid as the grouped channels must belong to the respective DIO port group.
Example:
- GPIO
Suppose in GPIO_0_BITS_00_TO_18, only 0 to 18 channels are available. In this case, user should not consider channel 19 and channel 20 in channel grouping, since channel 19 and channel 20 does not belong to GPIO_0_BITS_00_TO_18.

ERR120009: The value for parameter 'DioPortMask' <value for the DioPortMask> present in the container 'DioChannelGroup' of the DIO port group <value for DioPortName parameter> is invalid. While masking, channels should be grouped in continuous order.
This error will occur, if value for parameter DioPortMask present in the container DioChannelGroup is not valid. The grouped channels should be continuous order.
Example: Channel 1 and Channel 5 cannot be grouped in one DIO Channel Group since they are not continuous channels.

ERR120010: The value for parameter 'DioPortOffset' <value for the DioPortOffset> present in the container 'DioChannelGroup' of the DIO port group <value for DioPortName parameter> is invalid. The value of the parameter 'DioPortOffset' should be equal to the start position of the DIO channel group.
This error will occur, if the value for parameter DioPortOffset present in the container DioChannelGroup is not equal to the start position of the DIO channel group (first bit 1 of DioPortMask).
Example: Suppose that the DIO channel grouping started from Channel 2 and value for parameter DioPortOffset is not 2.

ERR120011: The short name <short name for DioPort> configured for the container 'DioPort' should be unique.
This error will occur, if short name of the container DioPort is not unique in ECU Configuration Description File.

ERR120012: The short name <short name for DioChannel> configured for the container 'DioChannel' should be unique.

This error will occur, if short name of the container DioChannel is not unique in a configuration set.

ERR120013: The short name <short name for DioChannelGroup> configured for the container 'DioChannelGroup' should be unique.
This error will occur, if short name of the container DioChannelGroup is not unique in a configuration set.

ERR120014: The reference path <path> provided for the DEM parameter 'Parameter Name' within the container 'Container Name' is incorrect or is not configured.
This error will occur, if path provided for the DEM parameter in "DioDemEventParameterRefs" is incorrect and corresponding configured parameter is configured as true.

ERR120015: The parameter <DEM parameter> in the container 'DioDemEventParameterRefs' should be configured with a valid reference path when the parameter <parameter> in the container 'DioGeneral' is enabled
This error will occur, if path provided for the DEM parameter in "DioDemEventParameterRefs" is not configured when Configured parameter is configured as true.

5.7.2 Common Warning Messages

None.

5.7.3 Common Information Messages

None.

5.7.4 Common Deviation List

None

6.ETH

6.1 Overview

The ETH Software Component provides the service to initialize the whole ETH structure of the microcontroller.

The ETH Software Component comprises two sections, Embedded Software and the Generation Tool, to achieve scalability and configurability.

The chapter describes the features of the ETH Software Generation Tool. ETH Software Generation Tool is a command line tool that extracts information from ECU Configuration Description File and generates ETH Software C Header files and Source Files (Eth_Cfg.h and Eth_PBcfg.c).

This chapter contains information on the options, input and output files of the ETH Software Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

ECU Configuration Description File contains information about ETH configuration.

6.2 ETH Driver Generation Tool Overview

ETH Driver Generation Tool overview is shown in Figure 6.1

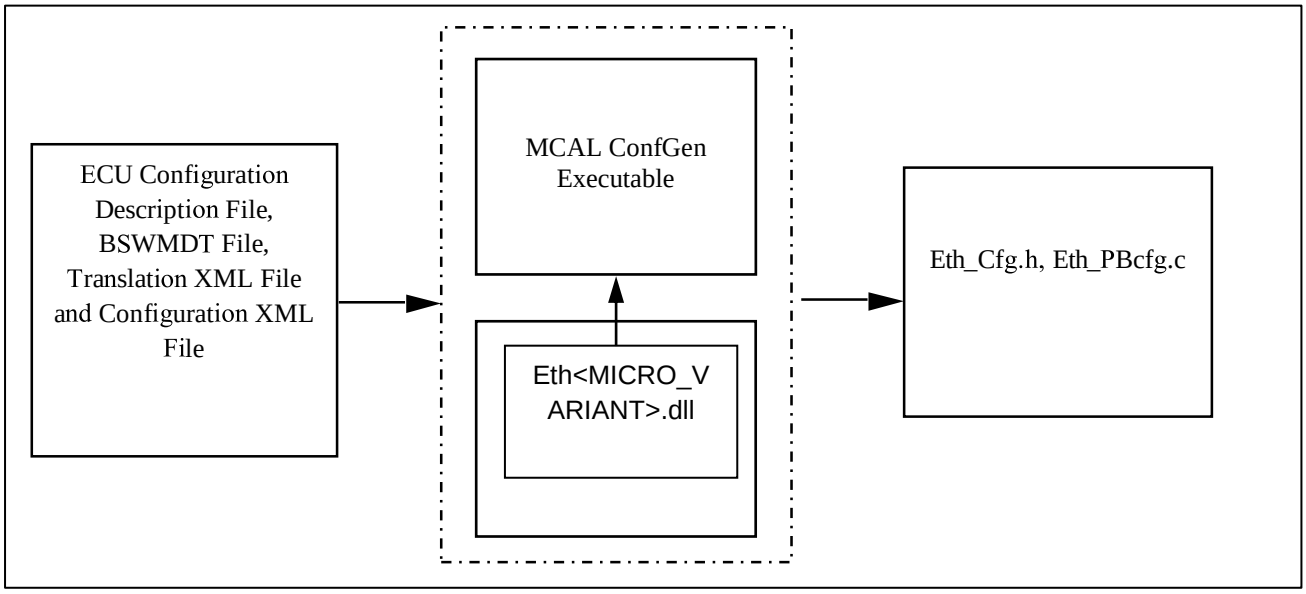


Figure 6.1 Overview of ETH Driver Generation Tool

ETH Driver Generation Tool is a command line tool that extracts and, analyzes the configuration details provided in the input file, and validates correctness of the data and provides scalability and configurability for ETH Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML and Configuration XML File as input and displays appropriate context-sensitive error messages for wrong input and exits. Tool creates the Log file (Eth.log) that contains the list of Error/Warning/Information messages in the output directory.

ETH Driver Generation Tool comprises 2 components, MCALConfGen executable and Eth<MICRO_VARIANT>.dll. At runtime, the executable loads the dll to generate output files.

For the error-free input file, the tool generates the following output files:

ETH Driver generation tool will generate Eth_Cfg.h and Eth_PBcfg.c files.

Eth_Cfg.h will be compiled and linked with ETH Driver Component. Eth_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash.

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

Note: The generation tool returns 1 when error, and 0 when no error.

ETH Driver Generation Tool uses “Common Published Information” from ETH module specific BSWMDT File. This file should not be updated manually since it is “Static Configuration” file.

6.3 Input Files

ETH Driver Generation Tool accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input. This tool needs information about ETH Driver module. Hence ECU Configuration Description File should contain configuration of ETH Driver module. Generation 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.

Note: The detailed explanations about the parameters and containers are found in Parameter Definition File.

6.4 Output Files

ETH Driver Generation Tool generates configuration details in C Header and C Source files (Eth_Cfg.h and Eth_PBcfg.c).

The content of each output file is given in the Table 6-1

Table 6-1 Output Files Description

Output File	Details
Eth_Cfg.h	This file contains pre-compile time parameters. It contains macros for general configuration parameters, Enabling/Disabling ISRs of hardware units, related Dem event, and Configuration Set Handles.
Eth_PBcfg.c	This file contains post-build configuration data. It contains module-specific configuration, structure for configuration.

Note: Output files generated by ETH Driver Generation Tool should not be modified or edited manually.

6.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.

- The input file must contain ETH Driver module.

- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Eth.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Eth.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension '.trxml'.
- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by ETH Driver Generation 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 ETH Driver Generation Tool. Otherwise the tool may not produce the expected results or may lead to errors/warnings/information messages.

Note For more information, please refer to the following appendix file: "V4M_Deviation_List.xlsx" in R-CAR V4M AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual.

6.6 Specific 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 Generation Tool.

6.6.1 Specific Error Messages

ERR088004: The parameter 'parameter name' in the container 'container name' should be configured.
This error will occur, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File. The list of mandatory parameters with respect to container is listed in the following table:

Table 6-2 Parameter List for ERR088004 (1/2)

Container	Parameters
EthGeneral	EthCriticalSectionProtection
	EthDevErrorDetect
	EthDeviceName
	EthInputClockRefImmediateValue
	EthGetDropCountApi
	EthGetEtherStatsApi
	EthGetCounterValuesApi
	EthGetRxStatsApi
	EthGetTxErrorCounterValuesApi
	EthGetTxStatsApi
	EthGlobalTimeSupport
	EthIndex
	EthUnintendedInterruptCheck
	EthIsrCategory
	EthMainFunctionPeriod
	EthMaxCtrlsSupported
	EthOsCounterRef
	EthQosSupport
	EthRegisterCheckInitTime
	EthRegisterCheckRunTime
	EthTimeout
	EthVersionCheckExternalModules
	EthVersionInfoApi
	EthSwitchManagementSupport

Table 6-3 Parameter List for ERR088004 (2/2)

Container	Parameters
EthCtrlConfig	EthBeTimeStampStore
	EthCtrlEnableMii
	EthCtrlEnableRxInterrupt
	EthCtrlEnableTxInterrupt
	EthCtrlIdx
	EthCtrlMacLayerSpeed
	EthCtrlMacLayerType
	EthCtrlPhyAddress
	EthDuplexMode
	EthEnableCBS
	EthInternalLoopBack
	EthNwCntrlFiltering
	EthRamSize
	EthSRPTalkerFiltering
	EthStreamFiltering
	EthStreamTimeStampStore
	EthTxQuePriority
	EthUnitSelection
	EthTxClockDelayMode
	EthRxClockDelayMode
EthTxQueueConfig	EthTxQueueIdx
	EthTxQueueBufs
EthRxQueueConfig	EthRxQueueIdx
	EthRxQueueBufs
EthCtrlTxQueueShaper	EthTxQueuePolicy
EthCtrlPriority	EthCtrlTxDefaultQueueRef
EthCtrlPriorityMapping	EthCtrlPriorityValue
	EthCtrlTxQueueRef
EthCtrlOffloading	EthCtrlEnableOffloadChecksumIPv4
	EthCtrlEnableOffloadChecksumICMP
	EthCtrlEnableOffloadChecksumTCP
	EthCtrlEnableOffloadChecksumUDP

ERR088007: The value for parameter '<DEM Error ID parameter name>' in the container 'EthDemEventParameterRefs' having short name '<short name>' is not correct.
This error will occur, if the parameter '<DEM Error ID parameter name>' is not correct.

Table 6-4 Parameter List for ERR088007

Container	Parameters
EthDemEventParameterRefs	ETH_E_ACCESS
	ETH_E_ALIGNMENT
	ETH_E_CRC
	ETH_E_DMA_ERROR
	ETH_E_INTERRUPT_CONTROLLER_FAILURE
	ETH_E_LATECOLLISION
	ETH_E_MULTIPLECOLLISION
	ETH_E_OVERSIZEFRAME
	ETH_E_REGISTER_CORRUPTION
	ETH_E_RX_FRAMES_LOST
	ETH_E_SINGLECOLLISION
	ETH_E_UNDERSIZEFRAME

ERR088015: The sub container 'EthCtrlTxQueueShaper' in the container 'EthTxQueueConfig' should be configured since the value of parameter 'EthQosSupport' is configured as <true> in the container 'EthGeneral'.

This error occurs, if EthQosSupport is <true>, and the container 'EthCtrlTxQueueShaper' does not exist.

Table 6-5 Parameter List for ERR088015

Container	Parameters
EthGeneral	EthQosSupport

ERR088016: The sub container 'EthCtrlPriority' in the container 'EthCtrlConfig' should be configured since the value of parameter 'EthQosSupport' is configured as <true> in the container 'EthGeneral'.

This error occurs, if EthQosSupport is <true>, and the container 'EthCtrlPriority' does not exist.

Table 6-6 Parameter List for ERR088016

Container	Parameters
EthGeneral	EthQosSupport

ERR088017: The value for parameter ' EthTxQueueIdx' ('EthRxQueueIdx') must use a zero-based index and is unique between 'EthTxQueueConfig' ('EthRxQueueConfig')

This error occurs, if the TX or RX Queues are not unique or not do have a zero-based index for a particular ConfigSet / Controller.

Table 6-7 Parameter List for ERR088017

Container	Parameters
EthTxQueueConfig	EthTxQueueIdx
EthRxQueueConfig	EthRxQueueIdx

ERR088018: The value of parameter 'EthCtrlIdx' or 'EthUnitSelection' in the container 'EthCtrlConfig' should be different across 'EthCtrlConfig' Configsets.

This error occurs, if any value of EthCtrlIdx and EthUnitSelection is the same as other EthCtrlIdx and EthUnitSelection across EthCtrlConfig Configsets.

Table 6-8 Parameter List for ERR088018

Container	Parameters
EthCtrlConfig	EthCtrlIdx
	EthUnitSelection

ERR088020: Same priority value 'EthCtrlPriorityValue' is assigned to different TX Queues across EthCtrlPriorityMapping containers

This error occurs, if 2 or more TX Queues are assigned to the same traffic priority.

Table 6-9 Parameter List for ERR088020

Container	Parameters
EthCtrlPriorityMapping	EthCtrlPriorityValue

ERR088021: No TX Queue found for this traffic class. The 'EthCtrlTxDefaultQueueRef'/'EthCtrlTxQueueRef' value in the Container 'EthCtrlPriority'/ 'EthCtrlPriorityMapping' points to an invalid TX Queue.

This error occurs, if the any of the 'EthCtrlTxDefaultQueueRef'/'EthCtrlTxQueueRef' value in the Container 'EthCtrlPriority'/'EthCtrlPriorityMapping' points to an invalid TX Queue.

Table 6-10 Parameter List for ERR088021

Container	Parameters
EthCtrlPriority	EthCtrlTxDefaultQueueRef
EthCtrlPriorityMapping	EthCtrlTxQueueRef

ERR088022: When EthSRPTalkerFiltering is TRUE, 0 must be set to the last 2 bytes of EthPatternStream.

This error occurs, when EthSRPTalkerFiltering is TRUE and the last 2 bytes of EthPatternStream are not 0.

Table 6-11 Parameter List for ERR088022

Container	Parameters
EthCtrlConfig	EthSRPTalkerFiltering
EthRxQueueConfig	EthPatternStream

ERR088027: The combination of the parameters EthCtrlMacLayerType and EthCtrlMacLayerSubType is incorrect.

The following combinations are supported:
If selected EthCtrlMacLayerType = <ETH_MAC_LAYER_TYPE_XGMII> and EthCtrlMacLayerSubType = <REDUCED>, which chooses <ETH_RGMII> for Ethernet interface.
This error occurs when the values of the parameters 'EthCtrlMacLayerType' and 'EthCtrlMacLayerSubType' are other than the followings:

When EthCtrlMacLayerType = <ETH_MAC_LAYER_TYPE_XGMII> and EthCtrlMacLayerSubType = <REDUCED>.

Table 6-12 Parameter List for ERR088027

Container	Parameters
EthCtrlConfig	EthCtrlMacLayerSubType
	EthCtrlMacLayerType

ERR088028: When parameter EthEnableCBS is configured as <false>, parameter EthTxQuePriority must not be configured as <ETH_AVBALT>.

This error occurs if the parameter 'EthEnableCBS' is configured as <false>, when the parameter 'EthTxQuePriority' is configured as <ETH_AVBALT>.

Table 6-13 Parameter List for ERR088028

Container	Parameters
EthCtrlConfig	EthEnableCBS
	EthTxQuePriority

ERR088030: EthRamSize' set to 'EthCtrlConfig' where 'EthCtrlIdx' is <Value of EthCtrlIdx> is too small for the configuration.
At a minimum, <TotalSize> is required.
It does not include the area will obtain by Eth_ProvideTxBuffer.

This error is displayed when 'EthRamsize' is smaller than 'TotalSize'.
The formula to calculate 'TotalSize' is as the following 'Pseudo rule':

TotalSize = 0.

/* TxRx descriptor chain */
RequiredSize = 16 + (64 + 64) * 8.
TotalSize = TotalSize + RequiredSize.

/* Tx descriptor */
FOR (i = 0; i < EthTxQueueConfig; i++)
  RequiredSize = 16 + (EthTxQueueBufs + 1) * 16.
  TotalSize = TotalSize + RequiredSize;
ENDFOR

```
/* Rx descriptor */
FOR (i = 0; i < EthRxQueueConfig; i++)
   RequiredSize = 16 + (EthRxQueueBufs + 1) * 24;
   TotalSize = TotalSize + RequiredSize;
ENDFOR

/* Rx buffer */
FOR (i = 0; i < EthRxQueueConfig; i++)
   FOR (j = 0; j < EthRxQueueBufs; j++)
      RequiredSize = 16 + 1518 + 4 + 2;
      TotalSize = TotalSize + RequiredSize;
   ENDFOR
ENDFOR
```

Table 6-14 Parameter List for ERR088030

Container	Parameters
EthCtrlConfig	EthRamsize
EthTxQueueConfig	EthTxQueueBufs
EthRxQueueConfig	EthRxQueueBufs

ERR088033: The reference path <value of EthOsCounterRef> provided for the parameter 'EthOsCounterRef' in the container 'EthGeneral' having short name <short name of container EthGeneral> is incorrect.

This error will occur if the reference path is incorrect for the parameter 'EthOsCounterRef' in the container 'EthGeneral'.

Table 6-15 Parameter List for ERR088033

Container	Parameters
EthGeneral	EthOsCounterRef

ERR088035: When parameter 'EthTxQueueIdx' has queue <EthTxQueueIdx value>, 'EthCtrlTxQueueBwFraction' parameter must be configured in container 'EthCtrlTxQueueShaper' having short name <short name EthCtrlTxQueueShaper> since parameter 'EthTxQueuePolicy' is <ETH_CBS>.

This error occurs when the following combination is configured for the parameters:
EthEnableCBS = true
EthTxQueueIdx = 2, 3
EthTxQueuePolicy = ETH_CBS
EthCtrlTxQueueBwFraction is not configured

Table 6-16 Parameter List for ERR088035

Container	Parameters
EthCtrlConfig	EthEnableCBS
EthTxQueueConfig	EthTxQueueIdx
EthCtrlTxQueueShaper	EthTxQueuePolicy
	EthCtrlTxQueueBwFraction

ERR088039: The parameter 'ETH_E_INTERRUPT_CONTROLLER_FAILURE' in the container 'EthDemEventParameterRefs' should be configured since the value of the parameter 'EthUnintendedInterruptCheck' of 'EthGeneral' container is configured as <true or enable>.

This error occurs if the value of parameter EthUnintendedInterruptCheck is configured as <true or enable> but the parameter 'ETH_E_INTERRUPT_CONTROLLER_FAILURE' in the container 'EthDemEventParameterRefs not configured.

Table 6-17 Parameter List for ERR088039

Container	Parameters
EthGeneral	EthUnintendedInterruptCheck
EthDemEventParameterRefs	ETH_E_INTERRUPT_CONTROLLER_FAILURE

ERR088040: The value of parameter <EthMaxCtrlsSupported> in container <EthGeneral> must be configured equal to the number of controllers in container <EthConfigSet> is(are) configured as <1/2/3>.

This error occurs if the The number of controllers in container EthConfigSet is(are) configured more than or small than the value of EthMaxCtrlsSupported.

Table 6-18 Parameter List for ERR088040

Container	Parameters
EthGeneral	EthMaxCtrlsSupported

ERR088043: The reference path provided for parameter '<DemEventParameter>' in container '<EthDemEventParameterRefs>' should not be duplicated with the reference path of other parameter.

This error occurs if the reference path provided for Dem Parameter in the container EthDemEventParameterRefs is not unique.

6.6.2 Specific Warning Messages

WRN088001: The container 'EthCtrlTxQueueShaper' having short name <short name of EthCtrlTxQueueShaper> is ignored since parameter 'EthEnableCBS' in container 'EthCtrlConfig' having short name <short name of EthCtrlConfig> is set as false.

This warning will occur if EthEnableCBS is false and the container EthCtrlTxQueueShaper is configured in the EthCtrlConfig container.

WRN088002: Parameter 'EthTxQueuePolicy' should be <ETH_NONE> in container 'EthCtrlTxQueueShaper' having short name <short name EthCtrlTxQueueShaper> since queue <EthTxQueueIdx value> does not support CBS.

This warning occurs when the following combination is configured for the parameters:
EthEnableCBS = true
EthTxQueueIdx = 0, 1
EthTxQueuePolicy = ETH_CBS

Table 6-19 Parameter List for WRN088002

Container	Parameters
EthCtrlConfig	EthEnableCBS
EthTxQueueConfig	EthTxQueueIdx
EthCtrlTxQueueShaper	EthTxQueuePolicy

WRN088003: Value of 'EthCtrlTxQueueBwFraction' parameter is not used since parameter 'EthTxQueuePolicy' is <ETH_NONE>.

This warning occurs when the following combination is configured for the parameters:
EthEnableCBS = true
EthTxQueuePolicy = ETH_NONE
EthCtrlTxQueueBwFraction is configured

Table 6-20 Parameter List for WRN088003

Container	Parameters
EthCtrlConfig	EthEnableCBS
EthCtrlTxQueueShaper	EthTxQueuePolicy
	EthCtrlTxQueueBwFraction

WRN088004: The configured value for the parameter '<EthDemEventParameter Name>' in the container 'EthDemEventParameterRefs' should not be configured since the value of the parameter '<Denpendent parameter>' of 'EthGeneral' container is configured as <false or disable>."

This warning occurs, if the DemEvenParameter ETH_E_INTERRUPT_CONTROLLER_FAILURE of EthDemEventParameterRefs container is configured when its dependent parameter EthUnintendedInterruptCheck in container EthGeneral is configured as false or disable.

Table 6-21 Parameter List for WRN088004

Container	Parameters
EthGeneral	EthUnintendedInterruptCheck
EthDemEventParameterRefs	ETH_E_INTERRUPT_CONTROLLER_FAILURE

6.6.3 Specific Information Messages

None

6.6.4 Specific Deviation List

None

6.7 Common 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 common errors, warning and information messages displayed by the ETH Generation Tool.

6.7.1 Common Error Messages

ERR088003: 'ETH Driver/MCU Driver/DEM' Component is not present in the input file(s).

For ETH Driver:
This error occurs, if ETH Driver component is not present in the input ECU Configuration Description File(s).

For MCU Driver:
This error occurs, if MCU Driver component is not present in the input ECU Configuration Description File(s).

For DEM:
This error occurs, if DEM component is not present in the input ECU Configuration Description File(s), and the DEM parameter is not empty.

ERR088006: The value for parameter 'parameter name' in the container 'container name' is invalid.
This error occurs, if the value for the parameter 'EthCtrlPhyAddress' in the container 'EthCtrlConfig' is not the format of the MAC address.

Table 6-22 Parameter List for ERR088006

Container	Parameters
EthCtrlConfig	EthCtrlPhyAddress

ERR088012: The parameter 'ETH_E_REGISTER_CORRUPTION' must be configured if any of 'EthRegisterCheckInitTime' and 'EthRegisterCheckRunTime' are enabled.
This error will occur, if ETH_E_REGISTER_CORRUPTION is not configured even though at least EthRegisterCheckInitTime or EthRegisterCheckRunTime is configured as TRUE.

Table 6-23 Parameter List for ERR088012

Container	Parameters
EthGeneral	EthRegisterCheckInitTime
	EthRegisterCheckRunTime
EthDemEventParameterRefs	ETH_E_REGISTER_CORRUPTION

ERR088029: The value for parameter 'EthCtrlIdx' must use a zero-based index.
This error occurs if the parameter 'EthCtrlIdx' does not have a zero-based index for a particular ConfigSet / Controller.

Table 6-24 Parameter List for ERR088029

Container	Parameters
EthCtrlConfig	EthCtrlIdx

6.7.2 Common Warning Messages

None

6.7.3 Common Information Messages

None

6.7.4 Common Deviation List

None.

7.FLS

7.1 Overview

The FLS Software component provides the service to initialize the whole FLS structure of the microcontroller.

The FLS Software Component comprises two sections, Embedded Software and the Generation Tool, to achieve scalability and configurability.

The chapter describes the features of the FLS Software Generation Tool. FLS Software Generation Tool is a command line tool that extracts information from ECU Configuration Description File and generates FLS. Software C Header files and Source Files (Fls_Cfg.h and Fls_PBcfg.c).

This chapter contains information on the options, input and output files of the FLS Software Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

ECU Configuration Description File contains information about FLS configuration.

This program is built based on Microsoft .NET Framework.

Refer to the following sites about license in detail.

https://msdn.microsoft.com/en-us/library/ms994405.aspx

7.2 FLS Driver Generation Tool Overview

FLS Driver Generation Tool overview is shown in Figure 7.1.

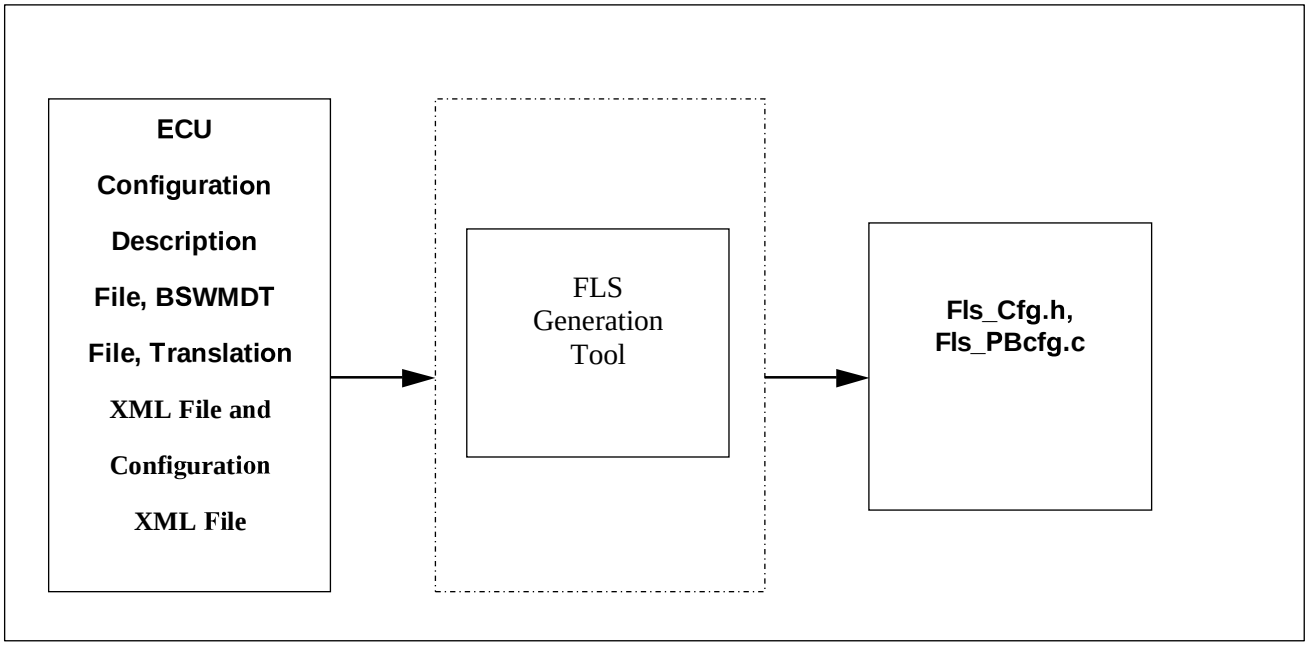


Figure 7.1 Overview of FLS Driver Generation Tool

FLS Driver Generation Tool is a command line tool that extracts and analyzes the configuration details provided in the input file, validates correctness of the data, and provides scalability and configurability for FLS Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input and displays appropriate context-sensitive error messages for wrong input and exits. Tool creates the Log file Fls.log) that contains the list of Error/Warning/Information messages in the output directory.

FLS Driver Generation Tool comprises 2 components, MCALConfGen executable and Fls<MICRO_VARIANT>.dll. At runtime, the executable loads the dll to generate output files.
For the error-free input file, the tool generates the following output files:

If FLS Driver is configured for Data Flash, then generation tool will generate

Fls_Cfg.h, Fls_PBcfg.c files.

Fls_Cfg.h will be compiled and linked with FLS Driver Component. Fls_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash.

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

The generation tool returns 1 when error, and 0 when no error.

- FLS Driver Generation Tool uses “Common Published Information” from FLS module-specific BSWMDT File. This file should not be updated manually, since it is “Static Configuration” file.

7.3 Input Files

FLS Driver Generation Tool accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input. FLS Driver Generation Tool needs information about FLS Driver module. Hence ECU Configuration Description File should contain configuration of FLS Driver module. Generation 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.

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

7.4 Output Files

FLS Driver Generation Tool generates configuration details in C Header and C Source files (Fls_Cfg.h, Fls_PBcfg.c).

The content of each output file is given in the Table 7.1:

Table 7.1 Output Files Description

Output File	Details
Fls_Cfg.h	This file contains pre-compile time parameters.
Fls_PBcfg.c	This file contains post-build time parameters.

Note Output files generated by FLS Driver Generation Tool should not be modified or edited manually.

7.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively. The input file must contain FLS Driver module and FLS related parameters of MemIf, Det, SchM, Dem, Os, Fee module.
- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Fls.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation tool will get this file as input. Otherwise, the Generation Tool will get the file Fls.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension '.trxml'.Configuration XML File should contain the file extension '.cfgxml'. 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.
- If the output files generated by FLS Driver Generation 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 FLS Driver Generation Tool. Otherwise, the tool may not produce the expected results or may lead to errors/warnings/information messages.

Note Please refer the FLS Component User Manual for deviations from AUTOSAR specifications, if any.

7.6 Specific Messages

7.6.1 Specific Error Messages

None.

7.6.2 Specific Warning Messages

None.

7.6.3 Specific Information Messages

None.

7.6.4 Specific Deviation Messages

None.

7.7 Common 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 Generation Tool.

7.7.1 Common Error Messages

ERR092003: <Module> Component is not present in the input file(s).

This error will occur, if <Module> Driver components mentioned below are not present in the input ECU Configuration Description File(s).

Table 7.2 Dependencies Modules

Module
Dem
Os

ERR092004: The parameter <parameter name> in the container <container name> should be configured.

This error occurs, if any of the mandatory configuration parameter(s) mentioned in Table 7.3 is (are) not configured in ECU Configuration Description File.

Table 7.3 List of Mandatory Parameters in FLS Driver's Configuration

Container	Parameters
FlsGeneral	FlsBaseAddress
	FlsBlankCheckApi
	FlsCancelApi
	FlsDevErrorDetect
	FlsRuntimeErrorDetect
	FlsGetJobResultApi
	FlsGetStatusApi
	FlsPageBufferWrap
	FlsSetModeApi
	FlsTotalSize
	FlsUseInterrupts
	FlsExclusiveControl
	FlsExpectedHwIdMask

Container	Parameters
	FlsReadTime
	FlsCpgExclusiveControl
	FlsSemaphore
	FlsVersionInfoApi
	FlsVersionCheckExternalModules
	FlsCriticalSectionProtection
	FlsDeviceName
	FlsTimeoutMonitoring
	FlsSelectPinGroup
	FlsWriteBuffer
	FlsExternalSpaceReadMode
	FlsAccess
	FlsWriteVerifyCheck
	FlsOsCounterRef
	FlsClockSetConfirmTimeout
	FlsClockSetTendCheckTimeout
	FlsBurstLengthValue
	FlsAcLoadOnJobStart
	FlsDriverIndex
	FlsCpgWriteVerifyCheck
	FlsTimeoutSupervisionEnabled
	FlsEraseVerificationEnabled
	FlsWriteVerificationEnabled
	FlsTransientFaultDetect
	FlsCompareApi
	FlsSuspendApi
	FlsResumeApi
	FlsSpecificConfigApi
FlsSerialFlashGeneral	FlsEnableEraseErrorBit

Container	Parameters
	FlsEnableProgramErrorBit
	FlsDDRCalibration
FlsHyperFlashGeneral	FlsEnableEraseErrorBit
	FlsEnableProgramErrorBit
FlsPublishedInformation	FlsWriteTime
	FlsEraseTime
	FlsErasedValue
	FlsExpectedHwId
	FlsAcLocationErase
	FlsAcLocationWrite
	FlsAcSizeErase
	FlsAcSizeWrite
	FlsSpecifiedEraseCycles
FlsConfigSet	FlsDefaultMode
	FlsMaxReadFastMode
	FlsMaxReadNormalMode
	FlsMaxWriteNormalMode
	FlsMaxWriteFastMode
	FlsProtection
	FlsAcErase
	FlsAcWrite
	FlsCallCycle
FlsSerialFlashDDRPattern	FlsNumberOfPattern
	FlsPatternAddress
	FlsPatternValue
FlsSerialFlashDDRVerify	FlsAddressAndDataBitSize
	FlsAddressLength
	FlsClockDelay
	FlsDummyCycle

Container	Parameters
	FlsNegationDelay
	FlsNextAccessDelay
	FlsOptionalData
	FlsOptionalDataLength
	FlsRpcFrequency
	FlsSDRReadCommand
FlsSerialFlashRead	FlsCommand
	FlsOptionalData
	FlsOptionalDataLength
	FlsDummyCycle
	FlsInternalStrobeDelay
	FlsNextAccessDelay
	FlsNegationDelay
	FlsClockDelay
	FlsRpcFrequency
	FlsAddressAndDataRate
	FlsAddressLength
	FlsAddressAndDataBitSize
FlsSerialFlashReadStatusReg	FlsCommand
	FlsDummyCycle
	FlsPositionOfBusyBit
	FlsPositionOfWriteEnableBit
	FlsPositionOfEraseErrorBit
	FlsPositionOfProgramErrorBit
	FlsNextAccessDelay
	FlsNegationDelay
	FlsClockDelay
	FlsRpcFrequency
	FlsDataRate
	FlsDataBitSize

Container	Parameters
FlsSerialFlashReadHwId	FlsCommand
	FlsDummyCycle
	FlsNextAccessDelay
	FlsNegationDelay
	FlsClockDelay
	FlsRpcFrequency
	FlsDataRate
	FlsDataBitSize
FlsSerialFlashErase	FlsCommand
	FlsDummyCycle
	FlsNextAccessDelay
	FlsNegationDelay
	FlsClockDelay
	FlsRpcFrequency
	FlsAddressAndDataRate
	FlsAddressLength
	FlsAddressAndDataBitSize
FlsSerialFlashProgram	FlsCommand
	FlsDummyCycle
	FlsNextAccessDelay
	FlsNegationDelay
	FlsClockDelay
	FlsRpcFrequency
	FlsAddressAndDataRate
	FlsAddressLength
	FlsAddressAndDataBitSize
FlsSerialFlashWriteEnable	FlsCommand
	FlsDummyCycle
	FlsNextAccessDelay
	FlsNegationDelay
	FlsClockDelay

Container	Parameters
	FlsRpcFrequency
	FlsDataRate
	FlsDataBitSize
FlsSector	FlsNumberOfSectors
	FlsSectorSize
	FlsSectorStartaddress
	FlsPageSize
FlsExternalDriver	FlsSpiReference
FlsDemEventParameterRefs	FLS_E_CLOCK_SET_FAILURE
FlsHyperFlashRead	FlsClockDelay
	FlsRpcFrequency
	FlsDummyCycle
	FlsNegationDelay
	FlsNextAccessDelay
FlsHyperFlashReadStatusReg	FlsClockDelay
	FlsRpcFrequency
	FlsDummyCycle
	FlsNegationDelay
	FlsNextAccessDelay
	FlsPositionOfDeviceReadyBit
	FlsPositionOfEraseErrorBit
	FlsPositionOfProgramErrorBit
FlsReadStatusSetupCycle	FlsTransactionAddress
	FlsTransactionData
FlsHyperFlashReadHwId	FlsIDReadAddress0
	FlsClockDelay
	FlsRpcFrequency
	FlsDummyCycle
	FlsNegationDelay
	FlsNextAccessDelay
FlsEnterCFI	FlsTransactionAddress
	FlsTransactionData
FlsExitCFI	FlsTransactionAddress
	FlsTransactionData
FlsHyperFlashErase	FlsCommand

Container	Parameters
	FlsClockDelay
	FlsRpcFrequency
	FlsDummyCycle
	FlsNegationDelay
	FlsNextAccessDelay
FlsEraseSetupCycle	FlsTransactionAddress
	FlsTransactionData
FlsHyperFlashProgram	FlsClockDelay
	FlsRpcFrequency
	FlsDummyCycle
	FlsNegationDelay
	FlsNextAccessDelay
FlsProgramSetupCycle	FlsTransactionAddress
	FlsTransactionData
FlsSerialFlashSpecificConfig	FlsAddress
	FlsAddressAndDataRate
	FlsAddressBitSize
	FlsAddressLength
	FlsClockDelay
	FlsCommand
	FlsCommandBitSize
	FlsDataBitSize
	FlsDataLength
	FlsRpcFrequency
	FlsDummyCycle
	FlsNegationDelay
	FlsNextAccessDelay
	FlsOperation
	FlsOptionalCommand
	FlsOptionalCommandBitSize
	FlsOptionalCommandEnable
	FlsOptionalData
	FlsOptionalDataBitSize
	FlsOptionalDataLength
FlsHyperFlashSpecificConfig	FlsAddress
	FlsClockDelay

Container	Parameters
	FlsDataLength
	FlsRpcFrequency
	FlsDummyCycle
	FlsNegationDelay
	FlsNextAccessDelay
	FlsOperation
FlsSpecificConfigSetupCycles	FlsTransactionAddress
	FlsTransactionData

Note:

The parameters related to functional safety are mandatory, but those are not listed in the table above. The parameters related to functional safety can be found in the description field of Parameter Definition File (PDF). Those parameters shall be configured as default value if functional safety is not applicable (usage of those parameters are explained in the safety application note, provided only for the users who apply functional safety).

ERR092005: The parameter 'FlsExpectedHwId/FlsExpectedHwIdMask' in the container 'FlsPublishedInformation/FlsGeneral' should be configured as a hexadecimal string.

This error will occur, if the parameter FlsExpectedHwId/FlsExpectedHwIdMask in the container FlsPublishedInformation is not configured as a hexadecimal string.

ERR092006: The value configured for the parameter 'FlsMaxReadNormalMode/FlsMaxReadFastMode/FlsMaxWriteNormalMode' must be an even number.

This error will occur, if the parameter 'FlsMaxReadNormalMode/FlsMaxReadFastMode/FlsMaxWriteNormalMode' is not configured as an even number when FlsSelectPinGroup is set to FLS_QSPI0_QSPI1_PIN_GROUP.

ERR092009: The value configured for the parameters in FlsSector are wrong. 'FlsSectorStartaddress', 'FlsSectorSize', and 'FlsNumberOfSectors' must be configured so that the start address of the first sector must be zero, the total size of all flashable sectors is equal to FlsTotalSize and sectors should be configured continuously.

Note: FlsTotalSize is a parameter of the container FlsGeneral.

This error will occur, if the value configured for the parameters in FlsSector are wrong. 'FlsSectorStartaddress', 'FlsSectorSize', and 'FlsNumberOfSectors' must be configured so that the start address of the first sector must be zero, the total size of all flashable sectors is equal to FlsTotalSize and sectors should be configured continuously.

ERR092012: The value configured for the parameter 'FlsAddressLength' in the container 'FlsSerialFlashRead' must be 'THREE_BYTES' or 'FOUR_BYTES' since the parameter 'FlsExternalSpaceReadMode' configured as 'True' in the container 'FlsGeneral'.

IF value of FlsAccess is FLS_SERIAL_FLASH_DEVICE. This error will occur, if the parameter 'FlsAddressLength' in the container 'FlsSerialFlashRead' is configured different 'THREE_BYTES' or 'FOUR_BYTES' since the parameter 'FlsExternalSpaceReadMode' configured as 'True' in the container 'FlsGeneral'.

ERR092015: The value configured for the parameter 'FlsJobEndNotification/FlsJobErrorNotification' in the container 'FlsConfigSet' should follow C syntax <[a-zA-Z][a-zA-Z0-9_]>.

This error will occur, if the value of configuration parameters mentioned does not adhere to C syntax as if the value contains characters other than (a-z, A-Z, 0-9 or "_"). The parameter value should always start with an alphabet.

ERR092016: The parameters 'FlsJobEndNotification' and 'FlsJobErrorNotification' in the container 'FlsConfigSet' should be configured with different values.

This error will occur, if the parameters FlsJobEndNotification and FlsJobErrorNotification are configured with same values in FlsConfigSet container.

Table 7.4 Expected value for parameters.

Container	Parameters
FlsConfigSet	FlsJobEndNotification
	FlsJobErrorNotification

ERR092017: The parameters 'FlsJobEndNotification' and 'FlsJobErrorNotification' in the container 'FlsConfigSet' should be configured together.

This error will occur, if any one of the parameters FlsJobEndNotification and FlsJobErrorNotification are not configured, as FlsJobEndNotification is configured and FlsJobErrorNotification is not configured or vice-versa.

ERR092018: The parameters 'FlsJobEndNotification' and 'FlsJobErrorNotification' in the container 'FlsConfigSet' should be configured since 'FlsUseInterrupts' in container 'FlsGeneral' is configured as <True>.

This error will occur, if the parameters 'FlsJobEndNotification' and 'FlsJobErrorNotification' are not configured since 'FlsUseInterrupts' in container 'FlsGeneral' is configured as <True>.

ERR092021: The parameter <DEM parameter> in the container 'FlsDemEventParameterRefs' should be configured with a valid reference path when the parameter <parameter> in the container 'FlsGeneral' is enabled.

Note: For FLS_E_CLOCK_SET_FAILURE, error message will be:

The parameter FLS_E_CLOCK_SET_FAILURE in the container 'FlsDemEventParameterRefs' should be configured with a valid reference path.

This error will occur, if DEM parameter is not configured while the function that used it is enabled:

Table 7.5 Expected value for parameters.

Parameters	DEM Parameters
FlsWriteVerifyCheck	FLS_E_WRITE_VERIFY_FAILURE
FlsExclusiveControl	FLS_E_GET_CONTROL_FAILURE
FlsSemaphore	FLS_E_RELEASE_SEMAPHORE_FAILURE
	FLS_E_GET_SEMAPHORE_FAILURE
FlsCpgExclusiveControl	FLS_E_CPG_GET_CONTROL_FAILURE

Parameters	DEM Parameters
FlsCpgWriteVerifyCheck	FLS_E_CPG_WRITE_VERIFY_FAILURE

ERR092022: The parameters ‘FlsPositionOfDeviceReadyBit’, ‘FlsPositionOfBusyBit’, ‘FlsPositionOfWriteEnableBit’, ‘FlsPositionOfEraseErrorBit’ and ‘FlsPositionOfProgramErrorBit’ configured with same value '$value'.

When the FlsAccess is FLS_SERIAL_FLASH_DEVICE. If the value configured for the parameters 'FlsPositionOfBusyBit’, ‘FlsPositionOfWriteEnableBit’, ‘FlsPositionOfEraseErrorBit’ and ‘FlsPositionOfProgramErrorBit’ in the ‘FlsSerialFlashReadStatusReg' container is not different.

When the FlsAccess is FLS_HYPER_FLASH_DEVICE. If the value configured for the parameters 'FlsPositionOfDeviceReadyBit’, ‘FlsPositionOfEraseErrorBit’ and ‘FlsPositionOfProgramErrorBit’ in the ‘FlsHyperFlashReadStatusReg' container is not different.

ERR092023: The command configured at the containers ‘FlsSerialFlashErase’, ‘FlsSerialFlashReadHwId’, ‘FlsSerialFlashProgram’, ‘FlsSerialFlashRead‘, ‘FlsSerialFlashReadStatusReg’, ‘FlsSerialFlashWriteEnable’, ‘FlsSerialFlashDDRVerify’ have same value '$value'

IF value of FlsAccess is FLS_SERIAL_FLASH_DEVICE, this error will occur, if the values configured for the parameters 'FlsCommand’ or ‘FlsSDRReadCommand’ in the containers ‘FlsSerialFlashErase’, ‘FlsSerialFlashReadHwId’, ‘FlsSerialFlashProgram’, ‘FlsSerialFlashRead‘, ‘FlsSerialFlashReadStatusReg’, ‘FlsSerialFlashWriteEnable’, ‘FlsSerialFlashDDRVerify’ (if ‘FlsDDRCalibration’ is 'true') are not different.

ERR092027: The value configured for the parameter ‘FlsExclusiveSelection' and 'FlsCpgExclusiveSelection' should be different.

This error will occur, Since the paramter ' FlsExclusiveSelection' and ‘FlsCpgExclusiveSelection' configured as same value in the container 'FlsGeneral', the paramters should be configured different values.

ERR092034: The path configured for FlsOsCounterRef is not resolved.

This error occurs if the reference path <$counter_ref> provided for the parameter 'FlsOsCounterRef' in the container 'FlsGeneral' is incorrect.

ERR092035: The OsSecondsPerTick value of the referenced Os counter in the container FlsGeneral should be configured and not equal to 0.

This error occurs if the 'Os/OsCounter/OsSecondsPerTick' parameter is not configured or equal to 0.

ERR092800: The container <container name> should be configured.

This error will occur, if any of the mandatory configuration containers is not configured in ECU Configuration Description File. The list of mandatory containers is listed below:

Table 7.6 List of Mandatory containers.

Container name
FlsGeneral
FlsSerialFlashGeneral
FlsPublishedInformation
FlsConfigSet
FlsSerialFlashConfig
FlsSerialFlashRead
FlsSerialFlashReadStatusReg
FlsSerialFlashReadHwId
FlsSerialFlashErase
FlsSerialFlashProgram
FlsSerialFlashWriteEnable
FlsSectorList
FlsSector
FlsDemEventParameterRefs
FlsHyperFlashGeneral
FlsHyperFlashConfig
FlsHyperFlashRead
FlsHyperFlashReadStatusReg
FlsReadStatusSetupCycle
FlsHyperFlashReadHwId
FlsEnterCFI
FlsExitCFI
FlsHyperFlashErase
FlsEraseSetupCycle
FlsHyperFlashProgram
FlsProgramSetupCycle

ERR092801: The number of container <Container name> should be in valid range from <Min value> to <Max value>.

This error occurs, if the number of containers is configured out of valid range.

Table 7.7 Valid range of containers.

Container name	Min value	Max value
FlsGeneral	1	1
FlsSerialFlashGeneral	1	1
FlsPublishedInformation	1	1
FlsConfigSet	1	1
FlsSerialFlashConfig	1	1
FlsSerialFlashRead	1	1

Container name	Min value	Max value
FlsSerialFlashReadStatusReg	1	1
FlsSerialFlashReadHwId	1	1
FlsSerialFlashErase	1	1
FlsSerialFlashProgram	1	1
FlsSerialFlashWriteEnable	1	1
FlsSectorList	1	1
FlsDemEventParameterRefs	1	1
FlsExternalDriver	0	1
FlsHyperFlashGeneral	1	1
FlsHyperFlashConfig	1	1
FlsHyperFlashRead	1	1
FlsHyperFlashReadStatusReg	1	1
FlsHyperFlashReadHwId	1	1
FlsReadStatusSetupCycle	1	*
FlsEnterCFI	1	1
FlsExitCFI	1	1
FlsHyperFlashErase	1	1
FlsEraseSetupCycle	1	*
FlsHyperFlashProgram	1	1
FlsProgramSetupCycle	1	*

ERR092802: The value of parameter <parameter name> in the container <container name> should be in valid range from <Min value> to <Max value>.

This error occurs, if the configured value of parameter is not in valid range.

Table 7.8 Valid range of parameters.

Container name	Parameter name	Min value	Max value
FlsGeneral	FlsBaseAddress	0	4294967295
	FlsDriverIndex	0	254
	FlsExclusiveTimeout	0	65.535
	FlsSemaphoreTimeout	0	65.535
	FlsBurstLengthValue	0	31
	FlsTotalSize	0	4294967295
	FlsCpgExclusiveTimeout	0	65.535
	FlsClockSetConfirmTimeout	0	65.535
	FlsClockSetTendCheckTimeout	0	65.535
	FlsReadTime	0	65.535
	FlsEraseTimeoutValue	0	65.535
	FlsWriteTimeoutValue	0	65.535
FlsPublishedInformation	FlsAcLocationErase	0	4294967295
	FlsAcLocationWrite	0	4294967295
	FlsAcSizeErase	0	4294967295
	FlsAcSizeWrite	0	4294967295
	FlsErasedValue	0	4294967295
	FlsEraseTime	0	65.535
	FlsSpecifiedEraseCycles	0	4294967295
	FlsWriteTime	0	65.535
FlsConfigSet	FlsAcErase	0	4294967295
	FlsAcWrite	0	4294967295
	FlsCallCycle	0	1
	FlsMaxReadFastMode	0	4294967295
	FlsMaxReadNormalMode	0	4294967295
	FlsMaxWriteFastMode	0	4294967295
	FlsMaxWriteNormalMode	0	4294967295
	FlsProtection	0	4294967295
FlsSerialFlashRead	FlsCommand	0	255
	FlsDummyCycle	0	20
	FlsInternalStrobeDelay	0	7
	FlsOptionalData	0	4294967295
FlsSerialFlashReadStatusReg	FlsCommand	0	255
	FlsDummyCycle	0	20

Container name	Parameter name	Min value	Max value
	FlsPositionOfBusyBit	0	7
	FlsPositionOfEraseErrorBit	0	7
	FlsPositionOfProgramErrorBit	0	7
	FlsPositionOfWriteEnableBit	0	7
FlsSerialFlashReadHwId	FlsCommand	0	255
	FlsDummyCycle	0	20
FlsSerialFlashErase	FlsCommand	0	255
	FlsDummyCycle	0	20
FlsSerialFlashProgram	FlsCommand	0	255
	FlsDummyCycle	0	20
FlsSerialFlashWriteEnable	FlsCommand	0	255
	FlsDummyCycle	0	20
FlsSector	FlsNumberOfSectors	0	65535
	FlsPageSize	1	4294967295
	FlsSectorSize	0	4294967295
	FlsSectorStartaddress	0	4294967295
FlsHyperFlashRead	FlsDummyCycle	0	20
FlsHyperFlashReadStatusReg	FlsDummyCycle	0	20
	FlsPositionOfDeviceReadyBit	0	7
	FlsPositionOfEraseErrorBit	0	7
	FlsPositionOfProgramErrorBit	0	7
FlsReadStatusSetupCycle	FlsTransactionAddress	0	4294967295
	FlsTransactionData	0	65535
FlsHyperFlashReadHwId	FlsIDReadAddress0	0	4294967295
	FlsIDReadAddress1	0	4294967295
	FlsIDReadAddress2	0	4294967295
	FlsIDReadAddress3	0	4294967295
	FlsDummyCycle	0	20
FlsEnterCFI	FlsTransactionAddress	0	4294967295
	FlsTransactionData	0	65535
FlsExitCFI	FlsTransactionAddress	0	4294967295
	FlsTransactionData	0	65535
FlsHyperFlashErase	FlsCommand	0	65535
	FlsDummyCycle	0	20
FlsEraseSetupCycle	FlsTransactionAddress	0	4294967295
	FlsTransactionData	0	65535

Container name	Parameter name	Min value	Max value
FlsHyperFlashProgram	FlsDummyCycle	0	20
FlsProgramSetupCycle	FlsTransactionAddress	0	4294967295
	FlsTransactionData	0	65535

ERR092803: Value of parameter <parameter name> in container <container name> should be a valid enumeration literal value.

This error occurs, if Enumeration type parameter is configured out of range of values.

Table 7.9 Valid enumeration values of parameter.

Container name	Parameter name	Enumeration
FlsGeneral	FlsAccess	FLS_SERIAL_FLASH_DEVICE, FLS_HYPER_FLASH_DEVICE
	FlsDeviceName	Depend on device
	FlsExclusiveSelection	MFISLCKR0, MFISLCKR1, MFISLCKR2, MFISLCKR3, MFISLCKR4, MFISLCKR5, MFISLCKR6, MFISLCKR7
	FlsSelectPinGroup	FLS_QSPI0_PIN_GROUP, FLS_QSPI0_QSPI1_PIN_GROUP
	FlsPageBufferWrap	FLS_PAGE_BUFFER_WRAP_256, FLS_PAGE_BUFFER_WRAP_512
	FlsCpgExclusiveSelection	MFISLCKR0, MFISLCKR1, MFISLCKR2, MFISLCKR3, MFISLCKR4, MFISLCKR5, MFISLCKR6, MFISLCKR7
FlsConfigSet	FlsDefaultMode	MEMIF_MODE_FAST, MEMIF_MODE_SLOW
FlsSerialFlashRead	FlsAddressAndDataBitSize	ONE_BIT, FOUR_BITS
	FlsAddressAndDataRate	SDR, DDR
	FlsAddressLength	ONE_BYTE, TWO_BYTES, THREE_BYTES, FOUR_BYTES
	FlsClockDelay	ONE_CYCLE, TWO_CYCLES,

Container name	Parameter name	Enumeration
		THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz, FREQ_44MHz, FREQ_33MHz
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES, EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsOptionalDataLength	NOT_USED, ONE_BYTE, TWO_BYTES, THREE_BYTES, FOUR_BYTES
FlsSerialFlashReadStatusReg	FlsClockDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsDataBitSize	ONE_BIT, FOUR_BITS
	FlsDataRate	SDR, DDR
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz, FREQ_44MHz, FREQ_33MHz

Container name	Parameter name	Enumeration
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES, EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
FlsSerialFlashReadHwId	FlsClockDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsDataBitSize	ONE_BIT, FOUR_BITS
	FlsDataRate	SDR, DDR
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz, FREQ_44MHz, FREQ_33MHz
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES, EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
FlsSerialFlashErase	FlsAddressAndDataBitSize	ONE_BIT, FOUR_BITS
	FlsAddressAndDataRate	SDR, DDR

Container name	Parameter name	Enumeration
	FlsAddressLength	ONE_BYTE, TWO_BYTES, THREE_BYTES, FOUR_BYTES
	FlsClockDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz, FREQ_44MHz, FREQ_33MHz
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES, EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
FlsSerialFlashProgram	FlsAddressAndDataBitSize	ONE_BIT, FOUR_BITS
	FlsAddressAndDataRate	SDR, DDR
	FlsAddressLength	ONE_BYTE, TWO_BYTES, THREE_BYTES, FOUR_BYTES
	FlsClockDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES

Container name	Parameter name	Enumeration
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz, FREQ_44MHz, FREQ_33MHz
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES, EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
FlsSerialFlashWriteEnable	FlsClockDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsDataBitSize	ONE_BIT, FOUR_BITS
	FlsDataRate	SDR, DDR
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz, FREQ_44MHz, FREQ_33MHz
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES, EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES,

Container name	Parameter name	Enumeration
		SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
FlsHyperFlashRead	FlsClockDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz, FREQ_44MHz, FREQ_33MHz
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES, EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
FlsHyperFlashReadStatusReg	FlsClockDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz, FREQ_44MHz, FREQ_33MHz
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES,

Container name	Parameter name	Enumeration
		EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
FlsHyperFlashReadHwId	FlsClockDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz, FREQ_44MHz, FREQ_33MHz
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES, EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
FlsHyperFlashErase	FlsClockDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz,

Container name	Parameter name	Enumeration
		FREQ_44MHz, FREQ_33MHz
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES, EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
FlsHyperFlashProgram	FlsClockDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES
	FlsRpcFrequency	FREQ_160MHz, FREQ_80MHz, FREQ_53MHz, FREQ_40MHz, FREQ_133MHz, FREQ_66MHz, FREQ_44MHz, FREQ_33MHz
	FlsNegationDelay	FIVE_DOT_FIVE_CYCLES, SIX_DOT_FIVE_CYCLES, SEVEN_DOT_FIVE_CYCLES, EIGHT_DOT_FIVE_CYCLES
	FlsNextAccessDelay	ONE_CYCLE, TWO_CYCLES, THREE_CYCLES, FOUR_CYCLES, FIVE_CYCLES, SIX_CYCLES, SEVEN_CYCLES, EIGHT_CYCLES

ERR092038: The parameter <Related parameter> in the container ' FlsGeneral' should be configured when the parameter <parameter> in the container 'FlsGeneral' is enabled.

This error will occur, if related parameter is not configured while the function that used it is enabled.

Table 7.10 Function configuration parameters and related parameters.

Parameters	Related Parameters
FlsExclusiveControl	FlsExclusiveSelection
	FlsExclusiveTimeout
FlsCpgExclusiveControl	FlsCpgExclusiveSelection
	FlsCpgExclusiveTimeout
FlsSemaphore	FlsSemaphoreTimeout

ERR092040: The parameter <ID Address n> in container 'FlsHyperFlashReadHwId' should be configured when the length of 'FlsExpectedHwIdMask' is <value>.

This error will occur, if related parameter is not configured while the length of 'FlsExpectedHwIdMask' is larger in case of FlsAccess is set to FLS_HYPER_FLASH_DEVICE.

ERR092041: The parameters "FlsIDReadAddress0", FlsIDReadAddress1', 'FlsIDReadAddress2' and 'FlsIDReadAddress3' configured with same value.

'This error will occur: When the FlsAccess is FLS_HYPER_FLASH_DEVICE. If the value configured for the parameters 'FlsIDReadAddress0', 'FlsIDReadAddress1', 'FlsIDReadAddress2' and 'FlsIDReadAddress3' in the 'FlsHyperFlashReadHwId' container is not different.

ERR092042: The value configured for the parameter 'FlsSelectPinGroup' must be 'FLS_QSPI0_QSPI1_PIN_GROUP' when FlsAccess is 'FLS_HYPER_FLASH_DEVICE'.

This error will occur, if the parameter 'FlsSelectPinGroup' is not configured as 'FLS_QSPI0_QSPI1_PIN_GROUP' when FlsAccess is 'FLS_HYPER_FLASH_DEVICE'.

ERR092043: The container 'FlsSerialFlashDDRPattern/FlsSerialFlashDDRVerify' in the container 'FlsSerialFlashConfig' should be configured when the parameter 'FlsDDRCalibration' in the container 'FlsSerialFlashGeneral' is enabled.

This error will occur if 'FlsSerialFlashDDRPattern' or 'FlsSerialFlashDDRVerify' container is not configured when FlsDDRCalibration is enabled.

ERR092044: The value configured for the parameter 'FlsPatternAddress' is not included in range of any sector.

This error will occur if parameter 'FlsPatternAddress' is not the address of a sector.

ERR092045: The value configured for the parameter 'FlsNumberOfPattern' must be less than or equal the value ('FlsSectorSize' / 4).

This error will occur if parameter 'FlsNumberOfPattern' is configured higher than the value ('FlsSectorSize' / 4).

ERR092046: The reference path provided for parameter 'Dem Parameter' in container 'FlsDemEventParameterRefs' should not be duplicated with the reference path of other parameter.

This error occurs, if the reference path provided for Dem Parameter in the container FlsDemEventParameterRefs is not unique.

7.7.2 Common Warning Messages

WRN092002: The length of expected HW ID mask configured by the parameter 'FlsExpectedHwIdMask' or the length of expected HW ID configured by the parameter 'FlsExpectedHwId' is greater than 4 bytes. Only first four bytes will be read for HW ID mask or HW ID.

This warning occurs if value of parameter 'FlsExpectedHwIdMask/FlsExpectedHwId' is greater than 0xFFFFFFFF.

WRN092003: If FlsAccess is FLS_SERIAL_FLASH_DEVICE, the parameter 'FlsDDRCalibration' in the container 'FlsSerialFlashGeneral' should be configured as 'true' when the parameter 'FlsAddressAndDataRate' in the container 'FlsSerialFlashRead/FlsSerialFlashSpecificConfig' is configured as 'DDR'.

This warning occurs if value of parameter FlsDDRCalibration is configured as 'false' when the FlsAddressAndDataRate in the container 'FlsSerialFlashRead/FlsSerialFlashSpecificConfig' is configured as 'DDR'.

7.7.3 Common Information Messages

INF092001: The parameter 'FlsJobErrorNotification/FlsJobEndNotification' in the container 'FlsConfigSet' is not configured.

This message occurs if value of parameter 'FlsJobErrorNotification/FlsJobEndNotification' in the container 'FlsConfigSet' is not configured.

7.7.4 Common Deviation List

None.

8.GPT

8.1 Overview

The GPT Driver module provides the service for initializing the whole GPT structure of the microcontroller.
The GPT Driver module comprises two sections, Embedded Software and the Generation Tool, to achieve scalability and configurability.
This chapter describes the features of the GPT Driver Generation Tool. GPT Driver Generation Tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File, and generates GPT Driver C Source and C Header files (Gpt_PBcfg.c, Gpt_Cfg.h and Gpt_Cbk.h).
This chapter contains information on the options, input and output files of the GPT Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.
ECU Configuration Description File contains information about GPT configuration.
This program is built based on Microsoft .NET Framework.
Refer to the following sites about license in detail.
https://msdn.microsoft.com/en-us/library/ms994405.aspx

8.2 GPT Driver Generation Tool Overview

GPT Driver Generation Tool overview is shown below.

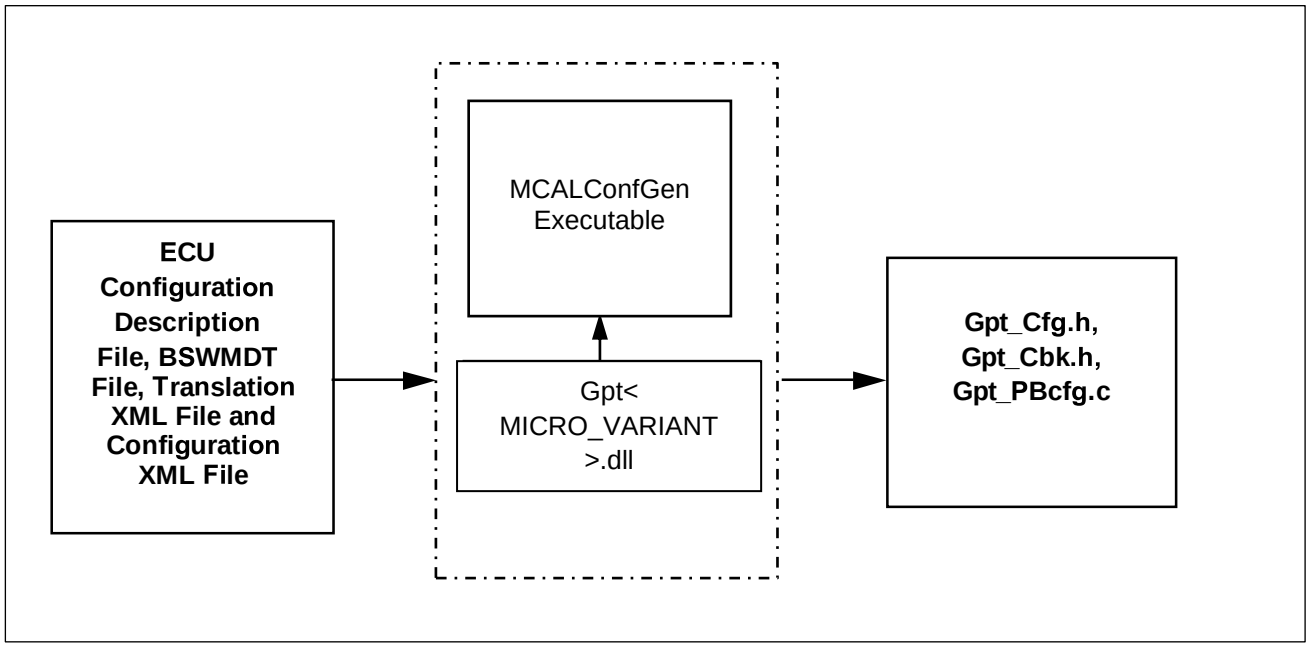


Figure 8.1 Overview of GPT Driver Generation Tool

GPT Driver Generation Tool is a command line tool that extracts and analyzes the configuration details provided in the input file, validates correctness of the data, and provides scalability and configurability for GPT Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input and displays appropriate context-sensitive error messages for wrong input and exits. Tool creates the Log file (Gpt.log) that contains the list of Error/Warning/Information messages in the output directory.

GPT Driver Generation Tool comprises two components, MCALConfGen executable and Gpt<MICRO_VARIANT>.dll. At runtime, the executable loads the dll to generate output files.
Gpt_Cfg.h and Gpt_Cbk.h will be compiled and linked with GPT Driver module. Gpt_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash.
ECU Configuration Description File can be created or edited using ECU Configuration Editor.
For the error-free input file, the tool generates the following output files:
C header Gpt_Cfg.h, Gpt_Cbk.h and C source Gpt_PBcfg.c.
Note: The generation tool returns 1 when error, and 0 when no error.
GPT Driver Generation Tool uses “Common Published Information” from GPT module-specific BSWMDT File. This file should not be updated manually since it is “Static Configuration” file.

8.3 Input Files

GPT Driver Generation Tool accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input. GPT Driver Generation Tool needs information about GPT Driver module. Hence ECU Configuration Description File should contain configuration of GPT Driver module. Generation 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.

Note The detailed explanations about the parameters and containers are found in Parameter Definition File.

8.4 Output Files

GPT Driver Generation Tool generates configuration details in C Header and C Source files (Gpt_PBcfg.c, Gpt_Cfg.h and Gpt_Cbk.h).
The content of each output file is given in the **Table 8.1**:

Table 8.1 Output Files Description

Output File	Details
Gpt_Cfg.h	This file contains the macro definitions for general configuration, GPT API configurations, GPT Channels configured and configuration set handles.
Gpt_Cbk.h	This file contains prototype declarations for GPT callback notification functions.
Gpt_PBcfg.c	This file contains structure for GPT Data Initialization, GPT Channel Configurations and GPT Clock Configurations.

Note Output files generated by GPT Driver Generation Tool should not be modified or edited manually.

8.5 Precautions

ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for:

- R19-11 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain GPT Driver module.
- When a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Gpt.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- When a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Gpt.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension ".trxml".
- Configuration XML File should contain the file extension ".cfgxml".
- 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.
- If the output files generated by GPT Driver Generation 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 GPT Driver Generation Tool. Otherwise, the tool may not produce the expected results or may lead to errors/warnings/information.

Note For more information, please refer to the following appendix file: "ASR_MCAL_Deviation_List.xlsx" in R-CarGen4 AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual.

8.6 Specific 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 Generation Tool.
For the list of common error/warning/information messages for all modules,
refer to “R-Car Gen4 AUTOSAR R19-11 MCAL”.

8.6.1 Specific Error Messages

ERR100069: The reference parameter 'GptWakeupSourceRef' in the container 'GptWakeupConfiguration' should be configured since the parameter 'GptEnableWakeup' in the container 'GptChannelConfiguration' and the parameter 'GptReportWakeupSource’ in the container 'GptDriverConfiguration' are configured as 'TRUE'.
This error will occur, if the reference parameter 'GptWakeupSourceRef' in the container 'GptWakeupConfiguration' is not configured when the parameter 'GptEnableWakeup' in the container 'GptChannelConfiguration' and the parameter 'GptReportWakeupSource' in the container 'GptDriverConfiguration' are configured as true.

ERR100088: The reference path <reference path> provided for the parameter ‘parameter name’ within the container ‘container name’ is incorrect.
This error will occur, if the reference path provided for the following parameters in the respective container is incorrect.

Table 8.2 The parameters need to configure unique

Container	Parameters
GptChannelConfiguration GptTMUClkSrcRef	GptChannelConfiguration GptTMUClkSrcRef

ERR100089: External clock pin is available in TMU channels 3, 4 and 5 as TCLK1; 6, 7 and 8 as TCLK2; 9, 10 and 11 as TCLK3; 12, 13 and 14 as TCLK4 only. External clock should not be configured for the rest channels.
This error will occur, if external clock option is configured by parameter GptChannelClkPrescaler for channels other than channel from 3 to 14 in parameter GptTimerInputSelection.

ERR100005: The value <configured value> configured for the parameter ‘parameter name' present in the container 'container name' should be unique in all instances.
This error will occur, if the value configured for the following parameters in the respective container is not unique in GptChannelConfigSet container. For more details, refer to **Table 8.3**.

Table 8.3 The parameters need to configure unique

Container	Parameters
GptChannelConfiguration	GptTimerInputSelection

GptPredefTimer1Us32BitConfiguration	GptPredefTimerChannelSelection (channel selection)
GptPredefTimer100Us32BitConfiguration	GptPredefTimerChannelSelection (channel selection)

ERR100079: The value "{value of GptGetPredefTimerValueApi}" configured for "{GptGetPredefTimerValueApi}" parameter is conflict with the setting for "{GptPredefTimer1usEnablingGrade}" and "{GptPredefTimer100usEnablingGrade}" parameter. Only either of "{GptPredefTimer1usEnablingGrade}" or "{GptPredefTimer100usEnablingGrade}" is true. We can not configure 2 Predefs at the same instance.

This error occurs, when the "{GptGetPredefTimerValueApi}" parameter is 'true' while all the parameters of "{GptPredefTimer1usEnablingGrade}" and "{GptPredefTimer100us32bitEnable}" are disabled or enabled.

ERR100086: The Predef type '{value of Predef type used}' is used but container for {value of Predef type used} is null or {_value of Predef type is not used} is not null.

This error will occur if Predef timer 1us32bit is used but container for Predef 1us32bit (GptPredefTimer1Us32BitConfiguration) is null, or 100us32bit (GptPredefTimer100Us32BitConfiguration) container is not null.

Similar with Predef 100us32bit.

ERR100090: Dem CDF is not found while 'GptUnintendedInterruptCheck' parameter is configured as true.

This error occurs when Dem CDF is not found in the case of GptUnintendedInterruptCheck is true.

ERR100091: The parameter 'GPT_E_INTERRUPT_CONTROLLER_FAILURE' in '{shortname_container_of_GPT_E_INTERRUPT_CONTROLLER_FAILURE}' container is not configured when 'GptUnintendedInterruptCheck' parameter is true.

This error will occur, if GptUnintendedInterruptCheck is enabled and GPT_E_INTERRUPT_CONTROLLER_FAILURE is not configured.

ERR100092: The referenced path '{value_of_GPT_E_INTERRUPT_CONTROLLER_FAILURE}' of '{GPT_E_INTERRUPT_CONTROLLER_FAILURE}' parameter is not correct in '{shortname_container_of_GPT_E_INTERRUPT_CONTROLLER_FAILURE} container.

This error occurs when the reference path configured for the parameter GPT_E_INTERRUPT_CONTROLLER_FAILURE is wrong in the case of GptUnintendedInterruptCheck is true.

WRN100016: The parameter 'GptChannelClkEdge' should be configured, if the value of the parameter 'GptChannelClkPrescaler' of the container 'GptChannelConfiguration' is configured as <EXTERNAL_CLK>.

This warning will occur, if the parameter 'GptChannelClkPrescaler' has a value of <EXTERNAL_CLK>and the parameter 'GptChannelClkEdge' is not configured.

WRN100018: The configured value of parameter '{GPT_E_INTERRUPT_CONTROLLER_FAILURE}' in '{GptDemEventParameterRefs}' container is ignored, because parameter '{GptUnintendedInterruptCheck}' in container '{GptDriverConfiguration_shortname}' is configured as <false>.

This warning will occur, if parameter 'GPT_E_INTERRUPT_CONTROLLER_FAILURE' is configured and parameter GptUnintendedInterruptCheck is configured as false.

8.6.2 Specific Information Messages

INF100005: The value configured for the parameter 'GptChannelClkEdge' of the container 'GptChannelConfiguration' is not applicable, since the value of the parameter 'GptChannelClkPrescaler' of the same container is NOT configured as <EXTERNAL_CLK>.
This information will be generated, if the clock edge option is configured for a prescaler other than External Clock.

8.6.3 Specific Deviation List

None.

8.7 Common Messages

The following section gives the list of error, warning and information messages displayed by the Generation Tool for common devices.

8.7.1 Common Error Messages

ERR100003: 'GPT Driver' Component is not present in the input file(s)
This error will occur, if GPT Driver component is not present in the input ECU Configuration Description File(s).

ERR100004: The parameter "parameter name" in the container "container name" should be configured.
This error will occur, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File.
For the list of mandatory parameters with respect to container, refer to **Table 8.4**.
Note: GptDeviceName is checked by generic.

Table 8.4 Mandatory configuration parameters:

Container	Parameters
GptDriverConfiguration	GptDevErrorDetect
	GptPredefTimer100us32bitEnable
	GptPredefTimer1usEnablingGrade
	GptReportWakeupSource
	GptAlreadyInitDetCheck
	GptCriticalSectionProtection
	GptDeviceName
	GptUnintendedInterruptCheck
	GptInterruptConsistencyCheck(not supported)
	GptIsrCategory(not supported)
	GptVersionCheckExternalModules
GptClockReferencePoint	GptClockReference
GptConfigurationOfOptApiServices	GptDeinitApi
	GptEnableDisableNotificationApi

	GptTimeElapsedApi
	GptTimeRemainingApi
	GptVersionInfoApi
	GptWakeupFunctionalityApi
	GptGetPredefTimerValueApi
GptChannelConfiguration	GptChannelId
	GptChannelClkEdge
	GptChannelClkPrescaler
	GptTimerInputSelection
	GptChannelMode
	GptChannelTickFrequency
	GptChannelTickValueMax
	GptTMUClkSrcRef

Table 8.5 Mandatory parameters after container is configured by user:

Container	Parameters
GptPredefTimer1Us32BitConfiguration	GptPredefTimerClockSelection
	GptPredefTimerChannelSelection
GptPredefTimer100Us32BitConfiguration	GptPredefTimerClockSelection
	GptPredefTimerChannelSelection

ERR100006: The value configured for the parameter “GptNotification” should follow C syntax <[a-zA-Z][a-zA-Z0-9_]>.

This error will occur, if the value of the parameter ‘GptNotification’ configured in the container ‘GptChannelConfiguration’ does not adhere to C syntax (characters other than (a-z, A-Z, 0-9 or “_”) are configured for GptNotification). The parameter value should start with an alphabet.

ERR100009: In general, the value of '{channelID}' in '{GptChannelConfiguration}' container should be started at <0> and should be sequential without any gaps. The value of '{channelID}' in '{GptChannelConfiguration}' should be '{proposed value}'. At:
FileName: {Name of file}.
ChannelPath: {path of setting}.

This error will occur, if the parameter ‘GptChannelId’ does not start with 0 and is not continuous in the container ‘GptChannelConfiguration’ for any configset. The channel Ids can be configured in any order, but there should not be any gap in the channel Ids configured. Example: if there are 3 channels in a configset, then the possible IDs are 0, 1, and 2 in any order.

ERR100068: The value for parameter 'GptChannelTickValueMax' of 'GptChannelConfiguration' container should not be greater than <65535>(apply for G4MH) and should not be greater than <2^32 - 1>(apply for V4M/V4H/CR52) (, since the parameter 'GptChannelSelection' of 'GptChannelConfiguration' container is selected as <value of GptChannelSelection>.

This error occurs, if the value of the parameter ‘GptChannelTickValueMax’ of GptChannelConfiguration container configured is greater than 65335 when using TAUD and greater than 2^32 – 1 when using TMU.

ERR100072: EcuM CDF is not found or The reference path <reference value of GptWakeupSourceRef> provided for the parameter 'GptWakeupSourceRef' in the container ‘GptWakeupConfiguration’ is incorrect.

This error will occur, if the reference path value provided for the parameter 'GptWakeupSourceRef' in the container 'GptWakeupConfiguration' is incorrect or EcuM CDF is not found when 'GptEnableWakeup' of the

'GptChannelConfiguration' container and 'GptReportWakeupSource' of 'GptDriverConfiguration 'container are configured as true.
Explanation: “incorrect” means that the value input in 'GptWakeupSourceRef' parameter is null or the value is not the same as EcuMWakeupSource of EcuM module.
Note: This error is applicable for **Tmu and Tauj3** unit only.

ERR100071: The parameter 'GptReportWakeupSource' in the container 'GptDriverConfiguration' and the parameter 'GptWakeupFunctionalityApi' in the container 'GptConfigurationOfOptApiServices' should be configured as true while the parameter GptEnableWakeup present in the container 'GptChannelConfiguration' is configured as true.
This error will occur, if the value configured for the parameter ‘GptEnableWakeup’ present in the container 'GptChannelConfiguration' is configured as true, but the parameter 'GptWakeupFunctionalityApi' in container 'GptConfigurationOfOptApiServices' or the parameter 'GptReportWakeupSource' in container 'GptDriverConfiguration' is configured as false.

8.7.2 Common Warning Messages

WRN100001: For all the configured channels, the value of the parameter “GptNotification” of the container “GptChannelConfiguration” should not be <NULL or NULL_PTR or empty string> since the value of the parameter “GptEnableDisableNotificationApi” of “GptConfigurationOfOptApiServices” container is configured as <true>.

This warning will occur, if the value of the parameter ‘GptEnableDisableNotificationApi’ in the container ‘GptConfigurationOfOptApiServices’ is configured as true, and for all configured channels, the parameter ‘GptNotification’ in the ‘GptConfigurationOfOptApiServices’ container is configured as NULL or NULL_PTR or empty string.

8.7.3 Common Information Messages

INF100001: The value configured for the parameter 'GptNotification' in the container 'GptChannelConfiguration' is not considered for the implementation, since the parameter 'GptEnableDisableNotificationApi' in the container 'GptConfigurationOfOptApiServices' is configured as <false>.

This information will occur, if the parameter ‘GptNotification’ in the container is configured, but the parameter ‘GptEnableDisableNotificationApi’ in the container ‘GptConfigurationOfOptApiServices’ is configured as false.

8.7.4 Common Deviation List

None.

9.ICCOM

9.1 Overview

The ICCOM Complex Device Driver module provides the service for initializing the whole CDDICCOM structure of the microcontroller.

The ICCOM Complex Device Driver module comprises of two sections as Embedded Software and the Generation Tool to achieve scalability and configurability.

The document describes the features of the ICCOM Complex Device Driver Generation Tool. ICCOM Complex Device Driver Generation Tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File and generates ICCOM Complex Device Driver C Source and C Header files (CDD_Iccom_PBcfg.c, CDD_Iccom_Cfg.h and CDD_Iccom_Cbk.h).

This document contains information on the options, input and output files of the ICCOM Complex Device Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

ECU Configuration Description File contains information about ICCOM CDD Configuration.

9.2 ICCOM Complex Device Driver Generation Tool Overview

ICCOM Complex Device Driver Generation Tool overview is shown below.

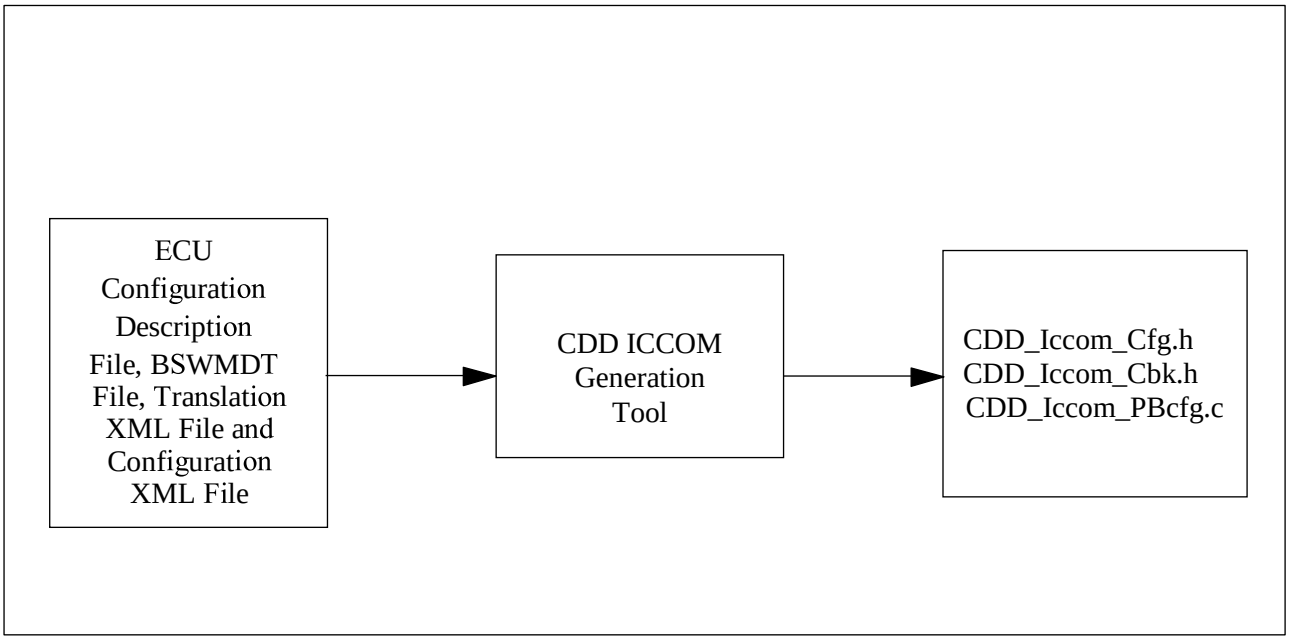


Figure 9.1 Overview of ICCOM Complex Device Driver Generation Tool

ICCOM Complex Device Driver Generation Tool is a command line tool that extracts, analyzes the configuration details provided in the input file and validates correctness of the data and provides scalability and configurability for ICCOM
Complex Device Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input and displays appropriate context sensitive error messages for wrong input and exits.

Tool creates the Log file (CddIccom.log) that contains the list of Error/Warning/Information messages in the output directory.

CDD_Iccom_Cfg.h and CDD_Iccom_Cbk.h will be compiled and linked with ICCOM Complex Device Driver module. CDD_Iccom_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash.

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

For the error free input file, the tool generates the following output files: C Header CDD_Iccom_Cfg.h and CDD_Iccom_Cbk.h, C source CDD_Iccom_PBcfg.c.

Note:

- In case of errors the generation tool returns a 1, in case of no errors the generation tool returns a 0.
- ICCOM Complex Device Driver Generation Tool uses “Common Published Information” from CDD ICCOM module specific BSWMDT File. CDD ICCOM module specific BSWMDT File should not be updated manually since it is “Static Configuration” file.

9.3 Input Files

ICCOM Complex Device Driver Generation Tool accepts ECU Configuration Description File(s), Translation XML File, BSWMDT File and Configuration XML File as input. ICCOM Driver Generation Tool needs information about ICCOM Driver module.
Hence ECU Configuration Description File should contain configuration of ICCOM Complex Device Driver module. Generation 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.

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

9.4 Output Files

ICCOM Complex Device Driver Generation Tool generates configuration details in C Header and C Source files (CDD_Iccom_PBcfg.c, CDD_Iccom_Cfg.h, CDD_Iccom_Cbk.h).

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

Table 9.1 Output Files Description

Output File	Details
CDD_Iccom_Cfg.h	This file contains pre-compile time parameters and handles.
CDD_Iccom_Cbk.h	This file contains notification callback function prototypes.
CDD_Iccom_PBcfg.c	This file contains post-build time parameters.

Note: Output files generated by ICCOM Complex Device Driver Generation Tool should not be modified or edited manually.

9.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain ICCOM Complex Device Driver module.

- If a Translation XML File is provided on the command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddIccom.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddIccom.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension '.trxml'.
- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by ICCOM Complex Device Driver Generation 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 ICCOM Complex Device Driver Generation Tool. Otherwise Tool may not produce the expected results or may lead to errors/warnings/information's.
- The configuration <FILTER-RENESAS> should be configured as ON in Configuration XML File (<Module>.cfgxml) if same module of different vendor is used with this module of Renesas. If <FILTER-RENESAS> is ON, the configuration <FILTER-NAME> will contain the name of vendors which will be processed by <Module name> Generation Tool.

  Example: CDF file contains configuration of Renesas, AUTOSAR and VendorX -> Renesas <Module name> Generation Tool will only process the configuration of Renesas and AUTOSAR if <FILTER-RENESAS> and <FILTER-NAME> are configured as below:

  <FILTER-RENESAS>ON</FILTER-RENESAS>

  <FILTER-NAME>Renesas</FILTER-NAME>

  <FILTER-NAME>AUTOSAR</FILTER-NAME>

Note: For more information, please refer to the following appendix file: "*ASR_MCAL_R-CarGen4_Deviation_List.xlsx*" in R-Car Gen4 AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual

9.6 Specific 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 Generation Tool.

9.6.1 Specific Error Messages

ERR255004: The parameter 'parameter name' in the container 'container name' should be configured.

This error occurs, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File. The list of mandatory parameters with respect to container is listed as Table below:

Table 9.2 Parameter List for ERR255004

Container	Parameters
CddGeneral	CddInstanceId
	CddIccomAlreadyInitDetCheck
	CddIccomCriticalSectionProtection
	CddIccomDevErrorDetect
	CddIccomVersionCheckExternalModules
	CddIccomVersionInfoApi
	CddIccomWriteVerifyCheck
	CddIccomCrNumber
	CddIccomUnintendedInterruptCheck
CddIccomChannel	CddIccomChannelCtaAddress
	CddIccomChannelCtaPartitionSize
	CddIccomChannelId
	CddIccomChannelMfisSelection
	CddIccomChannelTimeoutCounterRef
	CddIccomChannelTimeoutDuration
CddIccomDemEventParameterRefs	CDDICCOM_E_FATAL
	CDDICCOM_E_WRITE_VERIFY_FAILURE
	CDDICCOM_E_INIT_NEGOTIATION
	CDDICCOM_E_TIMEOUT
	CDDICCOM_E_INVALID_ACK
	CDDICCOM_E_INTERRUPT_CONTROLLER_FAILURE
	CDDICCOM_E_DEINIT_NEGOTIATION

ERR255011: The reference parameter <DEM Parameter> of the container <CddIccomDemEventParameterRefs> should be configured, since the value of the parameter <Parameter> of the container <CddGeneral> is configured as <true>.

This error occurs, if the parameter <Parameter> was configured while the reference parameter <DEM Parameter> was not.

Paramter	DEM Parameter
CddIccomWriteVerifyCheck	CDDICCOM_E_WRITE_VERIFY_FAILURE
CddIccomUnintendedInterruptCheck	CDDICCOM_E_INTERRUPT_CONTROLLER_FAILURE

ERR255012: The reference path <reference path> provided for the parameter <Parameter name> within the container <container name> is incorrect.

This error occurs, if the path provided for the following parameters in the respective container is incorrect.

Table 9.3 Parameter List for ERR255012

Container	Parameter
CddIccomDemEventParameterRefs	CDDICCOM_E_FATAL
	CDDICCOM_E_WRITE_VERIFY_FAILURE
	CDDICCOM_E_INIT_NEGOTIATION
	CDDICCOM_E_TIMEOUT
	CDDICCOM_E_INVALID_ACK
	CDDICCOM_E_INTERRUPT_CONTROLLER_FAILURE
	CDDICCOM_E_DEINIT_NEGOTIATION
CddIccomChannel	CddIccomChannelTimeoutCounterRef

ERR255802: The value of parameter 'parameter_name' in container <container name> should be in valid range from <min_value> to <max_value>.

This error occurs, if the numeric parameter is not configured in the valid range.

Table 9.4 Parameter List for ERR255802

Container	Parameter
CddGeneral	CddIccomCrNumber
CddIccomChannel	CddIccomChannelId
	CddIccomChannelCtaAddress
	CddIccomChannelCtaPartitionSize
	CddIccomChannelTimeoutDuration

ERR255803: Value of parameter 'parameter_name', in container <container name> should be a valid enumeration literal value.

This error occurs, if the enumeration parameter is not configured in the valid range.

Table 9.5 Parameter List for ERR255803

Container	Parameter
CddIccomChannel	CddIccomChannelMfisSelection

ERR255016: The reference path <reference path> provided for the parameter <Parameter name> within the container <container name> is duplicated.
This error occurs, if the path provided for the following parameters in the respective container is duplicated.

Table 9.6 Parameter List for ERR255016

Container	Parameter
CddIccomDemEventParameterRefs	CDDICCOM_E_FATAL
	CDDICCOM_E_WRITE_VERIFY_FAILURE
	CDDICCOM_E_INIT_NEGOTIATION
	CDDICCOM_E_TIMEOUT
	CDDICCOM_E_INVALID_ACK
	CDDICCOM_E_INTERRUPT_CONTROLLER_FAILURE
	CDDICCOM_E_DEINIT_NEGOTIATION

9.6.2 Specific Warning Messages

WRN255001: The value configured for the parameter <DEM Parameter> in the container <CddIccomDemEventParameterRefs> should not be configured since the value of the parameter <Parameter> of 'CddGeneral' container is configured as <false>.
This warning occurs, if the value of parameter <DEM Parameter> is configured while the <Parameter> is configured as **false**.

Paramter	DEM Parameter
CddIccomWriteVerifyCheck	CDDICCOM_E_WRITE_VERIFY_FAILURE
CddIccomUnintendedInterruptCheck	CDDICCOM_E_INTERRUPT_CONTROLLER_FAILURE

9.6.3 Specific Information Messages

INF255001: The value of parameter 'CddIccomChannelTimeoutDuration' in the container <container name> is rounded and configured as <channel_timeout_tick>.
This information message provide the actual value in OsTickType to be used for timeout detection.

9.6.4 Specific Deviation List

None.

9.7 Common 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 common errors, warning and information messages displayed by the Generation Tool.

9.7.1 Common Error Messages

ERR255003: <Module> Driver Component is not present in the input file(s).
This error occurs, if <Module> Driver Component mentioned below is not present in the input ECU Configuration Description File(s).

Table 9.7 Modules for ERR255003

Module
Dem
Os
ICCOM

ERR255005: The value configured for the parameter 'CddIccomChannelId' is not unique.
This error occurs, if the channel id configured across the ICCOM channel containers is the same.

ERR255008: The value configured for the parameter <CddIccomChannelNotification> should follow C syntax [a-zA-Z][a-zA-Z0-9_].
This error occurs, if the configured value of CddIccomChannelNotification is not a valid C function name.

ERR255009: The value of parameter 'OsSecondsPerTick' provided for the parameter 'CddIccomChannelTimeoutDuration' in the container <ChannelShortName> should be configured and cannot equal to 0.
This error occurs if the 'OsSecondsPerTick' parameter is not configured or equal to 0.

ERR255013: The value of the parameter 'CddIccomChannelId' configured in container 'CddIccomChanel' should be 'index'. In general per configuration set, the value of CddIccomChannelId' parameter should start with <0> and should be sequential without any gaps.
This error occurs, if the channel ID of channels is not start from 0 and is not sequence.

ERR255014: The value configured for the parameter 'CddIccomChannelNotification' is duplicated.
This error occurs, if the channel notification function was configured in the same name.

ERR255015: The value configured for the parameter 'CddIccomChannelMfisSelection' is duplicated.
This error occurs, if the more than 1 channel select the same MFIS channel.

ERR255800: The container 'CddIccomChannel', 'CddIccomDemEventParameterRefs' should be configured.
This error occurs, if the container is not configured when its lower multiplicity is not '0'.

ERR255801: The number of container ‘container_name', should be in valid range from <min_value> to <max_value>.

This error occurs, if the number of the container in not configured in the valid range.

9.7.2 Common Warning Messages

WRN255004: The parameter <Parameter name> in the container <container name> is not presented.

This warning occurs, if any configured parameter is not belong to container.

9.7.3 Common Information Messages

None

9.7.4 Common Deviation List

None.

10.MCU

10.1 Overview

The MCU Driver Component provides services for basic microcontroller initialization, power down functionality, reset, and microcontroller-specific functions required by other SPAL components.
The MCU Driver Component comprises two sections, Embedded Software and Generation Tool, to achieve scalability and configurability.

The chapter describes the features of the MCU Driver Component Code Generation Tool. MCU Driver Component Code Generation Tool is a command line tool that extracts information from ECU Configuration Description File, BSWMDT File and generates MCU Driver Component C source and C header files (Mcu_PBcfg.c, Mcu_Cfg.h).

This program is built based on Microsoft .NET Framework.
Refer to the following site about license in detail.
https://msdn.microsoft.com/en-us/library/ms994405.aspx

10.2 MCU Driver Component Code Generation Tool Overview

MCU Driver Component Code Generation Tool overview is shown below.

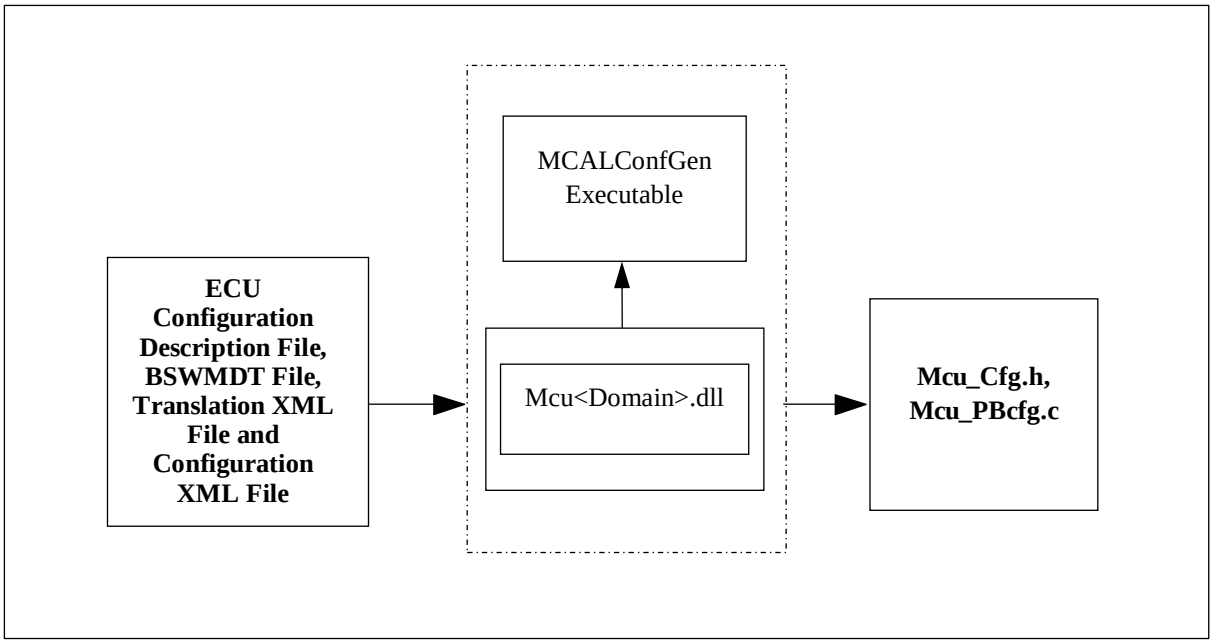


Figure 10.1 Overview of MCU Driver Component Code Generation Tool

MCU Driver Component Code Generation Tool is a command line tool that extracts and analyzes the configuration details provided in the input file, validates correctness of the data, and provides scalability and configurability for MCU Driver Component. It accepts ECU Configuration Description File(s), Translation XML File, BSWMDT File and Configuration XML File as input and displays appropriate context-sensitive error messages for wrong input and exits. Tool creates the Log file (Mcu.log) that contains the list of Error/Warning/Information messages in the output directory.

MCU Driver Component Code Generation Tool comprises 2 components, MCALConfGen executable, McuRCAR.dll. At runtime, the executable component loads the dll to generate output files.

For the error-free input file, the tool generates the following output files: Mcu_Cfg.h and Mcu_PBcfg.c.

Mcu_Cfg.h will be compiled and linked with MCU Driver Component. Mcu_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash.

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

Note:

The MCU Driver Component Code Generation Tool returns 1 when error, and 0 when no error.

MCU Driver Component Code Generation Tool uses “Common Published Information” from MCU module specific BSWMDT File. This file should not be updated manually since it is “Static Configuration” file.

10.3 Input Files

MCU Driver Component Code Generation Tool accepts ECU Configuration Description File(s), Translation XML File, BSWMDT File and Configuration XML File as input. MCU Driver Component Code Generation Tool needs information about MCU Driver Component. Hence ECU Configuration Description File should contain configuration of MCU Driver module. MCU Driver Component Code Generation 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.

Note: The detailed explanations about the parameters and containers are found in Parameter Definition File.

10.4 Output Files

MCU Driver Component Code Generation Tool generates configuration details in C Header and C Source files Mcu_Cfg.h and Mcu_PBcfg.c.

The content of each output file is given in Table 10-1

Table 10-1 Output Files Description

Output File	Details
Mcu_Cfg.h	This file contains pre-compile time parameters and handles.
Mcu_PBcfg.c	This file contains post-build configuration data.

Note: Output files generated by MCU Driver Component Code Generation Tool should not be modified or edited manually.

10.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain MCU Driver Component and DEM component related configuration.
- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Mcu.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Mcu.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension '.trxml'.
- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by MCU Driver Component Code Generation 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 must be provided as input to the MCU Driver Component Code Generation Tool. Otherwise, the tool may not produce the expected results or may lead to errors/warnings/information messages.
- Parameter "McuInitClock" cannot be configured as "FALSE". It's Enable/Disable parameter.
- Parameter "McuNoPll" cannot be configured as "TRUE". It's Enable/Disable parameter.

Note: For more information, please refer to the following appendix file: "*V4M_Deviation_List.xlsx*" in *R-CarGen4 AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual*.

- The MCU Driver Component Code Generation Tool only supports the maximum frequency of clock RPCφ is 320 MHz and clock RPCD2φ is 160 MHz. Any adjustments causing the greater values of those clocks are prohibited.

10.6 Specific 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 MCU Driver Component Code Generation Tool.

For the list of common error/warning/information messages for all modules, refer to *"[1] R-CarGen4 AUTOSAR R19-11 MCAL User's Manual Getting Started"*.

10.6.1 Specific Error Messages

ERR101004: The parameter 'parameter name' in the container 'container name' should be configured.
This error will occur, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File.

The mandatory parameters with respect to container are listed in from Table 10-2 **to** Table 10-12 **The list of mandatory parameters:**

Table 10-2 The list of mandatory parameters (1/11)

Container Name	Parameter
McuGeneralConfiguration	McuDevErrorDetect
	McuGetRamStateApi
	McuInitClock
	McuNoPll
	McuPerformResetApi
	McuVersionInfoApi
	McuVersionCheckExternalModules
	McuCriticalSectionProtection
	McuSwResetCall
	McuDeviceName
	McuIsrCategory
McuModuleConfiguration	McuClockSrcFailureNotification
	McuNumberOfMcuModes
	McuRamSectors
	McuResetSetting

McuModeSettingConf	McuPSOWakeUpFactor
	McuC4PowerOn
	McuTimeOutC4ModeChange
	McuModeType
	McuMode
McuClockSettingConfig	McuClockSettingId
	McuClockStabilityWaitingTime
McuCpuMainSysClk	McuClockName
	McuClockValue

Table 10-3 The list of mandatory parameters (2/11)

Container Name	Parameter
McuDebugTracePortClk	McuClockName
	McuClockSelection
	McuClockValue
McuDebugTraceBusClk	McuClockName
	McuClockSelection
	McuClockValue
McuDebugClk	McuClockName
	McuClockSelection
	McuClockValue
McuZR0Clk	McuClockName
	McuClockDivider
	McuClockValue
McuZR1Clk	McuClockName
	McuClockDivider
	McuClockCtrlSel
	McuClockValue
McuZR2Clk	McuClockName
	McuClockDivider

	McuClockCtrlSel
	McuClockValue
McuZB3Clk	McuClockName
	McuClockSelection
	McuClockValue
McuZC0Clk	McuClockName
	McuClockDivider
	McuClockCtrlSel
	McuClockValue
McuZC1Clk	McuClockName
	McuClockDivider
	McuClockCtrlSel
	McuClockValue
McuZC2Clk	McuClockName
	McuClockDivider
	McuClockCtrlSel
	McuClockValue
McuZC3Clk	McuClockName
	McuClockDivider
	McuClockCtrlSel
	McuClockValue
McuZDClk	McuClockName
	McuClockDivider
	McuClockValue
McuZGClk	McuClockName
	McuClockDivider
	McuClockCtrlSel
	McuClockValue
McuSDSRCClk	McuClockName

	McuClockSelection
	McuClockValue
McuSD0HClk	McuClockName
	McuClockCtrlSel
	McuClockSelection
	McuClockValue
McuSD0Clk	McuClockName
	McuClockCtrlSel
	McuClockSelection
	McuClockValue

Table 10-4 The list of mandatory parameters (3/11)

Container Name	Parameter
McuRPCClk	McuClockName
	McuClockCtrlSel
	McuClockSelection
	McuClockValue
McuRPCD2Clk	McuClockName
	McuClockCtrlSel
	McuClockValue
McuMSOClk	McuClockName
	McuClockCtrlSel
	McuClockDivider
	McuClockValue
McuCANFDClk	McuClockName
	McuClockCtrlSel
	McuClockDivider
	McuClockValue
McuCSIClk	McuClockName
	McuClockCtrlSel

	McuClockDivider
	McuClockValue
McuPOSTClk	McuClockName
	McuClockCtrlSel
	McuClockDivider
	McuClockValue
McuPOST2Clk	McuClockName
	McuClockCtrlSel
	McuClockDivider
	McuClockValue
McuPOST3Clk	McuClockName
	McuClockCtrlSel
	McuClockDivider
	McuClockValue
McuPOST4Clk	McuClockName
	McuClockCtrlSel
	McuClockDivider
	McuClockValue
McuDSIEXTClk	McuClockName
	McuClockCtrlSel
	McuClockDivider
	McuClockValue
McuZXClk	McuClockName
	McuClockValue

Table 10-5 The list of mandatory parameters (4/11)

Container Name	Parameter
McuSVVIPClk	McuClockName
	McuClockValue
McuSVIRClk	McuClockName

	McuClockValue
McuS0HSCClk	McuClockName
	McuClockValue
McuS0VCClk	McuClockName
	McuClockValue
McuS0VIOClk	McuClockName
	McuClockValue
McuIMPAD1Clk	McuClockName
	McuClockValue
McuIMPAD4Clk	McuClockName
	McuClockValue
McuIMPBClk	McuClockName
	McuClockName
McuVCBUSClk	McuClockName
	McuClockValue
McuVIOBUSClk	McuClockName
	McuClockValue
McuRCLKClk	McuClockName
	McuClockSelection
	McuClockValue
McuIMPASRCClk	McuClockName
	McuClockCtrlSel
	McuClockValue
McuIMPBSRCClk	McuClockName
	McuClockCtrlSel
	McuClockValue
McuVIOSRCClk	McuClockName
	McuClockCtrlSel
	McuClockValue

McuVCSRCClk	McuClockName
	McuClockCtrlSel
	McuClockValue
McuFRAYClk	McuClockName
	McuClockCtrlSel
	McuClockDivider
	McuClockValue
McuADGHClk	McuClockName
	McuClockCtrlSel
	McuClockValue

Table 10-6 The list of mandatory parameters (5/11)

Container Name	Parameter
McuPllClockSetting	McuMainOsc
McuPll1ClockSetting	McuPllCircuitEnable
	McuPllFrequency
McuPllStopConditions (McuPll1ClockSetting)	McuPllStopByA3IR
	McuPllStopBy3DGB
	McuPllStopByA3ISP0
	McuPllStopByA3ISP1
	McuPllStopByA3VIP0
	McuPllStopByA3VIP1
	McuPllStopByA3VIP2
	McuPllStopByA2E0D1
	McuPllStopByA2E0D0
	McuPllStopByA3DUL
McuPll2ClockSetting	McuPllCircuitEnable
	McuMultiplicationRatio
	McuFreqDitherMode
	McuDownSpreadModuleDepth
	McuFractionalMultiplication
	McuSSCGModulationFreq
	McuPllFrequency
McuPllStopConditions (McuPll2ClockSetting)	McuPllStopByA3IR
	McuPllStopBy3DGB
	McuPllStopByA3ISP1
	McuPllStopByA3ISP0
	McuPllStopByA3VIP0
	McuPllStopByA3VIP1
	McuPllStopByA3VIP2
	McuPllStopByA3DUL

McuPll3ClockSetting	McuMultiplicationRatio
	McuFreqDitherMode
	McuFractionalMultiplication
	McuPllFrequency
	McuPllCircuitEnable
McuPllStopConditions (McuPll3ClockSetting)	McuPllStopByA3IR
	McuPllStopBy3DGB
	McuPllStopByA3ISP1
	McuPllStopByA3ISP0
	McuPllStopByA3VIP0
	McuPllStopByA3VIP1
	McuPllStopByA3VIP2
	McuPllStopByA2E0D1
	McuPllStopByA2E0D0
	McuPllStopByA3DUL
McuPll4ClockSetting	McuPllCircuitEnable
	McuMultiplicationRatio
	McuFreqDitherMode
	McuDownSpreadModuleDepth
	McuFractionalMultiplication
	McuSSCGModulationFreq
	McuPllFrequency

Table 10-7 The list of mandatory parameters (6/11)

Container Name	Parameter
McuPllStopConditions (McuPll4ClockSetting)	McuPllStopByA3IR
	McuPllStopBy3DGB
	McuPllStopByA3ISP1
	McuPllStopByA3ISP0
	McuPllStopByA3VIP0
	McuPllStopByA3VIP1
	McuPllStopByA3VIP2
	McuPllStopByA2E0D1
	McuPllStopByA2E0D0
	McuPllStopByA3DUL
McuPll5ClockSetting	McuPllCircuitEnable
	McuPllFrequency
McuPllStopConditions (McuPll5ClockSetting)	McuPllStopByA3IR
	McuPllStopBy3DGB
	McuPllStopByA3ISP1
	McuPllStopByA3ISP0
	McuPllStopByA3VIP0
	McuPllStopByA3VIP1
	McuPllStopByA3VIP2
	McuPllStopByA2E0D1
	McuPllStopByA2E0D0
	McuPllStopByA3DUL
McuPll6ClockSetting	McuPllCircuitEnable
	McuMultiplicationRatio
	McuFreqDitherMode
	McuDownSpreadModuleDepth
	McuFractionalMultiplication
	McuSSCGModulationFreq

	McuPllFrequency
McuPllStopConditions (McuPll6ClockSetting)	McuPllStopByA3IR
	McuPllStopBy3DGB
	McuPllStopByA3ISP1
	McuPllStopByA3ISP0
	McuPllStopByA3VIP0
	McuPllStopByA3VIP1
	McuPllStopByA3VIP2
	McuPllStopByA2E0D1
	McuPllStopByA2E0D0
	McuPllStopByA3DUL

Table 10-8 The list of mandatory parameters (7/11)

Container Name	Parameter
McuPll7ClockSetting	McuPllCircuitEnable
	McuPllFrequency
McuPllStopConditions (McuPll7ClockSetting)	McuPllStopByA3IR
	McuPllStopBy3DGB
	McuPllStopByA3ISP1
	McuPllStopByA3ISP0
	McuPllStopByA3DUL
	McuPllStopByA3VIP0
	McuPllStopByA3VIP1
	McuPllStopByA3VIP2
	McuPllStopByA2E0D1
	McuPllStopByA2E0D0
McuModuleClockSupplySetting	McuIMPPSCClockSupplyEnable
	McuIMPDMAC0ClockSupplyEnable
	McuIMP1ClockSupplyEnable
	McuIMP0ClockSupplyEnable
	McuSPMCClockSupplyEnable
	McuIMPCNNClockSupplyEnable
	McuISP0ClockSupplyEnable
	McuUMFL0ClockSupplyEnable
	McuSMPS0ClockSupplyEnable
	McuSMPO0ClockSupplyEnable
	McuRGXClockSupplyEnable
	McuADGClockSupplyEnable
	McuSPMIClockSupplyEnable
	McuIMPSLVClockSupplyEnable
	McuIMPDTAClockSupplyEnable
	McuIMPDMAC1ClockSupplyEnable

	McuAVB2ClockSupplyEnable
	McuAVB1ClockSupplyEnable
	McuAVB0ClockSupplyEnable
	McuCSITOP0ClockSupplyEnable
	McuCANFDClockSupplyEnable
	McuDSITXLINK0ClockSupplyEnable
	McuDOC2CHClockSupplyEnable
	McuDIS0ClockSupplyEnable
	McuCSITOP1ClockSupplyEnable
	McuIMS1ClockSupplyEnable
	McuIMS0ClockSupplyEnable
	McuIMR1ClockSupplyEnable

Table 10-9 The list of mandatory parameters (8/11)

Container Name	Parameter
McuModuleClockSupplySetting	McuIMR0ClockSupplyEnable
	McuFRAY00ClockSupplyEnable
	McuFCPVD0ClockSupplyEnable
	McuFCPCSClockSupplyEnable
	McuPWMClockSupplyEnable
	McuMSI5ClockSupplyEnable
	McuMSI4ClockSupplyEnable
	McuIVCP1EClockSupplyEnable
	McuISPCS1ClockSupplyEnable
	McuISPCS0ClockSupplyEnable
	McuINTTPClockSupplyEnable
	McuVIN01ClockSupplyEnable
	McuVIN00ClockSupplyEnable
	McuTPU0ClockSupplyEnable
	McuVSPD0ClockSupplyEnable
	McuVIN17ClockSupplyEnable
	McuVIN16ClockSupplyEnable
	McuVIN15ClockSupplyEnable
	McuVIN14ClockSupplyEnable
	McuVIN13ClockSupplyEnable
	McuVIN12ClockSupplyEnable
	McuVIN11ClockSupplyEnable
	McuVIN10ClockSupplyEnable
	McuVIN07ClockSupplyEnable
	McuVIN06ClockSupplyEnable
	McuVIN05ClockSupplyEnable
	McuVIN04ClockSupplyEnable
	McuVIN03ClockSupplyEnable

	McuVIN02ClockSupplyEnable
	McuPFC2ClockSupplyEnable
	McuPFC1ClockSupplyEnable
	McuVSPX0ClockSupplyEnable
	McuFCPVX0ClockSupplyEnable
	McuMTIClockSupplyEnable
	McuCSBRGIRA2ClockSupplyEnable
	McuCSBRGIRA3ClockSupplyEnable
	McuTSNClockSupplyEnable
	McuCVE3ClockSupplyEnable
	McuCVE2ClockSupplyEnable

Table 10-10 The list of mandatory parameters (9/11)

Container Name	Parameter
McuModuleClockSupplySetting	McuCVE1ClockSupplyEnable
	McuCVE0ClockSupplyEnable
	McuFCPRCClockSupplyEnable
	McuSSIClockSupplyEnable
	McuSSIUClockSupplyEnable
	McuADVFSCClockSupplyEnable
	McuCR0ClockSupplyEnable
	McuINTAPClockSupplyEnable
	McuI2C3ClockSupplyEnable
	McuI2C2ClockSupplyEnable
	McuI2C1ClockSupplyEnable
	McuI2C0ClockSupplyEnable
	McuHSCIF3ClockSupplyEnable
	McuHSCIF2ClockSupplyEnable
	McuHSCIF1ClockSupplyEnable
	McuHSCIF0ClockSupplyEnable
	McuRTDM1ClockSupplyEnable
	McuRTDM0ClockSupplyEnable
	McuRPCClockSupplyEnable
	McuPCIE0ClockSupplyEnable
	McuMSI3ClockSupplyEnable
	McuMSI2ClockSupplyEnable
	McuMSI1ClockSupplyEnable
	McuMSI0ClockSupplyEnable
	McuIRQCClockSupplyEnable
	McuTMU4ClockSupplyEnable
	McuTMU3ClockSupplyEnable
	McuTMU2ClockSupplyEnable

	McuTMU1ClockSupplyEnable
	McuTMU0ClockSupplyEnable
	McuSYDM2ClockSupplyEnable
	McuSYDM1ClockSupplyEnable
	McuSDHI0ClockSupplyEnable
	McuSCIF4ClockSupplyEnable
	McuSCIF3ClockSupplyEnable
	McuSCIF1ClockSupplyEnable
	McuSCIF0ClockSupplyEnable
	McuUCMTClockSupplyEnable
	McuTSCClockSupplyEnable

Table 10-11 The list of mandatory parameters (10/11)

Container Name	Parameter
McuModuleClockSupplySetting	McuPFC0ClockSupplyEnable
	McuCMT3ClockSupplyEnable
	McuCMT2ClockSupplyEnable
	McuCMT1ClockSupplyEnable
	McuCMT0ClockSupplyEnable
	McuWDTClockSupplyEnable
	McuWCRC3ClockSupplyEnable
	McuWCRC2ClockSupplyEnable
	McuWCRC1ClockSupplyEnable
	McuWCRC0ClockSupplyEnable
	McuKCRC4ClockSupplyEnable
	McuFSOClockSupplyEnable
	McuCRC3ClockSupplyEnable
	McuCRC2ClockSupplyEnable
	McuCRC1ClockSupplyEnable
	McuCRC0ClockSupplyEnable
	McuKCRC7ClockSupplyEnable
	McuKCRC6ClockSupplyEnable
	McuKCRC5ClockSupplyEnable
	McuAESACCWrapperClockSupplyEnable
	McuAESACC0ClockSupplyEnable
	McuAESACC1ClockSupplyEnable
	McuAESACC2ClockSupplyEnable
	McuAESACC3ClockSupplyEnable
	McuAESACC4ClockSupplyEnable
	McuAESACC5ClockSupplyEnable
	McuAESACC6ClockSupplyEnable
	McuAESACC7ClockSupplyEnable

	McuPAPDEBUGClockSupplyEnable
	McuTSCClockSupplyEnable
	McuVDSP1MSTPCR28b30ClockSupplyEnable
	McuVDSP0MSTPCR28b21ClockSupplyEnable
	McuPAPSDMACClockSupplyEnable
	McuPAPTOPClockSupplyEnable
	McuPAPBUSClockSupplyEnable
	McuVDSP1MSTPCR28b02ClockSupplyEnable
	McuVDSP0MSTPCR28b01ClockSupplyEnable
	McuDDR1DDR0ClockSupplyEnable

Table 10-12 The list of mandatory parameters (11/11)

Container Name	Parameter
McuClockReferencePoint	McuClockReferencePointFrequency
McuRamSectorSettingConf (*1)	McuRamDefaultValue
	McuRamSectionBaseAddress
	McuRamSectionSize
	McuRamSectionWriteSize
McuResetReasonConf	McuResetReason
McuPowerOnResetConf	McuResetReason
McuSWDTResetConf	McuResetReason
McuRWDTResetConf	McuResetReason
McuMultiResetConf	McuResetReason
McuResetUndefinedConf	McuResetReason
McuNoneResetConf	McuResetReason
McuSwResetConf	McuResetReason

Note *1: If these containers are configured, then the respective parameters which are mandatory should be configured.

ERR101020: Calculated value for PLL<n>VCO <calculated PLL value>Hz should be equal to value of McuPllFrequency <value of McuPllFrequency>Hz in McuPll<n>ClockSetting.

This error will occur, if the calculated value of 'PLL<n>VCO' is not equal to 'McuPllFrequency' value when McuPllCircuitEnable is configured as 'true'.

ERR101021: McuClockCtrlSel' in container 'McuSD0HClk' should be configured as 'STOP_ID_1' since McuClockSelection is selected as <selected value of McuClockSelection>.

This error will occur, if 'McuClockCtrlSel' is not selected as 'STOP_ID_1' when McuClockSelection value is one of CLK_SDSRC_DIV4_ID_2, CLK_SDSRC_DIV8_ID_3, CLK_SDSRC_DIV16_ID_4.

ERR101022: Calculated value of <Clock container> '<actual_freq>Hz' is not matched with <value of McuClockValue>Hz of parameter 'McuClockValue'.

This error will occur, if the calculated value of clock domain is not equal the expected clock value which input from parameter McuClockValue when McuClockCtrlSel does not exist or configured as ACTIVATE_ID_0.

ERR101023: McuClockSelection value <value of McuSD0Clk.McuClockSelection> in container McuSD0Clk should not be selected since McuClockSelection in container McuSD0HClk is selected as <value of McuSD0HClk.McuClockSelection>.

This error will occur, if the value of McuSD0Clk.McuClockSelection is CLK_SD0H_DIV2_ID_0 and the value of McuSD0HClk.McuClockSelection is in (CLK_SDSRC_DIV1_ID_0, CLK_SDSRC_DIV4_ID_2, CLK_SDSRC_DIV8_ID_3, CLK_SDSRC_DIV16_ID_4).

ERR101024: The calculated value of SSCG modulation frequency (Fmod) '<actual Fmod value>' KHz should be in range 20 KHz to <Fmod maximum> KHz.

This error will occur, if the actual value of SSCG modulation frequency (Fmod) is out of range 20 KHz to <Fmod maximum> KHz when McuPllCircuitEnable is configured as 'true', McuFreqDitherMode is configured as 'DITHERED_FREQUENCY_MODE_DOWN_ID_6' and McuDownSpreadModuleDepth is bigger than zero. The <Fmod maximum> value is mentioned in ***Table 10-13***

Table 10-13 The maximum value of Fmod

Item	Value						
McuDownSpreadModuleDepth	0	20	40	60	80	100	120
Fmod maximum of PLL2	Don’t care	60	60	51.9	48.9	31.1	25.9
Fmod maximum of PLL4	Don’t care	60	60	60	60	60	60
Fmod maximum of PLL6	Don’t care	60	60	52.5	39.4	31.5	26.2

ERR101060: The value of parameter 'McuFractionalMultiplication' in the container 'McuPll<n>ClockSetting' must be configured as 0 when the value of parameter 'McuFreqDitherMode' configured as INTEGER_FIXED_FREQUENCY_MODE_ID_0

This error will occur, if the value of parameter 'McuFractionalMultiplication' in the container 'McuPll<n>ClockSetting' was configured difference from 0 when the value of parameter 'McuFreqDitherMode' was configured as INTEGER_FIXED_FREQUENCY_MODE_ID_0

ERR101025: The reference path <path> provided for the parameter ‘Parameter Name’ within the container ‘Container Name’ is incorrect.

This error will occur, if the paths provided for the reference parameters mentioned in ***Table 10-14*** are incorrect:

Table 10-14 The Parameters List of DEM Container

Container Name	Parameter
McuDemEventParameterRefs	MCU_E_CLOCK_FAILURE
	MCU_E_MODE_TRANSITION_FAILURE
	MCU_E_SOFTWARE_RESET_FAILURE

10.6.2 Specific Warning Messages

None.

10.6.3 Specific Information Messages

INF101002: All clock modules belong to McuPll<n>ClockSetting are not operated since the parameter McuPllCircuitEnable in the container McuPll<n>ClockSetting is configured as False.

This information will occur if parameter McuPllCircuitEnable in the container McuPll<n>ClockSetting is configured as False and related clock configuration for these containers are not operated:

Table 10-15 List Containers related to McuPll<n>ClockSetting

McuPll<n>ClockSetting	Container Name
McuPll1ClockSetting	McuDebugTracePortClk
	McuDebugTraceBusClk
	McuZXClk
	McuSVVIPClk
	McuSVIRClk
	McuS0HSCClk
	McuIMPAD1Clk
	McuIMPAD4Clk
	McuIMPBClk
McuPll2ClockSetting	McuZC0Clk
	McuZC1Clk
	McuZC2Clk
	McuZC3Clk
McuPll3ClockSetting	McuZB3Clk
McuPll4ClockSetting	McuZGClk
McuPll5ClockSetting	McuSDSRCClk
	McuSD0HClk
	McuSD0Clk

	McuRPCClk
	McuRPCD2Clk
	McuMSOClk
	McuCANFDClk
	McuCSIClk
	McuPOSTClk
	McuPOST2Clk
	McuPOST3Clk
	McuPOST4Clk
	McuDSIEXTClk
	McuSASYNCRTClk
	McuSASYNCPERD1Clk
	McuSASYNCPERD2Clk
	McuSASYNCPERD4Clk
	McuVCBUSClk
	McuVIOBUSClk
McuPll6ClockSetting	McuZR0Clk
	McuZR1Clk
	McuZR2Clk

10.6.4 Specific Deviation List

None.

10.7 Common Messages

10.7.1 Common Error Messages

ERR101005: The value for parameter 'McuInitClock' present in container 'McuGeneralConfiguration' should not be configured as <false>.
This error will occur, if the value of the parameter 'McuInitClock' present in the container 'McuGeneralConfiguration' is configured as false.

ERR101006: The value for parameter 'McuNoPll' present in container 'McuGeneralConfiguration' should not be configured as <true>.
This error will occur, if the value of the parameter 'McuNoPll' present in the container 'McuGeneralConfiguration' is configured as true.

ERR101007: The value of parameter 'McuClockSettingId' should start with <0> and should be sequential without any gaps.
This error will occur, if the values of the following parameters are not unique or not started from <0> with sequential order in the respective container per configuration set.

ERR101008: The value configured for parameter 'McuRamSectionBaseAddress' in container 'McuRamSectorSettingConf' is not scoped for System RAM, RT-VRAM0 MIRRORED or RT-VRAM1 area.
This error will occur, if the configured for parameter 'McuRamSectionBaseAddress' in container 'McuRamSectorSettingConf' is not scoped for System RAM, RT-VRAM1 or RT-VRAM0 MIRRORED area.

ERR101009: The value configured for parameter 'McuRamSectionSize' in container 'McuRamSectorSettingConf' is configured initial size from 'McuRamSectionBaseAddress'. But 'McuRamSectionSize' is over of selected 'McuRamSectionBaseAddress' area.
This error will occur, if the value of McuRamSectionSize exceeds the selected McuRamSectionBaseAddress area.

ERR101010: The value configured for parameter 'McuRamSectionSize' in container 'McuRamSectorSettingConf' is less than minimum write size of selected 'McuRamSectionBaseAddress' area.
This error will occur, if the value configured for the parameter 'McuRamSectionSize' in the container 'McuRamSectorSettingConf' is less than the minimum write size of the selected 'McuRamSectionBaseAddress' area.

ERR101011: The value configured for parameter 'McuRamSectionBaseAddress' in container 'McuRamSectorSettingConf' is not aligned for selected 'McuRamSectionBaseAddress' area.
This error will occur, if the value configured for the parameter 'McuRamSectionBaseAddress' in the container 'McuRamSectorSettingConf' is not aligned in the selected 'McuRamSectionBaseAddress' area.

ERR101012: The value configured for parameter 'McuRamSectionSize' in container 'McuRamSectorSettingConf' is not aligned for selected 'McuRamSectionBaseAddress' area.
This error will occur, when the value configured for the parameter 'McuRamSectionSize' in the container 'McuRamSectorSettingConf' is not aligned in the selected 'McuRamSectionBaseAddress' area.

ERR101029: The value of parameter 'McuMode' should start with <0> and should be sequential without any gaps.

This error will occur, if the value of the following parameters is either not unique or it is not started from <0> with sequential order in respective container per configuration set.

10.7.2 Common Warning Messages

None.

10.7.3 Common Information Messages

None.

10.7.4 Common Deviation List

None.

11.PORT

11.1 Overview

The PORT Driver component provides the service to initialize the whole PORT structure of the microcontroller.

The PORT Driver Component comprises two sections, Embedded Software and the Generation Tool, to achieve scalability and configurability.

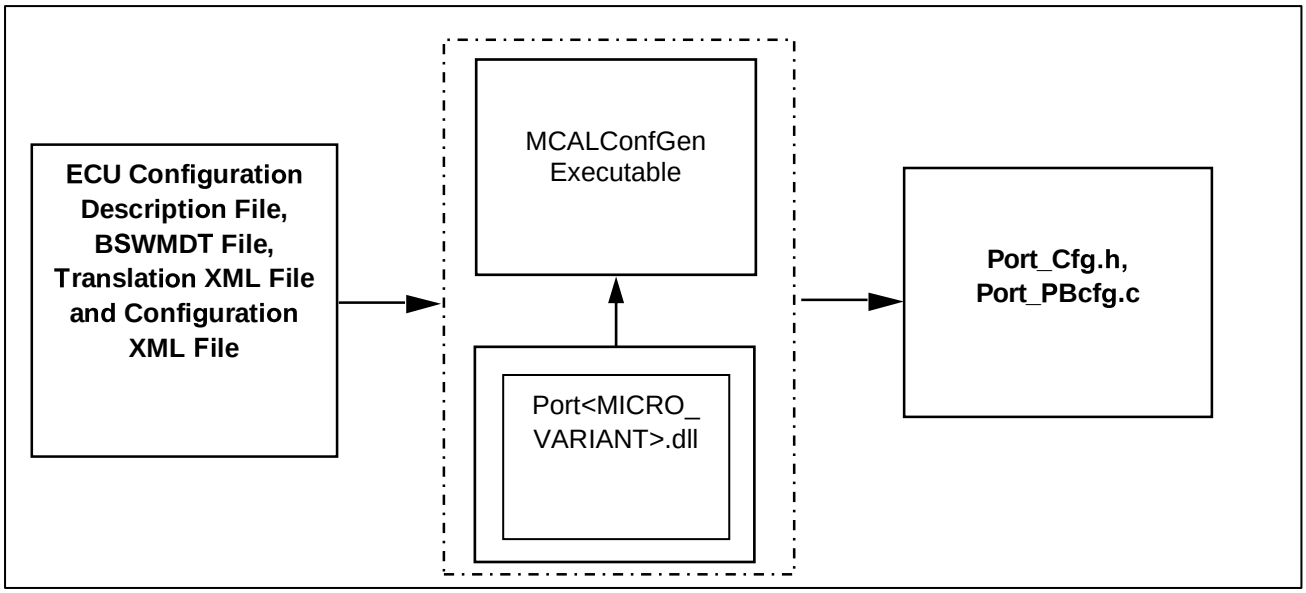
The chapter describes the features of the PORT Driver Generation Tool. This tool is a command line tool that extracts information from ECU Configuration Description File, and generates PORT Driver C Source and C Header files (Port_PBcfg.c and Port_Cfg.h).

This chapter contains information on the options, input and output files of the PORT Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

ECU Configuration Description File contains information about PORT General Configuration, and Port Pin Configurations.

11.2 PORT Driver Generation Tool Overview

PORT Driver Generation Tool overview is shown in **Figure 11.1**



.

Figure 11.1 Overview of PORT Driver Generation Tool

PORT Driver Generation Tool is a command line tool that provides scalability and configurability for PORT Driver component. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input, and generates the C Header and C Source files (Port_Cfg.h and Port_PBcfg.c)
Port_Cfg.h will be compiled and linked with PORT Driver Component. Port_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash.
ECU Configuration Description File can be created or edited using ECU Configuration Editor.

PORT Driver Generation Tool extracts and analyzes the configuration details provided in the input file, and validates correctness of the data. Tool displays appropriate context-sensitive error messages for wrong input and exits. Tool creates the Log file (Port.log) that contains the list of Error/Warning/Information messages in the output directory.

PORT Driver Generation Tool comprises two components, MCALConfGen executable and Port<MICRO_VARIANT>.dll. At runtime, the executable loads the dll to generate output files.

For the error-free input file, the tool generates the following output files: C header Port_Cfg.h and C source Port_PBcfg.c file names.

- The generation tool returns 1 when error, and 0 when no error.
- PORT Driver Generation Tool uses “Common Published Information” from PORT module-specific BSWMDT File. This file should not be updated manually, since it is “Static Configuration” file.

11.3 Input Files

PORT Driver Generation Tool accepts ECU Configuration Description File(s), Configuration XML file, BSWMDT File, and Translation XML File as input. This tool needs information about PORT Driver component. Hence ECU Configuration Description File should contain configuration of PORT Driver component. Generation 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.

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

11.4 Output Files

PORT Driver Generation Tool generates configuration details in C Header and C Source files (Port_Cfg.h and Port_PBcfg.c).

The content of each output file is given in the **Table 11-1:**

Table 11-1 Output Files Description

Output File	Details
Port_Cfg.h	This file contains the macro definitions for general configuration, total number of Port Pins configured and configuration set handles. It also includes the Port Pin handles for each configuration set.
Port_PBcfg.c	This file contains structure for Port Configuration, Port Pin Configuration, Chatting Filter Initial Value, and Port Pin Initialization during runtime.

Note: Output files generated by PORT Driver Generation Tool should not be modified or edited manually.

11.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.

- The input file must contain PORT Driver component.

- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Port.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.

- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Port.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.

- Translation XML File should contain the file extension '.trxml

- Configuration XML File should contain the file extension '.cfgxml'.

- 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.

- If the output files generated by PORT Driver Generation 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 PORT Driver Generation Tool. Otherwise, the tool may not produce the expected results or may lead to "errors/warnings/information messages".

- If no certain port filter is configured in this Port Module, the device specific default-settings will take effect on this filter.

- If user selects the alternate signal in the port group container, then the respective port filter container should be configured. For example: If signal NMI is selected in the port group container, the respective filter group container has to be configured.

- In post-build time, sub containers of PortFilterGroupConfig containers should not be added or deleted.

- Edge/Level settings for External Interrupt (INTP) signals can be overwritten by ICU component to change the default activation type and the type of activation at run-time.

- The value for parameter “PortPinInitialMode” from each Port Pin if they are configured as <DIO>, so the configuration for Port Pin Direction will be ignored.

Note For more information, please refer to the following appendix file: "ASR_MCAL_R-CarGen4_Deviation_List.xlsx" in R-Car Gen4 AUTOSAR R19-11 MCAL User’s Manual Driver Component Embedded User's Manual.

11.6 Specific Message

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 errors, warning and information messages displayed by the Generation Tool.

11.6.1 Specific Error Messages

None.

11.6.2 Specific Warning Messages

None.

11.6.3 Specific Information Messages

None.

11.6.4 Specific Deviation List

None.

11.7 Common 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 common errors, warning and information messages displayed by the Generation Tool.

11.7.1 Common Error Messages

ERR124003: 'PORT Driver' Component is not present in the input file(s).

This error will occur, if PORT Driver Component is not present in the input ECU Configuration Description File(s).

ERR124004: The parameter 'parameter name' in the container 'container name' should be configured.
This error occurs, if any of the mandatory configuration parameter(s) mentioned in **Table 11-2** is (are) not configured in ECU Configuration Description File

Table 11-2 Parameter List for ERR124004

Parameter Name		Container Name	Remark
-		PortConfigSet	-
PortCriticalSectionProtection		PortGeneral	-
PortDevErrorDetect			-
PortDeviceName			-
PortMaxMode			-
PortSetPinDefaultDirectionApi			-
PortSetPinDirectionApi			-
PortSetPinModeApi			-
PortSetToDioAltModeApi			-
PortVersionCheckExternalModules			-
PortVersionInfoApi			-
PortExclusiveControl			-
PortExclusiveSelection			-
PortExclusiveTimeout			-
PortFUSEMonitoringApi			-
PortUnintendedModuleStopCheck			-
PortEcucPartitionRef			This parameter is not used for implementation
PortPinDioAltModeChangeable		Port Group Where in 'Port Group container': PortGroup<Group number name> for PortPin<Pin number>. E.g. PortGroup0, PortPin0, PortGroup1 PortPin1, PortPin0, etc.	-
PortPinDirection			-
PortPinDirectionChangeable			-
PortPinInitialMode			-
PortPinLevelValue			-
PortPinModeChangeable			-
PortPinPullOption			-
PortPinPullControl			-
PortPinPolaritySelect			-
PortPinSensitiveInterrupt			-
PortPinDetectionInterrupt			-
PortPinOutDataSelect			-
PortFilterClockFrequency		PortFilterGroupConfig Chattering Filter Container PortChatteringFilterGroup<Filter Group number> Ex: PortChatteringFilterGroup0, PortChatteringFilterGroup1… etc.	-
PortChatteringFilterInput <n>	PortChatteringFilterInput Option		n: 0-31

ERR124006: The value for parameter 'PortPinDirectionChangeable' from 'Container name' container should not be configured as <true> as the value for parameter 'PortSetPinDirectionApi' in the container 'PortGeneral' is configured as <false>.
File Name: CDF of parameter
Path: path to parameter

This error will occur, if the parameter 'PortSetPinDirectionApi' in the container 'PortGeneral' is configured as false, and the parameter 'PortPinDirectionChangeable' as true in at least one of the 'Port Group container' container.

ERR124008: The value for parameter 'PortPinDioAltModeChangeable' from 'Port container' container should not be configured as <true> as the value for parameter 'PortSetToDioAltModeApi' in the container 'PortGeneral' is configured as <false>.
File Name: CDF of parameter
Path: path to parameter

This error will occur, if the parameter 'PortPinDioAltModeChangeable' is configured as true in Port Pin container, and 'PortSetToDioAltModeApi' as false in Port General container.

ERR124010: The value for parameter ' PortPinModeChangeable' from 'Port container' container should not be configured as <true> as the value for parameter ' PortSetPinModeApi' in the container 'PortGeneral' is configured as <false>.
File Name: CDF of parameter
Path: path to parameter

This error will occur, if the parameter 'PortPinModeChangeable' is configured as true in Port Pin container, and 'PortSetPinModeApi' as false in Port General container.

ERR124012: The value for parameter 'PortSetPinDefaultDirectionApi' from 'Port General' container should not be configured as <true> as the value for parameter 'PortSetPinDirectionApi' in the container 'Port General' is configured as <false>.
File Name: CDF of parameter
Path: path to parameter

This error will occur, if the parameter 'PortSetPinDefaultDirectionApi' is configured as true, and 'PortSetPinDirectionApi' as false in Port General container.

ERR124014: The reference path <path> provided for the DEM parameter 'Parameter Name' within the container 'Container Name' is incorrect or is not configured.

This error will occur, if path provided for the DEM parameter in "PortDemEventParameterRefs" of **Table 11-3** is incorrect.

ERR124015: The parameter <DEM parameter> in the container 'PortDemEventParameterRefs' should be configured with a valid reference path when the parameter <parameter> in the container 'PortGeneral' is enabled.

This error will occur, if path provided for the DEM parameter in "PortDemEventParameterRefs" of **Table 11-3** is incorrect or is not configured when the parameter <parameter> in the container 'PortGeneral' is enabled.

Table 11-3 Parameter list for ERR124014/ERR124015/ERR124017

Parameter Name	Container Name	Remark
PORT_E_GET_CONTROL_FAILURE	PortDemEventParameterRefs	-
PORT_E_FUSE_MONITORING_FAILURE		-
PORT_E_UNINTENDED_MODULE_STOP_FAILURE		-

ERR124016: The parameter 'PortOTPMON0ExpectedValue' and 'PortOTPMON3ExpectedValue' in the container 'PortGeneral' should be configured when the parameter 'PortFUSEMonitoringApi' in the container 'PortGeneral' is enabled.

This error will occur, if parameter 'PortOTPMON0ExpectedValue' or 'PortOTPMON3ExpectedValue in 'PortGeneral' are not configured when the parameter 'PortFUSEMonitoringApi' in the container 'PortGeneral' is enabled.

ERR124017: The reference path <path> provided for the parameters 'parameter name' and 'parameter name' in the container 'container name' must not be the same.

This error will occur, if the reference path provided for the parameter in **Table 11-3** is duplicated with the other parameters in the container 'PortDemEventParameterRefs'. These values must not be the same.

ERR124021: The container name 'PortGroup8' or 'PortChatteringFilterGroup8' and parameters in this container must be configured when the parameter 'PortUnintendedModuleStopCheck' is configured as True.

This error occurs, if the container 'PortGroup8' or 'PortChatteringFilterGroup8' and the parameters in this container are not configured when the parameter 'PortUnintendedModuleStopCheck' is enabled.

11.7.2 Common Warning Messages

WRN124003: The configured value for the parameter 'PORT_E_GET_CONTROL_FAILURE/PORT_E_FUSE_MONITORING_FAILURE/PORT_E_UNINTENDED_MODULE_STOP_FAILURE' in the container 'PortDemEventParameterRefs' should not be configured since the value of the parameter 'PortExclusiveControl/PortFUSEMonitoringApi/PortUnintendedModuleStopCheck' is disabled.

This warning will occur, if the value of parameter 'PORT_E_GET_CONTROL_FAILURE/PORT_E_FUSE_MONITORING_FAILURE/ PORT_E_UNINTENDED_MODULE_STOP_FAILURE ' is configured while 'PortExclusiveControl/PortFUSEMonitoringApi/PortUnintendedModuleStopCheck' is configured as false.

WRN124004: The parameter 'PortPinDirection' of container 'Port Group container' should not be configured as <PORT_PIN_OUT>, since the parameter 'PortPinInitialMode' of the same 'Port Group container' container is configured as an Input type mode. The value for parameter 'PortPinDirection' is considered as <PORT_PIN_IN>.

This warning will occur, if the value configured for the parameter 'PortPinDirection' is Output, and that of the

parameter ‘PortPinInitialMode’ is Input type mode. The value of the parameter ‘PortPinDirection’ is considered as Input.

WRN124005: The parameter 'PortPinDirection' of container 'Port Group container' should not be configured as <PORT_PIN_IN>, since the parameter 'PortPinInitialMode' of the same 'Port Group container' container is configured as an Output type mode. The value for parameter 'PortPinDirection' is considered as <PORT_PIN_OUT>.

This warning will occur, if the value configured for the parameter ‘PortPinDirection’ is Input, and that of the parameter ‘PortPinInitialMode’ is Output type mode. The value of the parameter ‘PortPinDirection’ is considered as Output.

WRN124006: The parameter 'PortPinInitialMode' of 'PortPinX in PortGroupY' is configured as the same value of the parameter 'PortPinInitialMode' of 'PortPinA in PortGroupB'. In this case, the values of the parameter 'PortPinInitialMode' of 'PortPinX in PortGroupY' and 'PortPinA in PortGroupB' should be different.

This warning will occur, if the same peripheral mode value of the parameter PortPinInitialMode is configured on multiple pins.

11.7.3 Common Information Messages

None.

11.7.4 Common Deviation List

None.

12.SPI

12.1 Overview

The SPI Driver component provides the service to initialize the whole SPI structure of the microcontroller.

The SPI Driver Component comprises two sections, Embedded Software and the Generation Tool, to achieve scalability and configurability.

The chapter describes the features of the SPI Driver Generation Tool. This Tool is a command line tool that extracts information from ECU Configuration Description File, and generates SPI Driver C Source and C Header files (Spi_Cfg.h, Spi_PBcfg.c and Spi_Lcfg.c).

This chapter contains information on the options, input and output files of the SPI Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

ECU Configuration Description File contains information about SPI configuration.

This program is built based on Microsoft .NET Framework.
Refer to the following sites about license in detail.
https://msdn.microsoft.com/en-us/library/ms994405.aspx

12.2 SPI Driver Generation Tool Overview

SPI Driver Generation Tool overview is shown below.

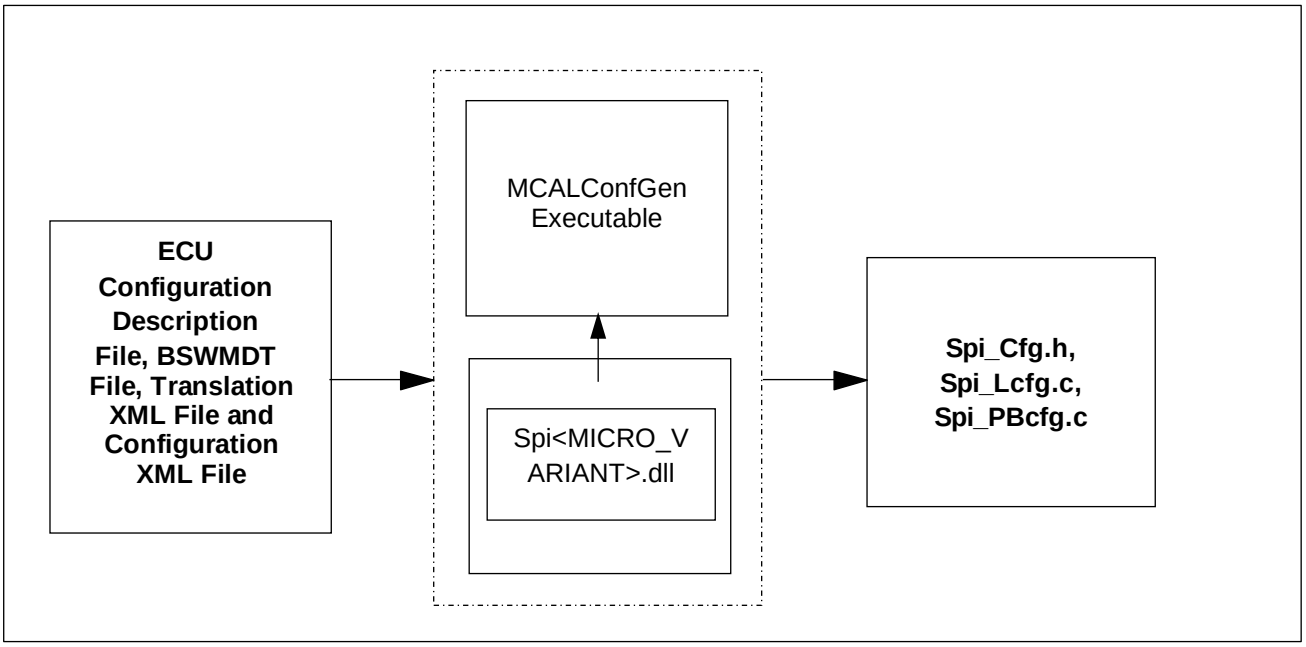


Figure 12.1 Overview of SPI Driver Generation Tool

SPI Driver Generation Tool is a command line tool that extracts and analyzes the configuration details provided in the input file, validates correctness of the data, and provides scalability and configurability for SPI Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input and displays appropriate context-sensitive error messages for wrong input and exits. Tool creates the Log file Spi.log that contains the list of Error/Warning/Information messages in the output directory.
SPI Driver Generation Tool comprises 2 components, MCALConfGen executable and Spi<MICRO_VARIANT>.dll. At runtime, the executable component loads the dll to generate output files.
For the error-free input file, the tool generates the following output files: Spi_Lcfg.c, Spi_PBcfg.c and Spi_Cfg.h. Spi_Cfg.h will be compiled and linked with SPI Driver Component. Spi_Lcfg.c and Spi_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash.

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

Note

- The generation tool returns 1 when error, and 0 when no errors.
- SPI Driver Generation Tool uses “Common Published Information” from SPI module-specific BSWMDT File. This file should not be updated manually since it is “Static Configuration” file.

12.3 Input Files

SPI Driver Generation Tool accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input. SPI Driver Generation Tool needs information about SPI Driver module. Hence ECU Configuration Description File should contain configuration of SPI Driver module. Generation Tool ignores any other AUTOSAR component configured in the ECU Configuration Description File. This file can be generated using configuration editor.

This file must comply with AUTOSAR standard ECU Configuration Description File format.

Note:

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

12.4 Output Files

SPI Driver Generation Tool generates configuration details in C Header and C Source files (Spi_Lcfg.c, Spi_PBcfg.c and Spi_Cfg.h).

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

Table 12.1 Output Files Description

Output File	Details
Spi_Cfg.h	This file contains pre-compile time parameters and handles.
Spi_PBcfg.c	This file contains post-build time parameters.
Spi_Lcfg.c	This file contains structures of link time parameters.

Note:

Output files generated by SPI Driver Generation Tool should not be modified or edited manually.

12.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain SPI Driver module.
- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the generation tool will get the file Spi.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the generation tool will get the file Spi.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension '.trxml'.
- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by SPI Driver Generation 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 must be provided as input to the SPI Driver Generation Tool. Otherwise the tool may not produce the expected results or may lead to errors/warnings/information messages.

Note For more information, please refer to the following appendix file: "ASR_MCAL_R-CarGen4_Deviation_List.xlsx" in R-Car Gen4 AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual.

12.6 Specific 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 Generation Tool.

For list of common error/warning/information messages for all modules, refer to “R-Car Gen4 AUTOSAR R19-11 MCAL User’s Manual Getting Started”.

12.6.1 Specific Error Messages

ERR083004: The parameter 'parameter name' in the container 'container name' having short name 'short name' should be configured.
This error occurs, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File.
The list of mandatory parameters with respect to the container is listed below:

Table 12.2 Parameter List for ERR083004 (1/3)

Container	Parameters
SpiGeneral	SpiAlreadyInitDetCheck
	SpiCancelApi
	SpiChannelBuffersAllowed
	SpiCriticalSectionProtection
	SpiDevErrorDetect
	SpiDeviceName
	SpiDmaMode
	SpiDmaRedundancyCheck
	SpiEcucPartitionRef
	SpiEnableClkImmediateValue
	SpiEnablePersistentHwConfiguration
	SpiEnableSeqStartNotification
	SpiEnableSyncSeqEndNotification
	SpiForceCancelApi
	SpiHwStatusApi
	SpiInterruptibleSeqAllowed
	SpiIsrCategory
	SpiKernelEcucPartitionRef
	SpiLevelDelivered
	SpiMainFunctionPeriod
	SpiSupportConcurrentAsyncTransmit
	SpiSupportConcurrentSyncTransmit
	SpiTimeoutWaitingTime

Container	Parameters
	SpiUnintendedInterruptCheck
	SpiUserCallbackHeaderFile
	SpiVersionCheckExternalModules
	SpiVersionInfoApi
	SpiWriteVerifyCheck

Table 12.3 Parameter List for ERR083004 (2/3)

Container	Parameters
SpiDriver	SpiClockFrequencyRefImmediateValue
	SpiClockFrequencyRef
	SpiMaxChannel
	SpiMaxJob
	SpiMaxSequence
SpiChannel	SpiChannelId
	SpiChannelType
	SpiDataWidth
	SpiDefaultData
	SpiEbMaxLength
	SpiIbNBuffers
	SpiTransferStart
SpiChannelList	SpiChannelIndex
	SpiChannelAssignment
SpiExternalDevice	SpiBaudrate
	SpiBaudrateConfiguration
	SpiCsHoldTiming
	SpiCsIdentifier
	SpiCsPolarity
	SpiCsSelection
	SpiCsSetupTime
	SpiDataShiftEdge
	SpiDeviceEcucPartitionRef
	SpiEnableCs
	SpiHwUnit
	SpiInputClockSelect
	SpiMasterMode
	SpiOutputControlSelect
	SpiShiftClockIdleLevel
	SpiClk2CsCount
	SpiBaudrate
	SpiBaudrateConfiguration
	SpiCsHoldTiming
	SpiCsIdentifier
	SpiCsPolarity

Table 12.4 Parameter List for ERR083004 (3/3)

Container	Parameters
SpiJob	SpiDeviceAssignment
	SpiHwUnitSynchronous
	SpiJobEndNotification
	SpiJobId
	SpiJobPriority
	SpiPortPinSelect
	SpiDeviceAssignment
SpiSequence	SpiInterruptibleSequence
	SpiJobAssignment
	SpiSeqEndNotification
	SpiSeqStartNotification
	SpiSequenceId
SpiDma	SpiTxDmaChannel
	SpiRxDmaChannel
	SpiDmaHwUnit
SpiPublishedInformation	SpiMaxHwUnit

Note:

If the container ‘SpiDma' is configured, then the respective parameters in those mandatory containers should be configured.

Table 12.5 Container List for ERR083004

Container
SpiDriver
SpiChannel
SpiExternalDevice
SpiJob
SpiChannellist
SpiSequence
SpiGeneral
SpiPublishedInformation

This container list is used to check if the mandatory parameter belongs to mandatory container or not.

ERR083020: The reference path <path> provided for the parameter ‘parameter name’ in the container ‘container name’, having short name 'short name' is incorrect.

This error occurs, if incorrect reference provided for any of the reference parameters (SPI_E_HARDWARE_ERROR, SPI_E_DATA_TX_TIMEOUT_FAILURE, SPI_E_INTERRUPT_CONTROLLER_FAILURE, SPI_E_WRITE_VERIFY_FAILURE, SpiDeviceAssignment, SpiChannelAssignment, SpiJobAssignment, SpiClockFrequencyRef).

ERR083021: The reference path <path> provided for the parameters 'parameter name' and 'parameter name' in the container 'container name' having short name 'short name' should be unique.
This error occurs, if the reference path provided for the parameters SPI_E_HARDWARE_ERROR, SPI_E_DATA_TX_TIMEOUT_FAILURE, SPI_E_INTERRUPT_CONTROLLER_FAILURE, and SPI_E_WRITE_VERIFY_FAILURE is not unique.

ERR083027: The value of parameter 'SpiPortPinSelect' in the container 'SpiJob' should not be configured as <value of the parameter SpiPortPinSelect> since the value of the parameter 'SpiCsSelection' present in the container 'SpiExternalDevice' is configured as <value of the parameter SpiCsSelection>.
This error will occur, if the value of the parameter SpiPortPinSelect in the container SpiJob is configured as SYNC/SS1/SS2 and the parameter SpiCsSelectionpresent in the container SpiExternalDevice is configured as CS_VIA_GPIO.

ERR083028: The value of parameter 'SpiPortPinSelect' in the container 'SpiJob' should not be configured as <value of the parameter SpiPortPinSelect> since the value of the parameter 'SpiCsSelection' present in the container 'SpiExternalDevice' is configured as <value of the parameter SpiCsSelection>.
This error will occur, if the value of the parameter SpiPortPinSelect in the container SpiJob is configured as Port group related pins and the parameter SpiCsSelection present in the container SpiExternalDevice is configured as CS_VIA_PERIPHERAL_ENGINE.

ERR083029: The value of the parameter 'SpiDataWidth' in the container 'SpiChannel' having short name 'short name' is not in the range of <8/16/24/32> since the value of the parameter 'SpiHwUnit' in the container 'SpiExternalDevice' having short name 'short name' is configured as MSIOFn.
This error occurs, if the value of the parameter 'SpiDataWidth' in the container 'SpiChannel' is not in the range of <8/16/24/32>, and that of the parameter 'SpiHwUnit' in the container 'SpiExternalDevice' is configured as MSIOF<n>. Here <n> is integer numbers e.g. 0, 1, etc.

ERR083050: The parameter 'SpiPortPinSelect' in the container 'SpiJob' having short name <short name> configured should be unique value, since the value of parameter 'SpiHwUnit' present in the container 'SpiExternalDevice' having short name <short name> is configured as <value of the parameter SpiHwUnit>.
This error occurs, if the value of the parameter 'SpiHwUnit' present in the container 'SpiExternalDevice' is configured as MSIOF<n>, and more than one unique value is configured for the parameter 'SpiPortPinSelect' in the container 'SpiJob'. Here n is integer number e.g. 0, 1, etc.

ERR083058: Sum of the delays specified through DTDL and SYNCDL must be an integer in external device 'external device name' in the configuration set 'configuration set name'.
This error will occur, if parameters SpiCsSetupTime and SpiCsHoldTiming present in the container SpiExternalDevice are not configured as sum of them become an integer value.

ERR083059: SpiInputClockSelect is set to MSO_PERE_WCLK while SpiBaudrateConfiguration is not set to ONE in external device 'external device name' in the configuration set 'configuration set name'.
The error occurs if value of the parameter 'SpiBaudrateConfiguration' in the container 'SpiExternalDevice' is not 1 when SpiInputClockSelect is set to MSO_PERE_WCLK

ERR083090: The value of the <parameter> across the containers 'SpiExternalDevice container short name' and 'SpiExternalDevice container short name' should be same when 'SpiEnablePersistentHwConfiguration' in container SpiGeneral having short name <short name> is configured as TRUE and they are referred the same SpiHwUnit <SpiHwUnit>.
Here <n> is integer numbers e.g. 0, 1, etc.
The error occurs if the value of the one parameters: SpiDataShiftEdge, SpiShiftClockIdleLevel, SpiInputClockSelect, SpiBaudrateConfiguration, SpiCsSetupTime, SpiCsHoldTiming, SpiOutputControlSelect, SpiCsPolarity in the SpiExternalDevice container are not same across the external devices mapped to the same SpiHwUnit MSIOF<n>, they

are associated with jobs having same chip select line and HW channel when SpiEnablePersistentHwConfiguration is configured as true.
(Note: SpiCsPolarity having additional condition is SpiCsSelection is configured as CS_VIA_PERIPHERAL_ENGINE).

ERR083091:

- SpiChannel container: **When 'SpiEnablePersistentHwConfiguration' is configured as TRUE in container SpiGeneral having short name <short name>, the value of the parameter 'parameter name' should be same across the <short name> containers since they are associated with jobs having same chip select line SpiHwUnit <value of SpiHwUnit>.**

- SpiJob containers: **When 'SpiEnablePersistentHwConfiguration' is configured as TRUE in container SpiGeneral having short name <short name>, the value of the parameter 'parameter name' should be same across the <short name> containers since they are associated with jobs having same chip select line SpiHwUnitSelection <value of SpiHwUnitSelection>.**

Here <n> is integer numbers e.g. 0, 1, etc.
The error occurs if the value of one of the parameters: SpiPortPinSelect in the SpiJob container and SpiTransferStart, SpiDataWidth in the SpiChannel container are not same across the external devices mapped to the same SpiHwUnit MSIOF<n>, they are associated with jobs having same chip select line and HW channel when SpiEnablePersistentHwConfiguration is configured as true.
All the parameters in the channel container related to the registers:
MSIOFnSITMDR1, MSIOFnSIRMDR1, MSIOFnSICTR, MSIOFnSITSCR.

ERR083103: The value of parameter 'SpiPortPinSelect' in the container 'SpiJob' having short name <short name> should be configured as <value of the parameter 'SpiPortPinSelect'> since the value of the parameter 'SpiMasterMode' in the container 'SpiExternalDevice' having short name <short name> is configured as <false> for hardware unit <HWIP Type><n>.
Here n is integer numbers e.g. 0, 1, etc.
<HWIP Type><n>: MSIOFn (n = 0..5)
This error occurs, if the value of the parameter ‘SpiPortPinSelect' in the container 'SpiJob' is configured as <CS Type> and the value of the parameter 'SpiMasterMode’ is configured as <false> for hardware unit <HWIP Type><n>.

Here n is integer numbers e.g. 0, 1, etc.

<CS Type>: SS1/SS2/Not configured

<HWIP Type><n>: MSIOFn (n = 0..5)

12.6.2 Specific Warning Messages

None.

12.6.3 Specific Information Messages

INF083001: Calculated SPI baud rate for job 'SpiJob' in configuration set 'SpiDriver' should be equal to <Calculated Baud rate Hz>.
This information occurs to provide the calculated SPI baud rate for job (SpiJob) in configuration set SpiDriver.
If user configure SpiEnableClkImmediateValue as true, PCLK (E2x) or CLK_MSPI (U2x) will be SpiClockFrequencyRefImmediateValue.
If user configure SpiEnableClkImmediateValue as false, PCLK (E2x) or CLK_MSPI (U2x) will be referred peripheral clock from MCU.
The calculation of baud rate is done as follows:
MSIOF Baud rate = MSO_PERE_WCLK / (2^m *(SpiBaudrateConfiguration + 1)).
SpiInputClockSelect m

MSIOF:
MSO_PERE_WCLK: 0
MSO_PERE_WCLK_DIVBY_2 : 1
MSO_PERE_WCLK_DIVBY_4 : 2
MSO_PERE_WCLK_DIVBY_8 : 3
MSO_PERE_WCLK_DIVBY_16 : 4
MSO_PERE_WCLK_DIVBY_32 : 5

INF083006: .The number of loop-counts for updating the status of data transmission in status register of MSIOF is configured as <Calculated Time out loop count>.
This information will occur to provide the SPI time out loop-count.
The calculation of timeout loop count is done as follows:
SPI_TIMEOUT_COUNT = int (SpiTimeoutWaitingTimes)

12.6.4 Specific Deviation List

None.

12.7 Common 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 Generation Tool.

For list of common error/warning/information messages for all modules, refer to "R-Car Gen4 AUTOSAR R19-11 MCAL User's Manual Getting Started".

12.7.1 Common Error Messages

ERR083003:

- For SPI Driver: **SPI Component is not present in the input file(s).**

This error occurs, if SPI Driver component is not present in the input ECU Configuration Description File(s).

- For MCU Driver: **Mcu Driver Component is not present in the input file(s) when 'SpiEnableClkImmediateValue' is configured as false.**

This error occurs, if MCU Driver component is not present in the input ECU Configuration Description File(s) when SpiEnableClkImmediateValue is configured as False.

- For DEM: **Dem Driver Component is not present in the input file(s) when Dem parameter is not empty.**

This error occurs, if DEM component is not present in the input ECU Configuration Description File(s) and DEM parameter is not empty.

ERR083005: In general per configuration set, the value of 'SpiChannelId' parameter in the container 'SpiChannel' having short name <short name> should start with <0> and should be sequential without any gaps.
This error occurs, if the value of the parameter 'SpiChannelId' present in the container 'SpiChannel' is not sequential or has gaps or starts with zero in configuration set in the ECU Configuration Description File.

ERR083006: In general per configuration set, the value of 'SpiJobId' parameter in the container 'SpiJob' having short name <short name> should start with <0> and should be sequential without any gaps.
This error occurs, if the value of the parameter 'SpiJobId' present in the container 'SpiJob' is not sequential or has gaps or starts with zero in configuration set in the ECU Configuration Description File.

ERR083007: In general per configuration set, the value of 'SpiSequenceId' parameter in the container 'SpiSequence' having short name <short name> should start with <0> and should be sequential without any gaps.
This error occurs, if the parameter 'SpiSequenceId' is not sequential or has gaps or starts with zero in configuration set in the ECU Configuration Description File.

ERR083008: In general per configuration set, the value of 'SpiChannelIndex' parameter in the container 'SpiChannelList' having short name <short name> should start with <0> and should be sequential without any gaps.
This error occurs, if the value of the parameter 'SpiChannelIndex' in the container 'SpiChannelList' does not start with 0 and is not sequential or has any gaps for any SpiJobId parameter in the container SpiJob.

ERR083009: The value of the parameter 'SpiChannelType' in the container 'SpiChannel' having short name <short name> should be configured as <EB>, since the value of the parameter 'SpiChannelBuffersAllowed' in the container 'SpiGeneral' having short name <short name> is configured as <1>.
This error occurs, if the parameter 'SpiChannelBuffersAllowed' in the container 'SpiGeneral' is configured as 1 and, the parameter 'SpiChannelType' in the container 'SpiChannel' is not configured as EB.

ERR083010: The value of the parameter 'SpiChannelType' in the container 'SpiChannel' having short name <short name> should be configured as <IB>, since the value of the parameter 'SpiChannelBuffersAllowed' in the container 'SpiGeneral' is configured as <0>.
This error occurs, If the parameter 'SpiChannelBuffersAllowed' in the container 'SpiGeneral' is configured as 0, and the parameter 'SpiChannelType' in the container 'SpiChannel' is not configured as IB.

ERR083013: The value of the parameter 'SpiMaxChannel' in container 'SpiDriver' having short name <short name> should be equal to the total number of 'SpiChannel' container configured within each 'SpiDriver' container.
This error occurs, if the value of the parameter 'SpiMaxChannel' present in the container 'SpiDriver' is not equal to the total number of channels configured in each SpiDriver container in the ECU Configuration Description File.

ERR083014: The value of the parameter 'SpiMaxJob' in container 'SpiDriver' having short name <short name> should be equal to the total number of 'SpiJob' container configured within each 'SpiDriver' container.
This error occurs, if the value of the parameter 'SpiMaxJob' in the container 'SpiDriver' is not equal to the total number of jobs configured in each SpiDriver container in the ECU Configuration Description File.

ERR083015: The value of the parameter 'SpiMaxSequence' in container 'SpiDriver' having short name <short name> should be equal to the total number of 'SpiSequence' container configured within each 'SpiDriver' container.

This error occurs, if the value of the parameter ‘SpiMaxSequence’ in the container ‘SpiDriver’ is not equal to the total number of sequences configured in each SpiDriver container in the ECU Configuration Description File.

ERR083018: The SPI channel having channel id <value of the parameter SpiChannelId> configured in the container ‘SpiChannel’ having short name <short name> should be referred by any of the SPI job.
This error will occur, if the configured SPI channel is not referenced by any of the SPI job.

ERR083030: The parameter 'SpiInterruptibleSequence' in the container 'SpiSequence' having short name <short name> should be configured as <false>, since the parameter 'SpiLevelDelivered' in container ‘SpiGeneral’ having short name <short name> is configured as <1> or <2> and the parameter 'SpiInterruptibleSeqAllowed' in the container 'SpiGeneral' having short name <short name> is configured as <false>.
This error will occur, if the value of the parameter 'SpiInterruptibleSequence' in the container 'SpiSequence' is configured as true and the parameter 'SpiLevelDelivered' as 1 or 2, and the parameter 'SpiInterruptibleSeqAllowed' in the container 'SpiGeneral' as false.

ERR083035: At least one instance of the container ‘SpiDma’ should be configured, since the parameter ‘SpiDmaMode’ in the container ‘SpiGeneral’ having short name <short name> is configured as <true>.
This error occurs, if the parameter ‘SpiDmaMode’ present in the container ‘SpiGeneral’ is configured as true, and no instance of the container ‘SpiDma’ is configured.

ERR083036: The value configured for the parameters ‘SpiTxDmaChannel/SpiRxDmaChannel/SpiDmaHwUnit’ in the container 'SpiDma' having short name <short name> should be unique within a configuration set.
This error occurs, if the same DMA channel is configured for SpiTxDmaChannel or SpiRxDmaChannel of SpiDma container for hardware units (SpiDmaHwUnit) in a configuration set. DMA channel (Tx or Rx) should be unique within a configuration set.
This error occurs, if the same SPI HW unit is configured for SpiDmaHwUnit within a configuration set. SpiDmaHwUnit should not be the same within a configuration set.

ERR083037: The value of the parameter ‘SpiDmaHwUnit’ in the container ‘SpiDma’ having short name <short name> should be configured in any of the hardware units selected for jobs.
This error occurs, if the value of the parameter ‘SpiDmaHwUnit‘ in the container ‘SpiDma’ is not configured in any of the hardware units selected for jobs.

ERR083043: The value configured for the parameter ‘SpiJobEndNotification’ in container ‘SpiJob’ having short name <short name> should follow C Syntax <[a-zA-Z][a-zA-Z0-9_]>.
This error occurs, if the value of configuration parameter ‘SpiJobEndNotification’ does not adhere to C syntax i.e., the value should not contain characters other than (a-z, A-Z, 0-9 or “_”) and it also should start with an alphabet.

ERR083046: The value of the parameter ‘SpiCsSelection’ in the container ‘SpiExternalDevice’ having short name 'short name' should be configured, since the value of the parameter ‘SpiEnableCs’ in the container ‘SpiExternalDevice’ having short name 'short name' is configured as <true>.
This error occurs, if the value of the parameter ‘SpiEnableCs’ in the container ‘SpiExternalDevice’ is configured as true, and the parameter ‘SpiCsSelection’ in the container ‘SpiExternalDevice’ is not configured.

ERR083047: 'The value of the parameter ‘SpiPortPinSelect’ in the container ‘SpiJob’ having short name 'short name' should be configured, since the value of the parameter ‘SpiEnableCs’ in the container ‘SpiExternalDevice’ having short name 'short name' is configured as <true>.
This error occurs, if the value of the parameter ‘SpiEnableCs’ in the container ‘SpiExternalDevice’ is configured as true, and the parameter ‘SpiPortPinSelect’ in the container ‘SpiJob’ is not configured.

ERR083048: 'The value of the parameter ‘SpiCsSelection’ in the container ‘SpiExternalDevice’ having short name 'short name' should not be configured, since the value of the parameter ‘SpiEnableCs’ in the container ‘SpiExternalDevice’ having short name 'short name' is configured as <false>.
This error occurs, if the value of the parameter ‘SpiEnableCs’ in the container ‘SpiExternalDevice’ is configured as false, and the parameter ‘SpiCsSelection’ in the container ‘SpiExternalDevice’ is configured.

ERR083049: 'The value of the parameter ‘SpiPortPinSelect’ in the container ‘SpiJob’ having short name 'short name' should not be configured, since the value of the parameter ‘SpiEnableCs’ in the container ‘SpiExternalDevice’ having short name 'short name' is configured as <false>.
This error occurs, if the value of the parameter ‘SpiEnableCs’ in the container ‘SpiExternalDevice’ is configured as false, and the parameter ‘SpiPortPinSelect’ in the container ‘SpiJob’ is configured.

ERR083092: The value configured for the parameter ‘SpiSeqEndNotification/SpiSeqStartNotification’ in container ‘SpiSequence’ having short name <short name> should follow C Syntax <[a-zA-Z][a-zA-Z0-9_]>.
This error occurs, if the value of configuration parameters mentioned below does not adhere to C syntax i.e., the value should not contain characters other than (a-z, A-Z, 0-9 or “_”) and it also should start with an alphabet.

Table 12.6 Parameter List for ERR083092

Parameter Name	Container Name
SpiSeqStartNotification	SpiSequence
SpiSeqEndNotification	SpiSequence

ERR083105: The parameter 'SpiInterruptibleSequence' in the container 'SpiSequence' having short name <short name> should be configured as <false>, since the value of the parameter ‘SpiMasterMode’ in the container ‘SpiExternalDevice’ having short name <short name> connected to this sequence is configured as <false>.
This error occurs, if the value of the parameter 'SpiInterruptibleSequence’ in the container ‘SpiSequence' is configured as <true>, and that of the parameter ‘SpiMasterMode’ in the container ‘SpiExternalDevice’ as <false>.

ERR083107: The value of the parameter ‘SpiMasterMode’ in the container ‘SpiExternalDevice’ having short name <short name> should be same within a sequence <value of SpiSequence shortname>.
This error occurs, if the value of the parameter ‘SpiMasterMode’ in the container ‘SpiExternalDevice’ is not the same in a sequence.

ERR083108: The value of the parameter ‘SpiHwUnit’ in the container ‘SpiExternalDevice’ having short name <short name> should be same within a sequence <value of the SpiSequence shortname>, since the parameter ‘SpiMasterMode’ in the container ‘SpiExternalDevice’ having short name <short name> is configured as <false>.
This error occurs, if the value of the parameter ‘SpiHwUnit’ in the container ‘SpiExternalDevice’ is not the same, and that of the parameter ‘SpiMasterMode’ in the container ‘SpiExternalDevice’ is configured as <false> in a sequence.

ERR083109: The parameter ‘SpiMasterMode’ in the container ‘SpiExternalDevicex’ having short name <short name> and ‘SpiExternalDevicey’ having short name <short name> should be same since they are same hardware unit <Value of parameter SpiHwUnit>.
This error occurs, if parameter ‘SpiMasterMode’ in the container ‘SpiExternalDevicex’ and ‘SpiExternalDevicey’ are not the same and they are the same hardware.

ERR083162: The value of parameter 'SpiClockFrequencyRef' in the container 'SpiDriver' having short name 'short name' should be configured since the value of parameter 'SpiEnableClkImmediateValue' is configured as <false> in the container <SpiGeneral> having short name <short name>.
This error occurs, if the value of 'SpiClockFrequencyRef' is not configured, and 'SpiEnableClkImmediateValue' is configured as false.

ERR083163: The value of clock reference parameter configured in 'SpiClockFrequencyRef' parameter in the container 'SpiDriver' having short name <short name> is not consistent with the value of 'SpiClockFrequencyRefImmediateValue' while the value of parameter 'SpiEnableClkImmediateValue' is configured as <true> in the container 'SpiGeneral' having short name <short name>.
This error occurs, if the 'SpiEnableClkImmediateValue' is enabled and the reference parameter is configured but the value is not consistent.

ERR083179: The value of parameter 'SpiClockFrequencyRefImmediateValue' in the container 'SpiDriver' should be configured since the value of parameter 'SpiEnableClkImmediateValue' is configured as <true> in the container 'SpiGeneral'.
This error occurs, if the value of "SpiClockFrequencyRefImmediateValue" is not configured, and 'SpiEnableClkImmediateValue' is configured as true.

12.7.2 Common Warning Messages

WRN083001: The SpiJob having 'SpiJobId' as <value of the parameter 'SpiJobId'> in the container 'SpiJob' having short name 'short name' is not referred by any of the SPI sequence.
This warning occurs, if the configured SPI job is not referenced by any of the SPI sequence in the ECU Configuration Description File.

WRN083002: In the SpiChannel <value of shortname SpiChannel>, the value of parameter 'SpiDefaultData' is greater than (2 exponent of <value of parameter SpiDataWidth> - 1). Therefore, only lower part is used and the upper part is ignored from the value of parameter 'SpiDefaultData'.
This warning occurs, if the value configured in the parameter 'SpiDefaultData' of the container 'SpiChannel' is greater than the value (2^ SpiDataWidth – 1) of the same container.

WRN083003: The parameter 'SpiLevelDelivered' is configured as <0> and 'SpiInterruptibleSeqAllowed' in the container 'SpiGeneral' having short name 'short name' is configured as <true>. Hence the value of parameter 'SpiInterruptibleSeqAllowed' is ignored.
This information occurs, if the parameter 'SpiLevelDelivered' is configured as 0 (SPI Level 0 Driver), and 'SpiInterruptibleSeqAllowed' in the container 'SpiGeneral' as true. Hence the value of the parameter 'SpiInterruptibleSeqAllowed' is ignored.

WRN083004: The parameter 'SpiSeqStartNotification' in the container 'SpiSequence' having short name 'short name' will be ignored, since the pre-compile parameter 'SpiEnableSeqStartNotification' in the 'SpiGeneral' container having short name <short name> is configured as <false>.
The warning occurs if the parameter 'SpiSeqStartNotification' in the container 'SpiSequence' is configured, and the parameter 'SpiEnableSeqStartNotification' in the 'SpiGeneral' container is configured as false.

WRN083006: The value of parameter 'SpiDmaMode' from the container 'SpiGeneral' is configured as <false> and the container 'SpiDma' is configured. In this case, the configuration in 'SpiDma' container is ignored.
This warning occurs, if the value of the parameter 'SpiDmaMode' in the container 'SpiGeneral' is configured as false, and the container 'SpiDma' is configured. In this case, the configuration in SpiDma container is ignored.

WRN083007: The SpiExternalDevice container having short name <short name> configures for SpiHwUnit <value of SpiHwUnit> but it is not associated with any job.
This warning occurs, if the SpiExternalDevice is configured but not associated with any job.

WRN083011: The value of reference clock configured in the 'SpiClockFrequencyRef' parameter in the container 'SpiDriver' having short name <short name> is invalid since the 'SpiEnableClkImmediateValue' parameter in the container 'SpiGeneral' having short name <short name> is configured as <true>.
This warning occurs, if the value of the reference clock configured in the 'SpiClockFrequencyRef' parameter in the container 'SpiDriver' is invalid since the 'SpiEnableClkImmediateValue' parameter in the container 'SpiGeneral' is configured as < true >.

12.7.3 Common Information Messages

INF083004: The value of 'SpiClockFrequencyRefImmediateValue' parameter in the container 'SpiDriver' having short name 'short name' will not be used since the 'SpiEnableClkImmediateValue' parameter in the container 'SpiGeneral' having short name 'short name' is configured as < false >.
The information occurs, if the 'SpiEnableClkImmediateValue' parameter in the container 'SpiGeneral' is configured as < false >. Then the value of 'SpiClockFrequencyRefImmediateValue' parameter in the container 'SpiDriver' will not be used.

INF083005: The value of 'SpiClockFrequencyRefImmediateValue' parameter in the container 'SpiDriver' having short name 'short name' will not be used since the 'SpiEnableClkImmediateValue' parameter in the container 'SpiGeneral' having short name 'short name' is configured as < true >.
The information occurs, if the 'SpiEnableClkImmediateValue' parameter in the container 'SpiGeneral' is configured as < true >, and the 'SpiClockFrequencyRef' is not configured. Then the 'SpiEnableClkImmediateValue' parameter will be used without checking.

12.7.4 Common Deviation List

None.

13.WDG

13.1 Overview

The Watchdog Driver provides services for initialization, changing the operation mode and triggering the Watchdog.

The WDG Driver module comprises two sections, Embedded Software and the Generation Tool, to achieve scalability and configurability.

The document describes the features of the WDG Driver Generation Tool. This tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File and generates WDG Driver C Source and C Header files (Wdg_Cfg.h and Wdg_PBcfg.c).

This document contains information on the options, input and output files of the WDG Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool. ECU Configuration Description File contains information about WDG configuration.

This program is built based on Microsoft .NET Framework.
Refer to the following sites about license in detail.
https://msdn.microsoft.com/en-us/library/ms994405.aspx

13.2 WDG Driver Generation Tool Overview

WDG Driver Generation Tool overview is shown in Figure 13-1.

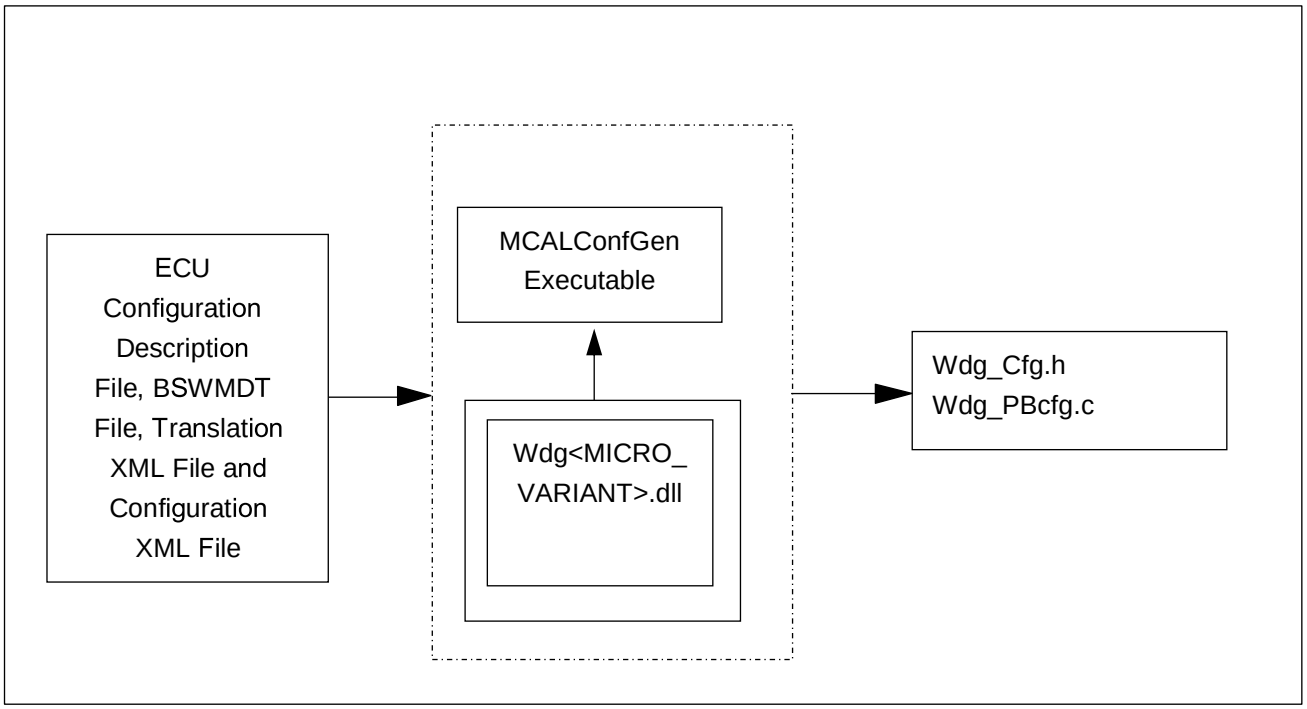


Figure 13-1 Overview of WDG Driver Generation Tool

WDG Driver Generation Tool is a command line tool that extracts and analyzes the configuration details provided in the input file, validates correctness of the data and provides scalability and configurability for WDG Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input and displays appropriate context-sensitive error messages for wrong input and exits. Tool creates the Log file (Wdg.log) that contains the list of Error/Warning/Information messages in the output directory.

WDG Driver Generation Tool comprises two components, MCALConfGen executable and Wdg<MICRO_VARIANT>.dll. At runtime, the executable loads the dll to generate output files.

For the error-free input file, the tool generates the following output files: Wdg_Cfg.h and Wdg_PBcfg.c Wdg_Cfg.h will be compiled and linked with WDG Driver Component. Wdg_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash.

Note:

- The generation tool returns 1 when error, and 0 when no errors.
- WDG Driver Generation Tool uses “Common Published Information” from WDG module specific BSWMDT File. This file should not be updated manually since it is “Static Configuration” file.

13.3 Input Files

WDG Driver Generation Tool accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input. WDG Driver Generation Tool needs information about WDG Driver module. Hence ECU Configuration Description File should contain configuration of WDG Driver module. Generation 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.

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

13.4 Output Files

WDG Driver Generation Tool generates configuration details in C Header and C Source files (Wdg_Cfg.h, Wdg_PBcfg.c).

The content of each output file is given in the Table 13-1.

Table 13-1 Output Files Description

Output File	Details
Wdg_Cfg.h	This file contains pre-compile time parameters. It also contains the macro definitions of the development error detection version info API, compile switch to allow/ forbid disabling the Watchdog Driver during runtime and Watchdog Driver Id. This file also contains information for maximum Watchdog Timer timeout, Minimum Watchdog Timer timeout, configuration set handles, resolution of Watchdog time out period and Watchdog trigger mode.
Wdg_PBcfg.c	This file contains post-build configuration data.

Note: Output files generated by WDG Driver Generation Tool should not be modified or edited manually.

13.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain WDG Driver, GPT driver, MCU Driver and DEM component related configuration.
- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Wdg.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file Wdg.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension '.trxml'.
- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by WDG Driver Generation 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.
- C Source and C Header files will be generated by the Watchdog Driver Generation Tool based on the configuration of the parameters 'VENDOR-ID' and 'VENDOR-API-INFIX' in the WDG Driver-specific BSWMDT File.
- An error-free ECU Configuration Description File generated from configuration editor must be provided as input to the WDG Driver Generation Tool. Otherwise, the tool may not produce the expected results or may lead to errors/warnings/information.

Note: For more information, please refer to the following appendix file: "V4M_Deviation_List.xlsx" in V4M AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual.

13.6 Specific 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 device-specific error, warning and information messages displayed by the Generation Tool.

13.6.1 Specific Error Messages

ERR102004: The parameter 'parameter name' in the container 'container name' should be configured.
This error will occur, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File.
The list of mandatory parameters with respect to the container is listed in Table 13-2.

Table 13-2 Mandatory Configuration Parameters List

Container	Parameters
WdgGeneral	WdgDevErrorDetect
	WdgDisableAllowed
	WdgIndex
	WdgVersionInfoApi
	WdgInitialTimeout
	WdgRunArea
	WdgTriggerLocation
	WdgMaxTimeout
	WdgDeviceName
	WdgTimeMargin
	WdgCriticalSectionProtection
	WdgVersionCheckExternalModules
	WdgClockRef
WdgPublishedInformation	WdgTriggerMode
WdgSettingsFast	WdgClkSettingsFast
	WdgFastTriggerTimeout
WdgSettingsSlow	WdgClkSettingsSlow
	WdgSlowTriggerTimeout
WdgSettingsConfig	WdgDefaultMode
WdgGptConfiguration	WdgGptContainerRef

ERR102037: The value configured for parameter 'GptChannelMode' in container'GptChannelConfiguration' should be <GPT_CH_MODE_CONTINUOUS> since this channel is used to call Wdg callback function.
This error will occur, if the parameter 'GptChannelMode' in 'GptChannelConfiguration' container is configured as "One shot" mode.

ERR102038: The value configured for parameter 'GptNotification' in container 'GptChannelConfiguration' should be configured and not be <NULL>since this channel is used to call Wdg callback function.

This error will occur, if the parameter 'GptNotification' in 'GptChannelConfiguration' container is not configured or configured as NULL.

ERR102039: The value configured for parameter 'GptChannelClkPrescaler' in container 'GptChannelConfiguration' shouldn't be <EXTERNAL_CLK> since this channel is used to call Wdg callback function.

This error will occur, if the parameter 'GptChannelClkPrescaler' in 'GptChannelConfiguration' container is configured as EXTERNAL_CLK.

ERR102040: The reference path <path> provided for the parameter 'GptTMUClkSrcRef' within the container 'GptChannelConfiguration' is incorrect.

This error will occur, if path provided for the 'GptTMUClkSrcRef' parameters is incorrect.

ERR102041: The reference path <path> provided for the parameter 'WdgGptContainerRef' within the container 'WdgGptConfiguration' is incorrect.

This error will occur, if path provided for the 'WdgGptContainerRef' parameters is incorrect.

ERR102042: The value for the DEM parameter 'WDG_E_DISABLE_REJECTED' in the container 'WdgDemEventParameterRefs' should be configured since the value of the parameter 'WdgDisableAllowed' in 'WdgGeneral' container is configured as <DISABLED>.

This error will occur, if WdgDisableAllowed is disabled but the DEM parameter 'WDG_E_DISABLE_REJECTED' is not configured.

ERR102044: The summation of GPT timeout that is represented by parameter 'WdgFastTriggerTimeout' or 'WdgSlowTriggerTimeout' and Margin timeout that is calculated by multiplication of GPT timeout and 'WdgTimeMargin'/100 is out of range of WDG counter value.

This error occurs when the summation of GPT timeout (represented by parameter 'WdgFastTriggerTimeout' or 'WdgSlowTriggerTimeout') and Margin timeout (calculated by multiplication of GPT timeout and 'WdgTimeMargin'/100) exceeds the maximum value of WDG counter value (65535).

ERR102050: The reference path <path> provided for the parameter 'Parameter Name' within the container 'Container Name' is incorrect.

This error will occur, if the path provided for the parameter 'WdgClockRef' in the container 'WdgGeneral' is incorrect.

ERR102051: The reference path <path> provided for the DEM parameter 'Parameter Name' within the container 'Container Name' is incorrect.

This error will occur, if the path provided for the DEM parameter in below table is incorrect.

Table 13-3 Reference Parameters List

Container Name	Parameter Name
WdgDemEventParameterRefs	WDG_E_DISABLE_REJECTED
	WDG_E_MODE_FAILED
	WDG_E_WRITE_REGISTER_FAILED
	WDG_E_VALUE_COUNTER_FAILED

ERR102052: The reference path <path> configured in the container 'Container Name' is duplicated.
This error will occur, if the path configured in DEM container in *Table 13-3 Reference Parameters List* is duplicated.

13.6.2 Specific Warning Messages

WRN102004: The value for DEM parameter 'WDG_E_DISABLE_REJECTED' in the container 'WdgDemEventParameterRefs' should not be configured since the value of the parameter 'WdgDisableAllowed' in 'WdgGeneral' container is configured as <ENABLED>.
This warning will occur, if the value configured for the parameter WdgDisableAllowed in the container WdgGeneral is configured as ENABLED and value of DEM reference parameter WDG_E_DISABLED_REJECTED in the container WdgDemEventParameterRefs is still configured.

13.6.3 Specific Information Messages

INF102006: The timeout duration of one WDG trigger cycle for slow mode is <'slow_time' msec>.
This information notify the slow mode timeout.

INF102007: The timeout duration of one WDG trigger cycle for fast mode is <'fast_time' msec>.
This information notify the fast mode timeout.

13.6.4 Specific Deviation List

None.

13.7 Common 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 common error, warning and information messages displayed by the Generation Tool.

13.7.1 Common Error Messages

ERR102003: WDG Driver or MCU Driver or GPT Driver or DEM Component is not present in the input file(s).
For WDG Driver:
This error will occur, if the WDG Driver Component is not present in the input ECU Configuration Description File(s).
For MCU Driver:
This error will occur, if MCU Driver Component is not present in the input ECU Configuration Description File(s).
For GPT Driver:
This error will occur, if GPT Driver Component is not present in the input ECU Configuration Description File(s).
For DEM:
This error will occur, if Dem Component is not present in the input ECU Configuration Description File(s) when parameter(s) in WdgDemEventParameterRefs container are configured.

ERR102006: The value configured for the parameter 'WdgClkSettingsFast' in the container 'WdgSettingsFast' having ShortName '<Container ShortName>' and value of the parameter 'WdgClkSettingsSlow' in the container 'WdgSettingsSlow' having ShortName '<Container ShortName>' are same.
This error will occur, if the value configured for the parameter 'WdgClkSettingsFast' in the container 'WdgSettingsFast' and the value of the parameter 'WdgClkSettingsSlow' in the container 'WdgSettingsSlow' are the same.

ERR102007: The value configured for the parameter 'WdgClkSettingsSlow' in the container 'WdgSettingsSlow' having ShortName '<Container ShortName>' is faster (valid configuration should be WdgClkSettingsSlow > WdgSettingsFast) than the value of the parameter 'WdgClkSettingsFast' in the container 'WdgSettingsFast' having ShortName '<Container ShortName>'.
This error will occur, if the value configured for the parameter 'WdgClkSettingsSlow' in the container 'WdgSettingsSlow' is faster than that of 'WdgClkSettingsFast' in the container 'WdgSettingsFast'.

ERR102008: The value of the parameter 'WdgInitialTimeout' is greater than the value of the parameter 'WdgMaxTimeout'.
This error will occur, if the value of the parameter 'WdgInitialTimeout' is greater than that of the parameter 'WdgMaxTimeout'.

ERR102009: The value of the parameter 'WdgDisableAllowed' should not be <false> since the value of the parameter 'WdgDefaultMode' is configured as <WDGIF_OFF_MODE>.

This error will occur, if the value of the parameter ‘WdgDisableAllowed’ is false, and that of the parameter ‘WdgDefaultMode’ is configured as WDGIF_OFF_MODE.

13.7.2 Common Warning Messages

WRN102001: The value of the parameter 'WdgInitialTimeout' is configured as <0> and 'WdgDefaultMode' is not configured as <WDGIF_OFF_MODE>. Hence, Watchdog hardware will be enabled directly after Wdg Module initialization and WDG counter will expire after <calculated_value in msec>.

This warning will occur, if WdgInitialTimeout parameter is configured as 0 and WdgDefaultMode is not configured as WDGIF_OFF_MODE.

13.7.3 Common Information Messages

None.

13.7.4 Common Deviation List

None.

“Generation Tool” and “Tool” terminologies are used interchangeably to refer to the WDG Driver Generation Tool.

14.THS

14.1 Overview

The THS Complex Driver module provides the service for initializing the whole CDD THS structure of the microcontroller.

The THS Complex Driver module comprises of two sections as Embedded Software and the Generation Tool to achieve scalability and configurability.

The document describes the features of the THS Complex Driver Generation Tool. THS Complex Driver Generation Tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File and generates THS Complex Driver C Source and C Header files (CDD_Ths_PBcfg.c, CDD_Ths_Cfg.h and CDD_Ths_Reg.h).

This document contains information on the options, input and output files of the THS Complex Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

ECU Configuration Description File contains information about THS CDD Configuration.

14.2 THS Complex Driver Generation Tool Overview

THS Complex Driver Generation Tool overview is shown below.

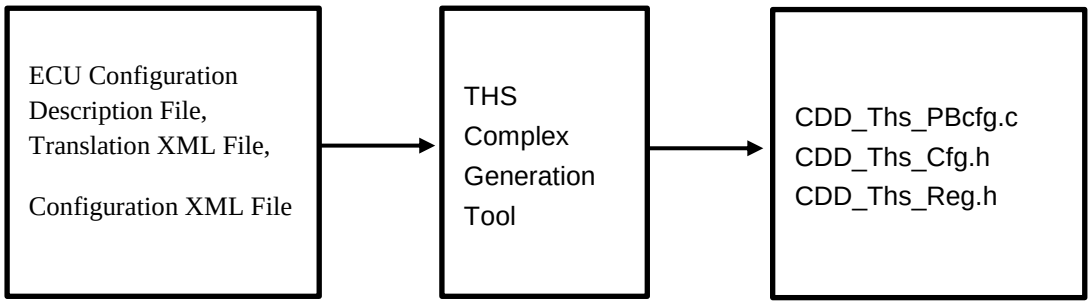


Figure 14-1 Overview of THS Complex Driver Generation Tool

THS Complex Driver Generation Tool is a command line tool that extracts, analyzes the configuration details provided in the input file and validates correctness of the data and provides scalability and configurability for THS Complex Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input and displays appropriate context sensitive error messages for wrong input and exits. Tool creates the Log file (CddThs.log) that contains the list of Error/Warning/Information messages in the output directory.

THS Complex Driver Generation Tool comprises 2 components, MCALConfGen executable, CddThsRCAR.dll. At runtime, the executable component loads the dll to generate output files.

CDD_Ths_Cfg.h and CDD_Ths_Reg.h will be compiled and linked with THS Complex Driver module. CDD_Ths_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash.

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

For the error free input file, the tool generates the following output files: C header CDD_Ths_Cfg.h, CDD_Ths_Reg.h and C source CDD_Ths_PBcfg.c.

Remark:
In case of errors the generation tool returns a 1, in case of no errors the generation tool returns a 0.

THS Complex Driver Generation Tool uses “Common Published Information” from CDD THS module specific BSWMDT File. CDD THS module specific BSWMDT File should not be updated manually since it is "Static Configuration” file.

14.3 Input Files

THS Complex Driver Generation Tool accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input. THS Complex Driver Generation Tool needs information about THS Complex Driver module. Hence ECU Configuration Description File should contain configuration of THS Complex Driver module. THS Complex Driver Generation 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.

Note: The detailed explanations about the parameters and containers are found in Parameter Definition File.

14.4 Output Files

THS Complex Driver Generation Tool generates configuration details in C Header and C Source files (CDD_Ths_PBcfg.c, CDD_Ths_Cfg.h, CDD_Ths_Reg.h).

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

Table 14-1 Output Files Description

Output File	Details
CDD_Ths_Cfg.h	This file contains pre-compile time parameters and handles.
CDD_Ths_Reg.h	This file contains defining macro for CDD THS register address.
CDD_Ths_PBcfg.c	This file contains structure for CDD THS channels initialization, thermal channel configuration during runtime.

Note: Output files generated by THS Complex Driver Generation Tool should not be modified or edited manually.

14.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R1911 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain THS Complex Driver module, Dem, Os component related configuration files.
- The configuration <FILTER-RENESAS> should be configured as ON in Configuration XML File (E.g. CddThs_RCar.cfgxml) if same module of different vendor is used with this module of Renesas.
- If <FILTER-RENESAS> is ON, the configuration <FILTER-NAME> will contain the name of vendors which will be processed by THS Complex Driver Generation Tool.
  Example : CDF file contains configuration of Renesas, AUTOSAR and VendorX. Renesas THS Complex Driver Generation Tool will only process the configuration of Renesas and AUTOSAR if <FILTER-RENESAS> and <FILTER-NAME> are configured as below :
  <FILTER-RENESAS>ON</FILTER-RENESAS>
  <FILTER-NAME>Renesas</FILTER-NAME>
  <FILTER-NAME>AUTOSAR</FILTER-NAME>
- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddThs.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddThs.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension '.trxml'.
- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by THS Complex Driver Generation 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 must be provided as input to the THS Complex Driver Generation Tool. Otherwise, Tool may not produce the expected results or may lead to errors/warnings/information.

Note: Please refer the CDD THS Component User Manual for deviations from AUTOSAR specifications, if any.

14.6 Specific 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 errors, warning and information messages displayed by the THS Complex Driver Generation Tool.

For the list of common error/warning/information messages for all modules, refer to "R-Car Gen4 AUTOSAR R19-11 MCAL User's Manual Getting Started".

14.6.1 Specific Error Messages

ERR255006: The parameter 'CDD_THS_E_WRITEVERIFY_FAILURE' in the container 'CddThsDemEventParameterRefs' must be configured with a valid reference path when the parameter 'CddThsWriteVerifyCheck' in the container 'CddGeneral' is enabled.
This error will occur, if the value for parameter CDD_THS_E_WRITEVERIFY_FAILURE present in container CddThsDemEventParameterRefs wasn't configured while CddThsWriteVerifyCheck is enabling.

ERR255011: The characteristics parameters <parameter 1>, <parameter 2>, <parameter 3> in the container 'CddGeneral' must not be configured the same value. This condition is to avoid division by zero in source code.
This error will occur, if one of three parameters in below group has configured with the same value as other(s).

Table 14-2 Parameter of ERR255011

Group	Parameter	Device
Group 1	CddThsPtat1 CddThsPtat2 CddThsPtat3	V4M
Group 2	CddThsThcode1 CddThsThcode2 CddThsThcode3	V4M

ERR255017: 'DEM' component is not present in the input file(s) when at least one parameter inside 'CddThsDemEventParameterRefs' is configured.

This error will occur, if DEM component is not present in the input ECU Configuration Description File(s) when at least one parameter inside CddThsDemEventParameterRefs is configured.

Table 14-3 Parameter CDD_THS_E_WRITEVERIFY_FAILURE of ERR255017

Container	Parameter	Device
CddThsDemEventParameterRefs	CDD_THS_E_WRITEVERIFY_FAILURE	V4M

14.6.2 Specific Warning Messages

WRN255001: The THS module is off and unable to change its operation since initial operation state is CDD_THS_IDLE and API to change the module operation state is disabling.
This warning will occur, if the value for parameter CddThsInitialOperationState is CDD_THS_IDLE while the parameter CddThsAllowConfigureOperationState in container CddGeneral is configured as <false>.

WRN255002: The interruption features are disable for all channels, since the parameter 'CddThsEnableThermalInterruption' in container 'CddGeneral' is un-checked.
This warning will occur, if the parameter CddThsThermalChannelEnable present in container CddThsThermalChannel_x (x = 0…2) is checked while the parameter CddThsEnableThermalInterruption in container CddGeneral is un-checked.

WRN255006: The configured value for the parameter <DEM parameter> in the container 'CddThsDemEventParameterRefs' should not be configured since the value of the parameter <parameter> of 'CddGeneral' container is configured as <DISABLED>.
This warning will occur, if the value of parameter <DEM parameter> is configured while <parameter> is configured as <DISABLED>.

Table 14-4 Parameter CDD_THS_E_WRITEVERIFY_FAILURE of ERR255006

Parameter	DEM Parameter	Device
CddThsWriteVerifyCheck	CDD_THS_E_WRITEVERIFY_FAILURE	V4M

14.6.3 Specific Information Messages

None.

14.6.4 Specific Deviation List

None.

14.7 Common 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 errors, warning and information messages displayed by the THS Complex Driver Generation Tool.

For the list of common error/warning/information messages for all modules, refer to "R-Car Gen4 AUTOSAR R19-11 MCAL User's Manual Getting Started".

14.7.1 Common Error Messages

ERR255001: Number of fields is not same for the entity 'Structure Name'.
This error will occur, if the number of fields is not same in the structure that is to be generated in the output file.

ERR255002: Field 'Field Name' is empty in the entity 'Structure Name'.
This error will occur, if the structure fields that are to be generated in the output file are empty.

ERR255003: <Module> Driver Component is not present in the input file(s).
This error will occur, if <Module> Driver component mentioned below is (are) not present in the input ECU Configuration Description File(s).

Table 14-5 List of modules require in ECU configuration

Module
CDD THS Driver

ERR255004: The parameter 'parameter name' in the container 'container name' must be configured.
This error will occur, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File.

The list of mandatory parameters with respect to container is listed below:

Table 14-6 List of parameters require in ECU configuration

Container	Parameters	Device
CddGeneral	CddInstanceId	V4M
	CddThsVersionInfoApi	V4M
	CddThsVoltageInfo	V4M
	CddThsTemperatureInfo	V4M
	CddThsAllowConfigureOperationState	V4M

	CddThsDevErrorDetection	V4M
	CddThsEnableThermalInterruption	V4M
	CddThsVersionCheckExternalModules	V4M
	CddThsCriticalSectionProtection	V4M
	CddThsWriteVerifyCheck	V4M
	CddThsHardWareUnitOption	V4M
	CddThsPtat1	V4M
	CddThsPtat2	V4M
	CddThsPtat3	V4M
	CddThsThcode1	V4M
	CddThsThcode2	V4M
	CddThsThcode3	V4M
	CddThsInitialOperationState	V4M
CddThsThermalChannel_0	CddThsThermalChannelId	V4M
	CddThsThermalChannelEnable	V4M
	CddThsThermalInterruptionType	V4M
	CddThsThermalInterruptionValue	V4M
CddThsThermalChannel_1	CddThsThermalChannelId	V4M
	CddThsThermalChannelEnable	V4M
	CddThsThermalInterruptionType	V4M
	CddThsThermalInterruptionValue	V4M
CddThsThermalChannel_2	CddThsThermalChannelId	V4M
	CddThsThermalChannelEnable	V4M
	CddThsThermalInterruptionType	V4M
	CddThsThermalInterruptionValue	V4M

ERR255800: The container <container> must be configured.

This error will occur if the following containers are not configured.

Table 14-7 List of containers must be configured

Container	Device
CddGeneral	V4M
CddThsThermalInterruption	V4M
CddThsThermalChannel_0	V4M
CddThsThermalChannel_1	V4M
CddThsThermalChannel_2	V4M

ERR255801: The number of containers <container> must be in valid multiplicity range from <min value> to <max value>.

This error will occur if the number of following containers is configured out of its range.

Table 14-8 List of containers and the range of its configuration

Container	Range		Device
	Min	Max	
CddGeneral	1	1	V4M
CddThsThermalInterruption	1	1	V4M
CddThsThermalChannel_0	1	1	V4M
CddThsThermalChannel_1	1	1	V4M
CddThsThermalChannel_2	1	1	V4M
CddThsDemEventParameterRefs	0	1	V4M

ERR255802: The value of parameter 'parameter name' in Container 'container name' must be in its valid range from <min value of range> to <max value of range>.

This error will occur if the value of below parameter is configured out of its range.

Table 14-9 List of parameters and the range of its configuration

Container	Parameters	Min value	Max value	Remark
CddGeneral	CddThsPtat1	0	4095	V4M
	CddThsPtat2	0	4095	V4M
	CddThsPtat3	0	4095	V4M
	CddThsThcode1	0	4095	V4M
	CddThsThcode2	0	4095	V4M
	CddThsThcode3	0	4095	V4M
CddThsThermalChannel_0	CddThsThermalInterruptionValue	-40	125	V4M
CddThsThermalChannel_1	CddThsThermalInterruptionValue	-40	125	V4M
CddThsThermalChannel_2	CddThsThermalInterruptionValue	-40	125	V4M

ERR255803: Value of parameter <parameter> in container <container> must be a valid enumeration literal value.

This error will occur, if the number of following parameters in correspond containers are not configured by a invalid enumeration literal value.

Table 14-10 List of parameters and the value of its configuration

Container	Parameter	Value	Device
CddGeneral	CddThsInitialOperationState	CDD_THS_NORMAL	V4M
		CDD_THS_IDLE	V4M
	CddThsHardWareUnitOption	TSC1	V4M
		TSC2	V4M
	CddThsThermalChannelId	CDD_THS_THERMAL_CH0	V4M

CddThsThermalChannel_0	CddThsThermalInterruptionType	CDD_THS_LOWER_BOUND	V4M
		CDD_THS_UPPER_BOUND	V4M
CddThsThermalChannel_1	CddThsThermalChannelId	CDD_THS_THERMAL_CH1	V4M
	CddThsThermalInterruptionType	CDD_THS_LOWER_BOUND	V4M
		CDD_THS_UPPER_BOUND	V4M
CddThsThermalChannel_2	CddThsThermalChannelId	CDD_THS_THERMAL_CH2	V4M
	CddThsThermalInterruptionType	CDD_THS_LOWER_BOUND	V4M
		CDD_THS_UPPER_BOUND	V4M

14.7.2 Common Warning Messages

None.

14.7.3 Common Information Messages

None.

14.7.4 Common Deviation List

None.

15.IPMMU

15.1 Overview

The IPMMU Complex Device Driver module provides the service for initializing the whole CDD IPMMU structure of the microcontroller.

The IPMMU Complex Device Driver module comprises of two sections as Embedded Software and the Generation Tool to achieve scalability and configurability.

The document describes the features of the IPMMU Complex Device Driver Generation Tool. IPMMU Complex Device Driver Generation Tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File and generates IPMMU Complex Device Driver C Source and C Header files (CDD_Ipmmu_Cfg.h, CDD_Ipmmu_PBcfg.c, CDD_Ipmmu_Cbk.h and CDD_Ipmmu_Hardware.h).

This document contains information on the options, input and output files of the IPMMU Complex Device Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.
ECU Configuration Description File contains information about IPMMU CDD Configuration.

15.2 IPMMU Driver Generation Tool Overview

IPMMU Complex Driver Generation Tool overview is shown below:

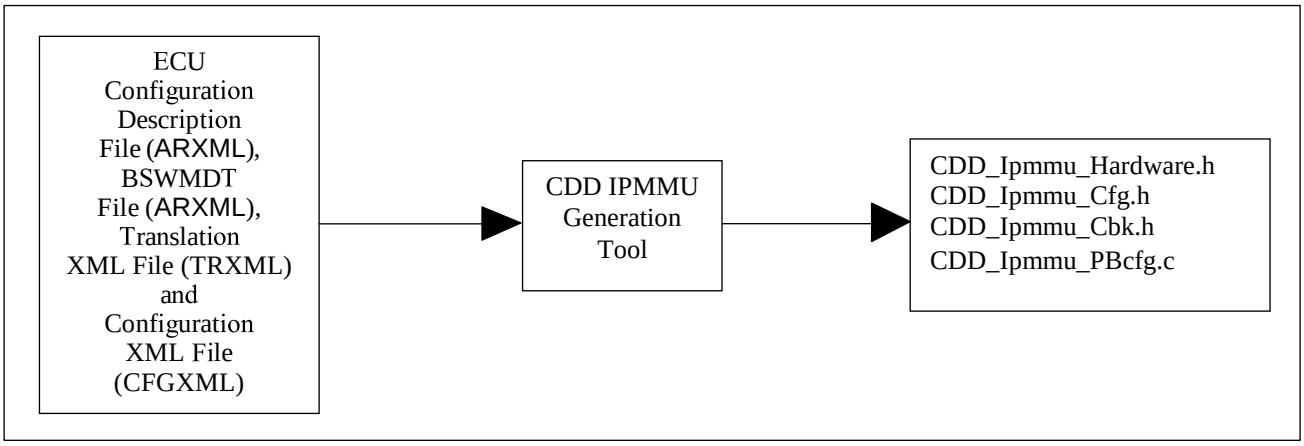


Figure 15-1 Overview of IPMMU Complex Device Driver Generation Tool

IPMMU Complex Driver Generation Tool is a command line tool that extracts, analyzes the configuration details provided in the input file, validates correctness of the data and provides scalability and configurability for IPMMU Complex Driver module. It accepVts ECU Configuration Description File(s), Translation XML File, BSWMDT File and Configuration XML File as input, displays appropriate context sensitive error messages for wrong input and exits. Tool creates the Log file (CDD_Ipmmu.log) that contains the list of Error/Warning/Information messages in the output directory.

For the error free input file, the tool generates the following output files CDD_Ipmmu_Cfg.h, CDD_Ipmmu_PBcfg.c, CDD_Ipmmu_Cbk.h and CDD_Ipmmu_Hardware.h.

CDD_Ipmmu_Cfg.h will be compiled and linked with IPMMU Complex Driver Component. CDD_Ipmmu_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash. CDD_Ipmmu_Cbk.h contain prototype of Callback Functions. CDD_Ipmmu_Hardware.h contain hardware information of IPMMU.
ECU Configuration Description File can be created or edited using ECU Configuration Editor.

Note:

- In case of errors the generation tool returns a1, in case of no errors the generation tool returns a0.
- IPMMU Complex Device Driver Generation Tool uses “Common Published Information” from CDD IPMMU module specific BSWMDT File. CDD IPMMU module specific BSWMDT File should not be updated manually since it is “Static Configuration” file.

15.3 Input Files

This chapter describes the inputs files accepted by the IPMMU Complex Driver Generation Tool. IPMMU Complex Driver Generation Tool needs information about IPMMU Complex Driver module.

Hence ECU Configuration Description File should contain configuration of IPMMU Complex Device Driver module. Generation 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.

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

15.4 Output Files

IPMMU Complex Driver Generation Tool generates configuration details in C Header and C Source files (CDD_Ipmmu_Cfg.h, CDD_Ipmmu_Cbk.h, CDD_Ipmmu_PBcfg.c and CDD_Ipmmu_Hardware.h).

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

Table 15-1 Output Files Description

Output File	Details
CDD_Ipmmu_Cfg.h	This file contains pre-compile time parameters.
CDD_Ipmmu_PBcfg.c	This file contains post-build time parameters.
CDD_Ipmmu_Cbk.h	This file contains prototype of callback functions.
CDD_Ipmmu_Hardware.h	This file contains hardware information of IPMMU.

Note: Output files generated by IPMMU Complex Device Driver Generation Tool should not be modified or edited manually.

15.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain IPMMU Complex Device Driver module.
- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddIpmmu.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddIpmmu.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension '.trxml'.
- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by IPMMU Complex Device Driver Generation 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 IPMMU Complex Device Driver Generation Tool. Otherwise, Tool may not produce the expected results or may lead to errors/warnings/information messages.
- User has to make sure that the respective device specific configuration file is used otherwise Tool may not produce the expected results or may lead to errors/warnings/information messages.
- The description file should always be generated using AUTOSAR specified configuration editor and it should not be edited manually.

Note: For more information, please refer to the following appendix file: "*ASR_MCAL_R-CarGen4_Deviation_List.xlsx*" in R-Car Gen4 AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual.

15.6 Specific 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 Generation.

15.6.1 Specific Error Messages

ERR255004: The parameter 'parameter name' in the container 'container name' should be configured.
This error occurs, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File. The list of mandatory parameters with respect to container is listed as table below:

Table 15-2 Parameter List for ERR255004

Container	Parameters
CddGeneral	CddInstanceId

Container	Parameters
	CddIpmmuDevErrorDetect
	CddIpmmuVersionInfoApi
	CddIpmmuAlreadyInitDetCheck
	CddIpmmuUnintendedInterruptCheck
	CddIpmmuWriteVerifyCheck
	CddIpmmuCriticalSectionProtection
	CddIpmmuVersionCheckExternalModules
	CddIpmmuEdcErrorDetect
	CddIpmmuC4PowerDomainSupport
CddIpmmuMmu	CddIpmmuMmuId
	CddIpmmuMmuSelection
	CddIpmmuMmuAArchSupport
	CddIpmmuMmuTranslationTableFormat
	CddIpmmuMmuStartLevel
	CddIpmmuMmuMemoryAttributeIndirection0
	CddIpmmuMmuMemoryAttributeIndirection1
	CddIpmmuMmuCallbackFunction
	CddIpmmuMmuAccessFlagEnable
CddIpmmuMmuTranslationTable	CddIpmmuMmuTranslationTableId
	CddIpmmuMmuUpperBaseAddress
	CddIpmmuMmuLowerBaseAddress
	CddIpmmuMmuTranslationSize
	CddIpmmuMmuShareAbility
	CddIpmmuMmuOuterCacheAbility
	CddIpmmuMmuInnerCacheAbility
CddIpmmuPmb	CddIpmmuPmbId
	CddIpmmuPmbSelection
	CddIpmmuPmbBusDomain

Container	Parameters
	CddIpmmuPmbVirtualPageNumber
	CddIpmmuPmbPhysicalPageNumber
	CddIpmmuPmbUpperPhysicalPageNumber
	CddIpmmuPmbPageSize
	CddIpmmuPmbCacheBit
	CddIpmmuPmbCallbackFunction
CddIpmmuTlb	CddIpmmuTlbId
	CddIpmmuTlbSelection
	CddIpmmuTlbBusDomain
	CddIpmmuTlbTranslationTable
	CddIpmmuTlbASID
	CddIpmmuTlbAddressTranslationEnable

ERR255007: The reference path <Dem_event_ref> provided for the parameter '<parameter_name>' within the container '<container_name>' is incorrect.

This error will occur, if the path provided for the following parameters in the respective container is incorrect.

The <parameter_name> can be: CDDIPMMU_E_WRITE_VERIFY_FAILURE.

ERR255009: The value of parameter '<parameter name>' in the corresponding container '<Container name>' should be configured in ascending order and continuously from zero.

This error will occur, if parameter <parameter_name> in container <container_name> are not configured in ascending order and continuously from zero.

Table 15-3 Parameter List for ERR255009

Container name	Parameter name
CddIpmmuTlb	CddIpmmuTlbId
CddIpmmuPmb	CddIpmmuPmbId
CddIpmmuMmu	CddIpmmuMmuId
CddIpmmuMmuTranslationTable	CddIpmmuMmuTranslationTableId

ERR255010: The configured value <value> for parameter '<parameter_name>' in the container '<container_name>' should follow C syntax '[a-zA-Z][a-zA-Z0-9_]'.

This error occurs, if the value of parameter <parameter_name> does not follow C syntax '[a-zA-Z][a-zA-Z0-9_]'. <parameter_name> could be CddIpmmuPmbCallbackFunction or CddIpmmuMmuCallbackFunction'.

ERR255011: The value of parameter 'CddIpmmuPmbPageSize' is <page_size>, so the parameter

'CddIpmmuPmbVirtualPageNumber' in <container_name> container should be configured in valid range. The valid range: Minimum value is <min>, Maximum value is <max>.
This error occurs, if the configured value of 'CddIpmmuPmbPageSize' is <page_size> but the configured value parameter 'CddIpmmuPmbVirtualPageNumber' is out of range.

ERR255012: The value of parameter 'CddIpmmuPmbPageSize' is <page_size>, so the parameter 'CddIpmmuPmbPhysicalPageNumber' in <container_name> should be configured in valid range. The valid range: Minimum value is <min>, Maximum value is <max>.
This error occurs, if the configured value of 'CddIpmmuPmbPageSize' is <page_size> but the configured value parameter 'CddIpmmuPmbPhysicalPageNumber' is out of range.

ERR255013: The configured value <value> of parameter 'CddIpmmuTlbSelection' should not be same in different 'CddIpmmuTlb' containers which locate in the same domain.
This error occurs, if there are two or more Tlb configuration located in same domain but the configured value of parameter 'CddIpmmuTlbSelection' is also same.

ERR255014: The MMU with 'CddIpmmuMmuSelection' is <mmu_name> should be configured when the value of parameter 'CddIpmmuTlbTranslationTable' in '<container_name>' container is configured as <mmu_name>.
This error occurs, if the configured value of 'CddIpmmuTlbTranslationTable' is <parameter_value> but there is not any value of parameter 'CddIpmmuMmuSelection' is configured as <parameter_value>.

ERR255015: The parameter 'CddIpmmuMmuStartLevel' in 'CddIpmmuMmu' containers is configured as <$recommend_value >, so 'CddIpmmuMmuTranslationSize' should be configured in the valid range [<$min> : <$max>].
This error occurs, if the parameter 'CddIpmmuMmuStartLevel' is configured as MMU_SALV1 and the 'CddIpmmuMmuTranslationSize' is configured from 2 to 7 or if the parameter 'CddIpmmuMmuStartLevel' is configured as MMU_SALV2 and the 'CddIpmmuMmuTranslationSize' is configured from 0 to 1.

ERR255016: The configured value <value> of parameter 'CddIpmmuPmbSelection' should not be same in different 'CddIpmmuPmb' containers which locate in the same domain.
This error occurs, if there are two or more PMB configuration located in same domain but the configured value of parameter 'CddIpmmuPmbSelection' is also same.

ERR255017: The configured value <value> of parameter 'CddIpmmuMmuSelection' should not be same in different 'CddIpmmuMmu' containers.
This error occurs, if the parameter 'CddIpmmuMmuSelection' is configured same value in different 'CddIpmmuMmu' containers.

ERR255018: The value of parameter 'CddIpmmuPmbCallbackFunction' in container 'CddIpmmuPmb' should be configured in same value if they locate in same domain.
This error occurs, if the value of parameter 'CddIpmmuPmbCallbackFunction' in container 'CddIpmmuPmb' which located in same domain is configured in the different value.

ERR255019: The reference path of parameter 'CDDIPMMU_E_INTERRUPT_CONTROLLER_FAILURE' should be configured correctly when 'CddIpmmuUnintendedInterruptCheck' is configured as <TRUE>.

This error occurs if the reference path of parameter 'CDDIPMMU_E_INTERRUPT_CONTROLLER_FAILURE' is not configured when 'CddIpmmuUnintendedInterruptCheck' is configured as TRUE.

ERR255020: The reference path of the parameter 'CDDIPMMU_E_WRITE_VERIFY_FAILURE' should be configured correctly when 'CddIpmmuWriteVerifyCheck' is configured as <TRUE>.
This error occurs, if the reference path of the parameter 'CDDIPMMU_E_WRITE_VERIFY_FAILURE' is not configured when 'CddIpmmuWriteVerifyCheck' is configured as TRUE.

ERR255021: PMB of <domain_name> domain should be configured when the parameter CddIpmmuTlbTranslationTable in '<container_name>' container is configured as <PMB>.
This error occurs, if the configured value of the parameter 'CddIpmmuTlbTranslationTable' in <container_name> is <PMB> but there is no PMB configuration for the <domain_name> domain.

ERR255025: The configured value <value> of parameter 'CddIpmmuPmbCallbackFunction' in container 'CddIpmmuPmb' is duplicated with the configured value of 'CddIpmmuMmuCallbackFunction' in container 'CddIpmmuMmu'.
This error occurs, if function name configured for parameter 'CddIpmmuMmuCallbackFunction' in container 'CddIpmmuMmu' is configured in the same name with the function name which is configured for parameter in 'CddIpmmuPmbCallbackFunction' in container 'CddIpmmuMmu'.

ERR255028: 'DEM' component is not present in the input file(s) when at least one parameter inside 'CddIpmmuDemEventParameterRefs' is configured.
This error will occur, if DEM component is not present in the input ECU Configuration Description File(s) when at least one parameter inside 'CddIpmmuDemEventParameterRefs' is configured.

Table 15-4 Parameter List for ERR255028

Container name	Parameter name
CddIpmmuDemEventParameterRefs	CDDIPMMU_E_WRITE_VERIFY_FAILURE (/ActiveEcuC/Dem/DemConfigSet/CDDIPMMU_E_WRITE_VERIFY_FAILURE)
	CDDIPMMU_E_INTERRUPT_CONTROLLER_FAILURE (/ActiveEcuC/Dem/DemConfigSet/CDDIPMMU_E_INTERRUPT_CONTROLLER_FAILURE)

ERR255029: The reference path 'reference path' provided for the parameter 'Parameter name' within the container 'container name' is duplicated.
This error occurs, if the path provided for the following parameters in the respective container is duplicated.
Note: See "Table 15-4 Parameter List for ERR255028" for detailed information about list of supported parameters.

15.6.2 Specific Warning Messages

WRN255002: The parameter <CDDIPMMU_E_WRITE_VERIFY_FAILURE> in the container <CddIpmmuDemEventParameterRefs> should not be configured since the value of the parameter <CddIpmmuWriteVerifyCheck> of 'CddIpmmuGeneral' container is configured as FALSE.

This warning occurs, if the value of parameter CDDIPMMU_E_WRITE_VERIFY_FAILURE is configured while the parameter CddIpmmuWriteVerifyCheck is configured as FALSE.

WRN255001: The parameter <CDDIPMMU_E_INTERRUPT_CONTROLLER_FAILURE> in the container < CddIpmmuDemEventParameterRefs> should not be configured since the value of the parameter < CddIpmmuUnintendedInterruptCheck> of 'CddIpmmuGeneral' container is configured as FALSE.
This warning occurs, if the value of parameter CDDIPMMU_E_INTERRUPT_CONTROLLER_FAILURE is configured while the parameter CddIpmmuUnintendedInterruptCheck is configured as FALSE.

15.6.3 Specific Information Messages

None.

15.6.4 Specific Deviation List

None.

15.7 Common 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 errors, warning and information messages displayed by the Generation Tool.

For the list of common error/warning/information messages for all modules, refer to "R-Car Gen4 AUTOSAR R19-11 MCAL User's Manual Getting Started".

15.7.1 Common Error Messages

ERR255003: <Module> Component is not present in the input file(s).
This error will occur, if <Module> Driver components mentioned below are not present in the input ECU Configuration Description File(s).

Table 15-5 Modules for ERR255003

Module
IPMMU

ERR255800: The container '<container name>' should be configured.
This error will occur, if any of the mandatory configuration container(s) mentioned below is (are) not configured in ECU Configuration Description File. The list of mandatory containers with respect to container is listed below.

Table 15-6 Parameter List for ERR255800

Container Name
CddIpmmuTlb
CddIpmmuMmuTranslationTable

ERR255801: The number of container '<container name>' should be in valid range from <Min value> to <Max value>.

This error occurs if the number of containers is configured out of valid range.

Table 15-7 Parameter List for ERR255801

Container name	V4M	
	Valid Range	
	Minimum value	Maximum value
CddGeneral	1	1
CddIpmmuTlb	1	640
CddIpmmuPmb	0	160
CddIpmmuMmu	0	16
CddIpmmuMmuTranslationTable	1	2
CddIpmmuDemEventParameterRefs	0	1

ERR255802: The value of parameter '<parameter name>' in the container '<container name>' should be in valid range from <Min value> to <Max value>.

This error occurs, if the configured value of parameter '<parameter name>' is not within valid range from <Min value> to <Max value>.

Table 15-8 Parameter List for ERR255802

Container Name	Parameter Name	Minimum value	Maximum value
CddIpmmuMmu	CddIpmmuMmuId	0	15
	CddIpmmuMmuMemoryAttributeIndirection0	0	4294967295
	CddIpmmuMmuMemoryAttributeIndirection1	0	4294967295
CddIpmmuMmuTranslationTable	CddIpmmuMmuTranslationTableId	0	1
	CddIpmmuMmuLowerBaseAddress	0	1048575
	CddIpmmuMmuUpperBaseAddress	0	255
	CddIpmmuMmuTranslationSize	0	7
CddIpmmuPmb	CddIpmmuPmbId	0	159
	CddIpmmuPmbUpperPhysicalPageNumber	0	255
CddIpmmuTlb	CddIpmmuTlbId	0	639
	CddIpmmuTlbSelection	0	63
	CddIpmmuTlbASID	0	255

ERR255803: Value of parameter '<parameter name>' in container '<container name>' should be a valid enumeration literal value.

This error occurs if Enumeration type parameter is configured in invalid Enumeration type range.

Table 15-9 Parameter List for ERR255803 (1/3)

Container name	Parameter name	Enumeration
		V4M
CddGeneral	CddIpmmuDeviceName	V4M
CddIpmmuTlb	CddIpmmuTlbBusDomain	CDD_IPMMU_VI0
		CDD_IPMMU_VI1
		CDD_IPMMU_VC0
		CDD_IPMMU_IR
		CDD_IPMMU_RT
		CDD_IPMMU_RT1
		CDD_IPMMU_DS0
		CDD_IPMMU_HC
		CDD_IPMMU_3DG
		CDD_IPMMU_VIP0
	CddIpmmuTlbTranslationTable	PMB
		MMU0
		MMU1
		MMU2
		MMU3
		MMU4
		MMU5
		MMU6
		MMU7
		MMU8
		MMU9
		MMU10
		MMU11
		MMU12
		MMU13

Container name	Parameter name	Enumeration
		V4M
		MMU14
		MMU15

Table 15-10 Parameter List for ERR255803 (2/3)

Container name	Parameter name	Enumeration
		V4M
CddIpmmuPmb	CddIpmmuPmbBusDomain	CDD_IPMMU_VI0
		CDD_IPMMU_VI1
		CDD_IPMMU_VC0
		CDD_IPMMU_IR
		CDD_IPMMU_RT
		CDD_IPMMU_RT1
		CDD_IPMMU_DS0
		CDD_IPMMU_HC
		CDD_IPMMU_3DG
		CDD_IPMMU_VIP0
	CddIpmmuPmbPageSize	PMB_SIZE_16MB
		PMB_SIZE_64MB
		PMB_SIZE_128MB
		PMB_SIZE_512MB
	CddIpmmuPmbSelection	PMB0
		PMB1
		PMB2
		PMB3
		PMB4
		PMB5
		PMB6
		PMB7
		PMB8
		PMB9
		PMB10
		PMB11
		PMB12
		PMB13

Container name	Parameter name	Enumeration
		V4M
		PMB14
		PMB15

Table 15-11 Parameter List for ERR255803 (3/3)

Container name	Parameter name	Enumeration
		V4M
CddIpmmuMmu	CddIpmmuMmuAArchSupport	VMSAV8_32
	CddIpmmuMmuSelection	MMU0
		MMU1
		MMU2
		MMU3
		MMU4
		MMU5
		MMU6
		MMU7
		MMU8
		MMU9
		MMU10
		MMU11
		MMU12
		MMU13
		MMU14
		MMU15
	CddIpmmuMmuStartLevel	MMU_SALV1
		MMU_SALV2
	CddIpmmuMmuTranslationTableFormat	MMU_LDTTF
CddIpmmuMmuTranslationTable	CddIpmmuMmuInnerCacheAbility	NON_CACHEABLE
		WRITE_BACK_WA
		WRITE_THROUGH
		WRITE_BACK_NWA
	CddIpmmuMmuOuterCacheAbility	NON_CACHEABLE
		WRITE_BACK_WA
		WRITE_THROUGH

Container name	Parameter name	Enumeration
		V4M
		WRITE_BACK_NWA
	CddIpmmuMmuShareAbility	NON_SHAREABLE
		OUTER_SHAREABLE
		INNER_SHAREABLE

15.7.2 Common Warning Messages

WRN255003: The parameter <parameter_name> in the container <container_name> is not presented.
This warning occurs if the parameter <parameter_name> is not belong to the container <container_name>.

WRN255004: The configured value <value> of parameter '<parameter_name>' in the container '<container_name>' contain <NULL> in middle or last of string, so it will not be generated to CDD_Ipmmu_Cbk.h file.
This warning occurs, if the configured value of parameter <parameter_name> is contain <NULL> in middle or last of string.

15.7.3 Common Information Messages

None.

15.7.4 Common Deviation List

None.

16. EMM

16.1 Overview

The EMM Complex Device Driver module provides the service for initializing the whole CDD EMM structure of the microcontroller.

The EMM Complex Device Driver module comprises of two sections as Embedded Software and the Generation Tool to achieve scalability and configurability.

The document describes the features of the CDD EMM Driver Generation Tool. CDD EMM Driver Generation Tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File and generates EMM Driver C Source and C Header files (CDD_Emm_PBcfg.c and CDD_Emm_Cfg.h).

This document contains information about the options, input, and output files of the EMM Complex Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

16.2 CDD EMM Driver Generation Tool Overview

EMM Complex Device Driver Generation Tool overview is shown below.

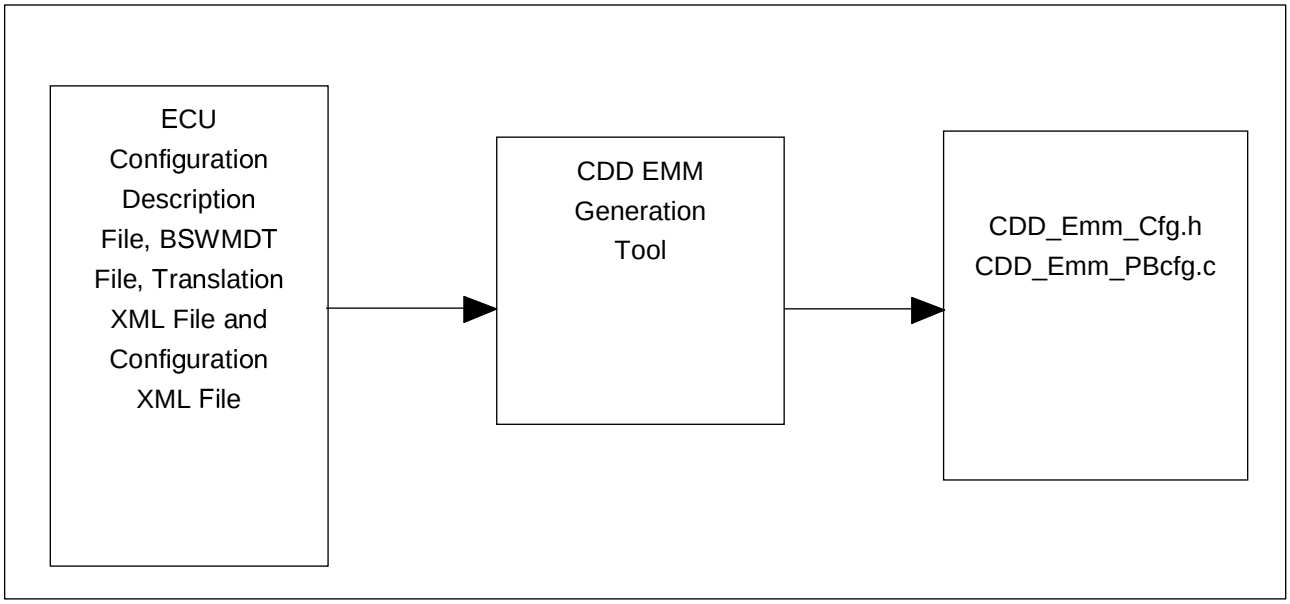


Figure 16-1 Overview of EMM Complex Driver Generation

CDD EMM Driver Generation Tool is a command line tool that extracts, analyzes the configuration details provided in the input file and validates correctness of the data and provides scalability and configurability for CDD EMM Driver module. It accepts ECU Configuration Description File(s), Translation XML File, BSWMDT File and Configuration XML File as input and displays appropriate context sensitive error messages for wrong input and exits. Tool creates the Log file (CddEmm.log) that contains the list of Error/Warning/Information messages in the output directory.

For the error free input file, the tool generates the following output files: C header and C source file names. ECU Configuration Description File can be created or edited using ECU Configuration Editor.

In case of errors the generation tool returns 1, in case of no errors the generation tool returns 0.
EMM Driver Generation Tool uses “Common Published Information” from EMM Driver module specific BSWMDT File. EMM Driver module specific BSWMDT File should not be updated manually since it is “Static Configuration” file.

16.3 Input Files

EMM Driver Generation Tool accepts ECU Configuration Description File(s), Translation XML File, BSWMDT File and Configuration XML File as input. EMM Driver Generation Tool needs information about CDD EMM Driver module.

Hence ECU Configuration Description File should contain configuration of EMM Driver module. Generation 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.

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

16.4 Output Files

EMM Driver Generation Tool generates configuration details in C Header and C Source files (CDD_Emm_Cfg.h and CDD_Emm_PBcfg.c).

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

Table 16-1 Output Files Description

Output File	Details
CDD_Emm_Cfg.h	This file contains pre-compile time parameters configured in the EMM Configuration
CDD_Emm_PBcfg.c	This file contains post-build time parameters configured in the EMM Configuration

Note: Output files generated by CDD EMM Driver Generation Tool should not be modified or edited manually.

16.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19.11 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain EMM Complex Device Driver module.

- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddEmm.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- Translation XML File should contain the file extension '.trxml'.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddEmm.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by EMM Complex Device Driver Generation 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 EMM Complex Driver Generation Tool. Otherwise, Tool may not produce the expected results or may lead to errors/warnings/information messages.
- User has to make sure that the respective device specific configuration file is used.
- The description file should always be generated using AUTOSAR specified configuration editor and it should not be edited manually.
- The configuration <FILTER-RENESAS> should be configured as ON in Configuration XML File (CddEmm_RCar.cfgxml) if same module of different vendor is used with this module of Renesas.

If <FILTER-RENESAS> is ON, the configuration <FILTER-NAME> will contain the name of vendors which will be processed by EMM Generation Tool.

E.g: CDF file contains configuration of Renesas, AUTOSAR and VendorX -> Renesas EMM Generation Tool will only process the configuration of Renesas and AUTOSAR if <FILTER-RENESAS> and <FILTER-NAME> are configured as below:

<FILTER-RENESAS>ON</FILTER-RENESAS>

<FILTER-NAME>Renesas</FILTER-NAME>

<FILTER-NAME>AUTOSAR</FILTER-NAME>

Note: For more information, please refer to the following appendix file: *"ASR_MCAL_R-CarGen4_Deviation_List.xlsx"* in *R-CarGen4 AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual.*

16.6 Specific 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 Generation Tool.

16.6.1 Specific Error Messages

ERR255004: The parameter 'Parameter Name' in the container 'Container Name' should be configured.
This error occurs, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File. The list of mandatory parameters with respect to container is listed below:

Table 16-2 Parameter List for ERR255004

Container	Parameters	Remark
CddGeneral	CddInstanceId	R-Car V4M
	CddEmmDevErrorDetect	R-Car V4M
	CddEmmAlreadyInitDetCheck	R-Car V4M
	CddEmmCriticalSectionProtection	R-Car V4M
	CddEmmUnintendedInterruptCheck	R-Car V4M
	CddEmmVersionCheckExternalModules	R-Car V4M
	CddEmmWriteVerifyCheck	R-Car V4M
	CddEmmVersionInfoApi	R-Car V4M
	CddEmmPseudoErrorApi	R-Car V4M
CddEmmDomain<*n*>	CddEmmAddressToSaveErrorStatus	R-Car V4M
CddEmmBit<*x*><*Error_Assignment_Name*>	CddEmmErrorSignalEnable	R-Car V4M
	CddEmmErrorSignalTarget	R-Car V4M
	CddEmmErrorMaxCount (this parameter is only applied for CddEmmDomain10, CddEmmDomain11 and CddEmmDomain12)	R-Car V4M

Note: CddEmmBit<*x*><Error_Assignment_Name> (*x* = 0..31) presents the identified name of the error signal.

CddEmmDomain<*n*>: <*n*> present the number of supported error domain in each device. *n* = 0..12, 16..36, 38..42 for V4M.

ERR255005: The referenced parameter 'Parameter Name' of the container 'CddEmmDemEventParameterRefs' should be configured since the parameter 'Parameter Name' of the container 'CddGeneral' is configured for use.
This error occurs, if the referenced path of DEM event reference parameter is not configured while the corresponding configuration parameter is set to true.

Table 16-3 Parameter List for ERR255005

Container Name	Parameter Name
CddEmmDemEventParameterRefs	CDDEMM_E_WRITE_VERIFY_FAILURE (/Renesas/EcucDefs_CddEmm/Cdd/CddGeneral/CddEmmWriteVerifyCheck)
	CDDEMM_E_INTERRUPT_CONTROLLER_FAILURE (/Renesas/EcucDefs_CddEmm/Cdd/CddGeneral/CddEmmUnintendedInterruptCheck)

ERR255008: The reference path <DEM Event Reference Path> provided for the DEM event reference parameter within the container 'CddEmmDemEventParameterRefs' is incorrect.

This error occurs, if the path provided for the following parameters in the respective container is incorrect.

Table 16-4 Parameter List for ERR255008

Container Name	Parameter Name
CddEmmDemEventParameterRefs	CDDEMM_E_WRITE_VERIFY_FAILURE (/Renesas/EcucDefs_CddEmm/Cdd/CddEmmDemEventParameterRefs/CDDEMM_E_WRITE_VERIFY_FAILURE)
	CDDEMM_E_INTERRUPT_CONTROLLER_FAILURE (/Renesas/EcucDefs_CddEmm/Cdd/CddEmmDemEventParameterRefs/CDDEMM_E_INTERRUPT_CONTROLLER_FAILURE)

ERR255802: The value configured for parameter 'Parameter Name' in the container 'Container Name' should be in valid range from <Min value> to <Max value>.

This error occurs if the value configured for parameter not in valid range from min value to max value.

Table 16-5 Parameter List for ERR255802

Container Name	Parameter Name	Valid range		Remarks
		Min value	Max Value	
CddGeneral	CddInstanceId	0	255	R-Car V4M
CddEmmDomain<*n*>	CddEmmAddressToSaveErrorStatus	1073741824	3221225471	R-Car V4M
CddEmmDomain10	CddEmmErrorMaxCount	1	31	R-Car V4M
CddEmmDomain11	CddEmmErrorMaxCount	1	31	R-Car V4M
CddEmmDomain12	CddEmmErrorMaxCount	1	31	R-Car V4M

ERR255803: The value of parameter 'Parameter name' in container 'container name' should be configured in valid enumeration literal value.

This error occurs if Enumeration type parameter is configured in invalid Enumeration type range.

Table 16-6 Parameter List for ERR255803

Container Name	Parameter Name	Enumeration
CddGeneral	CddEmmWriteVerifyCheck	WV_INIT_ONLY WV_RUNTIME WV_DISABLE
CddEmmBit<*x*><*Error_Assignment_Name*>	CddEmmErrorSignalTarget	INTC ERROROUT

ERR255007: 'DEM' component is not present in the input file(s) when at least one parameter inside CddEmmDemEventParameterRefs is configured.

This error will occur, if DEM component is not present in the input ECU Configuration Description File(s) when at least one parameter inside 'CddEmmDemEventParameterRefs' is configured.

Table 16-7 Parameter List for ERR255007

Container Name	Parameter Name
CddEmmDemEventParameterRefs	CDDEMM_E_WRITE_VERIFY_FAILURE
	CDDEMM_E_INTERRUPT_CONTROLLER_FAILURE

ERR255009: The reference path 'reference path' provided for the parameter 'Parameter name' within the container 'container name' is duplicated.

This error occurs, if the path provided for the following parameters in the respective container is duplicated.

Note: See *Table 16-4 Parameter List for ERR255008* for detailed information about list of supported parameters.

16.6.2 Specific Warning Messages

WRN255001: The reference path <Reference Path> provided for the parameter <Parameter Name> of the container 'CddEmmDemEventParameterRefs' should not be configured since the parameter <Parameter Name> of the 'CddGeneral' container is disabled.

This warning occurs, if the value of parameter reference path <Reference Path> provided for the DEM event reference parameter <Parameter Name> is configured while the corresponding parameter is disabled.

Table 16-8 Parameter List for WRN255001

Container Name	Parameter Name
CddEmmDemEventParameterRefs	CDDEMM_E_WRITE_VERIFY_FAILURE (/Renesas/EcucDefs_CddEmm/Cdd/CddGeneral/CddEmmWriteVerifyCheck)
	CDDEMM_E_INTERRUPT_CONTROLLER_FAILURE (/Renesas/EcucDefs_CddEmm/Cdd/CddGeneral/ CddEmmUnintendedInterruptCheck)

16.6.3 Specific Information Messages

INF255001: Address <Address Value> is set as user-address to save error status for which error signal belongs to 'CddEmmDomain*n*' with target configured to Interrupt Controller when the error occurred.

This information is generated to inform user the user-address just configured to save error status when the parameter 'CddEmmAddressToSaveErrorStatus' is configured correctly.

16.6.4 Specific Deviation List

None.

16.7 Common 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 errors, warning and information messages displayed by the Generation Tool. For the list of common error/warning/information messages for all modules, refer to *"ASR_MCAL_Generic_GenTool_ErrorList.xlsx"*.

16.7.1 Common Error Messages

ERR255003: 'CDD EMM Driver' component is not present in the input file(s).
This error occurs, if CDD EMM Driver component is not present in the input ECU Configuration Description File(s).

Table 16-9 Modules for ERR255003

Module
EMM

ERR255800: The mandatory container should be configured.
This error will occur, if any of the mandatory configuration container(s) mentioned below is (are) not configured in ECU Configuration Description File. The list of mandatory containers with respect to container is listed below.

Table 16-10 Parameter List for ERR255800

Container Name
CddEmmDomain*n*
CddEmmBit<*x*><*Error_Assignment_Name*>

ERR255801: The number of containers '<Container Name>' should be in valid range from <Min value> to <Max value>.
This error occurs if the number of containers is configured out of valid range.

Table 16-11 Parameter List for ERR255801

Container name	V4M	
	Valid Range	
	Min value	Max value
CddEmmDomainn	1	1
CddEmmBit<*x*><*Error_Assignment_Name*>	1	1
CddGeneral	1	1

CddEmmDemEventParameterRefs	0	1

16.7.2 Common Warning Messages

None.

16.7.3 Common Information Messages

None.

16.7.4 Common Deviation List

None.

17.RFSO

17.1 Overview

The RFSO Complex Driver Component comprises of two parts: Embedded Software and the Generation Tool to achieve scalability and configurability.

The document describes the features of the RFSO Driver Generation Tool. RFSO Driver Generation Tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File and generates RFSO Driver C Source and C Header files (name of output files).

This document contains information about the options, input and output files of the RFSO Complex Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

17.2 RFSO Complex Device Driver Generation Tool Overview

RFSO Complex Device Driver Generation Tool overview is shown below.

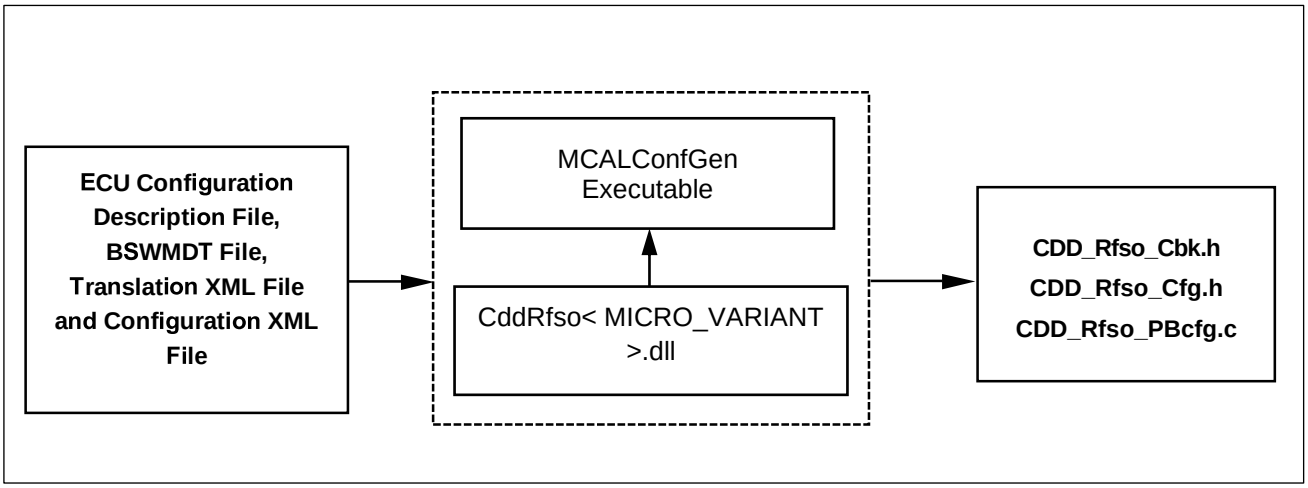


Figure 2.1: Overview of RFSO Complex Driver Generation

RFSO Complex Device Driver Generation Tool is a command line tool that extracts, analyzes the configuration details provided in the input file, validates correctness of the data and provides scalability and configurability for RFSO Complex Driver module. It accepts ECU Configuration Description File(s), Translation XML File, BSWMDT File and Configuration XML File as input, displays appropriate context sensitive error messages for wrong input and exits. Tool creates the Log file (CDD_Rfso.log) that contains the list of Error/Warning/Information messages in the output directory.

For the error free input file, the tool generates the following output files: CDD_Rfso_Cfg.h, CDD_Rfso_PBcfg.c and CDD_Rfso_Cbk.h.

CDD_Rfso_Cfg.h will be compiled and linked with RFSO Complex Driver Component. CDD_Rfso_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash. CDD_Rfso_Cbk.h contain prototype of Callback Functions.

ECU Configuration Description File can be created or edited using ECU Configuration Editor. Please refer to Getting Started document for [R-Car V4M], for more information on how to configure and generate the ECU Configuration Description File.
Remark In case of errors the generation tool returns 1, in case of no errors the generation tool returns 0. RFSO Complex Driver Generation Tool uses “Common Published Information” from RFSO module specific BSWMDT File. RFSO module specific BSWMDT File should not be updated manually since it is “Static Configuration” file.

17.3 Input Files

RFSO Complex Device Driver Generation Tool accepts ECU Configuration Description File(s), Translation XML File, BSWMDT File and Configuration XML File as input.

RFSO Complex Device Driver Generation Tool needs information about RFSO Complex Driver module, therefore the ECU Configuration Description File shall contain the configuration of RFSO Complex Driver module and RFSO related parameters of Dem module. Generation Tool ignores any other AUTOSAR component configured in the ECU Configuration Description File.

Note: The detailed explanation about the parameters and containers are found in Parameter Definition File

17.4 Output Files

RFSO Complex Device Driver Generation Tool generates configuration details in C Header and C Source files (CDD_Rfso_Cfg.h, CDD_Rfso_Cbk.h and CDD_Rfso_PBcfg.c).
The content of each output file is given in the table below:

Table 17.1 Output Files Description

Output File	Details
CDD_Rfso_Cfg.h	This file contains pre-compile time parameters.
CDD_Rfso_PBcfg.c	This file contains post-build time parameters.
CDD_Rfso_Cbk.h	This file contains prototype of callback functions.

Note: Output files generated by RFSO Complex Driver Generation Tool should not be modified or edited manually.

17.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain RFSO Complex Driver module.
- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddRfso.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.
- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddRfso.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.
- Translation XML File should contain the file extension ‘.trxml’.

- Configuration XML File should contain the file extension '.cfgxml'.
- 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.
- If the output files generated by RFSO Complex Driver Generation 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 RFSO Complex Driver Generation Tool. Otherwise Tool may not produce the expected results or may lead to errors/warnings/information messages.
- User has to make sure that the respective device specific configuration file is used otherwise Tool may not produce the expected results or may lead to errors/warnings/information messages.
- The description file should always be generated using AUTOSAR specified configuration editor and it should not be edited manually.
- The configuration <FILTER-RENESAS> should be configured as ON in Configuration XML File (<Module>.cfgxml) if same module of different vendor is used with this module of Renesas. If <FILTER-RENESAS> is ON, the configuration <FILTER-NAME> will contain the name of vendors which will be processed by <Module name> Generation Tool.

Example: CDF file contains configuration of Renesas, AUTOSAR and VendorX -> Renesas <Module name> Generation Tool will only process the configuration of Renesas and AUTOSAR if <FILTER-RENESAS> and <FILTER-NAME> are configured as below:

<FILTER-RENESAS>ON</FILTER-RENESAS>
<FILTER-NAME>Renesas</FILTER-NAME>
<FILTER-NAME>AUTOSAR</FILTER-NAME>

Note: For more information, please refer the following appendix file:
"ARS_MCAL_R_CarGen4_Deviation_List.xlsx" in R-CarGen4 AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual.

17.6 Specific 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 errors, warning and information messages displayed by the Generation Tool.

17.6.1 Specific Error Messages

ERR255006: The reference path <Dem_event_ref> provided for the parameter '<parameter_name>' within the container '<container_name>' is incorrect.
This error will occur, if the path provided for the following parameters in the respective container is incorrect.

Table 17.2 Parameter List for ERR255006

Container Name
CDDRFSO_E_INTERRUPT_CONTROLLER_FAILURE
CDDRFSO_E_WRITE_VERIFY

ERR255008: The value of parameter ‘CddRfsoChannelId’ in the corresponding 'CddRfsoChannel' should be configured in ascending order and continuously from zero.
This error will occur, if parameter ‘CddRfsoChannelId’ in container ‘CddRfsoChannel’ is not configured in ascending order and continuously from zero.

ERR255009: The configured value <value> of parameter ‘CddRfsoIntervalTimerCallbackFunction’ in the container ‘CddRfsoChannel’ should follow C syntax ‘[a-zA-Z][a-zA-Z0-9_]’.
This error occurs, if the value of parameter ‘CddRfsoIntervalTimerCallbackFunction’ does not follow C syntax ‘[a-zA-Z][a-zA-Z0-9_]’.

ERR255010: The configured value <value> of parameter ‘CddRfsoChannelSelection’ should not be same in different ‘CddRfsoChannel’ containers.
This error occurs, if the parameter ‘CddRfsoChannelSelection’ is configured same value in different ‘CddRfsoChannel’ containers.

ERR255011: The reference path of the parameter ‘CDDRFSO_E_INTERRUPT_CONTROLLER_FAILURE’ should be configured correctly when ‘CddRfsoUnintendedInterruptCheck’ is configured as <TRUE>.
This error occurs if the reference path of the parameter ‘CDDRFSO_E_INTERRUPT_CONTROLLER_FAILURE’ is not configured when ‘CddRfsoUnintendedInterruptCheck’ is configured as TRUE.

ERR255012: The reference path of the parameter ‘CDDRFSO_E_WRITE_VERIFY’ should be configured correctly when ‘CddRfsoWriteVerify’ is configured as <TRUE>.
This error occurs, if the reference path of the parameter ‘CDDRFSO_E_WRITE_VERIFY’ is not configured when ‘CddRfsoWriteVerify’ is configured as TRUE.

ERR255017: Regarding to the value of parameter ‘CddRfsoIndicateIntervalUnit’ is <unit_selection>, the value of parameter ‘parameter name’ in container ’CddRfsoChannel’ configured as <value> <parameter_unit> is out of valid range. Valid range is <Min value> to <Max value>.
This error occurs, if parameter ‘CddRfsoIndicateIntervalUnit’ configured <unit_selection> as CDDRFSO_MICROSECOND or CDDRFSO_CYCLE that determines <parameter_unit> of <parameter name> is microsecond(s) or cycle(s), and configured <value> of parameter <parameter name> is out of valid range.

Table 17.3 Parameter List for ERR255017

Parameter Name	Valid Range			
	Minimum value (cycle(s))	Maximum value (cycle(s))	Minimum value (microsecond(s))	Maximum value (microsecond(s))
CddRfsoIntervalTimerDuration	1	4294967295	0.0075* CddRfsoFrequencyDivision	0.0075* 4294967295* CddRfsoFrequencyDivision

Note: **0.0075 (microsecond) = 1/ (Internal clock, 133.33MHZ)**

ERR255018: Regarding to the value of parameter 'CddRfsoIndicateTimeoutUnit' is <unit_selection>, the value of parameter 'parameter name' in container 'CddRfsoChannel' configured as <value> <parameter_unit> is out of valid range. Valid range is <Min value> to <Max value>.

This error occurs, if parameter 'CddRfsoIndicateTimeoutUnit' configured <unit_selection> as CDDRFSO_MICROSECOND or CDDRFSO_CYCLE that determines <parameter_unit> of <parameter name> is microsecond(s) or cycle(s), and configured <value> of parameter <parameter name> is out of valid range.

Table 17.4 Parameter List for ERR255018

Parameter Name	Valid Range			
	Minimum value (cycle(s))	Maximum value (cycle(s))	Minimum value (microsecond(s))	Maximum value (microsecond(s))
CddRfsoTimeoutTimerMaxDuration	1	4294967296	0.0075* CddRfsoFrequencyDivision	0.0075* 4294967296* CddRfsoFrequencyDivision
CddRfsoTimeoutTimerMinDuration	0	4294967295	0	0.0075* 4294967295* CddRfsoFrequencyDivision

Note: **0.0075 (microsecond) = 1/ (Internal clock, 133.33MHZ)**

ERR255019: The value <min_value> <parameter_unit> of parameter 'CddRfsoTimeoutTimerMinDuration' in the container 'CddRfsoChannel' should not equal or exceed the value <max_value> <parameter_unit> of parameter 'CddRfsoTimeoutTimerMaxDuration', 'CddRfsoIndicateTimeoutUnit' is configured as <unit_selection>.

This error occurs, if the value of parameter 'CddRfsoTimeoutTimerMinDuration' equal or exceed the value of parameter 'CddRfsoTimeoutTimerMaxDuration', configured value <unit_selection> of parameter 'CddRfsoIndicateTimeoutUnit' maybe CDDRFSO_MICROSECOND or CDDRFSO_CYCLE that determines <parameter_unit> is microsecond(s) or cycle(s).

ERR255020: The value of parameter 'CddRfsoTimeoutTimerMinDuration' in the container 'CddRfsoChannel' should not set to 1 cycle, parameter 'CddRfsoIndicateTimeoutUnit' is configured as <CDDRFSO_CYCLE>.

This error occurs, if the value of parameter ‘CddRfsoTimeoutTimerMinDuration’ is configured as 1 when parameter ‘CddRfsoIndicateTimeoutUnit’ is configured as <CDDRFSO_CYCLE>.

ERR255021: The value of parameter ’CddRfsoTimeoutTimerMinDuration’ in the container 'CddRfsoChannel' cannot set to <value> microsecond(s). Because this value equals to 1 cycle when converting from microsecond unit to cycle unit, parameter ‘CddRfsoIndicateTimeoutUnit’ is configured as <CDDRFSO_MICROSECOND>.

This error occurs, value of parameter ’CddRfsoTimeoutTimerMinDuration’ equals to 1 cycle when converting from microsecond unit to cycle unit, and parameter ‘CddRfsoIndicateTimeoutUnit’ is configured as <CDDRFSO_MICROSECOND>.

ERR255022: The parameter ‘CddRfsoIntervalTimerCallbackFunction’ in container ‘CddRfsoChannel’ should not contain <NULL> in string.

This error occurs, if the parameter ‘CddRfsoIntervalTimerCallbackFunction’ contain <NULL> in string.

ERR255026: The value <min_value> of parameter ‘CddRfsoTimeoutTimerMinDuration’ and value <max_value> of parameter 'CddRfsoTimeoutTimerMaxDuration' in the container ‘CddRfsoChannel’ should not be equal after converting value from microsecond unit to cycle unit and be rounded, parameter ‘CddRfsoIndicateTimeoutUnit’ is configured as <CDDRFSO_MICROSECOND>.

This error occurs, if the value of the parameter 'CddRfsoTimeoutTimerMinDuration’ and 'CddRfsoTimeoutTimerMaxDuration' are equal after converting from milisecond unit to cycle unit and rounding, 'CddRfsoIndicateTimeoutUnit' is configured as CDDRFSO_MICROSECOND that determines unit of parameter is microsecond.

ERR255028: ‘DEM' component is not present in the input file(s) when at least one parameter inside ‘CddRfsoDemEventParameterRefs’ is configured.

This error will occur, if DEM component is not present in the input ECU Configuration Description File(s) when at least one parameter inside ‘CddRfsoDemEventParameterRefs’ is configured.

Table 17.5 Parameter List for ERR255028

Container	Parameters
CddRfsoDemEventParameterRefs	CDDRFSO_E_WRITE_VERIFY
	CDDRFSO_E_INTERRUPT_CONTROLLER_FAILURE

ERR255029: 'The reference path '<Dem_event_ref>' provided for the parameter 'CDDRFSO_E_INTERRPT_CONTROLLER_FAILURE' and 'CDDRFSO_E_WRITE_VERIFY' within the container '<container_name>' is duplicated.

'This error will occur when two or more optional Dem event parameters are configured with same reference path.

Table 17.6 Parameter List for ERR255029

Container	Parameters
CddRfsoDemEventParameterRefs	CDDRFSO_E_WRITE_VERIFY
	CDDRFSO_E_INTERRUPT_CONTROLLER_FAILURE

17.6.2 Specific Warning Messages

WRN255001: The configured value for the parameter 'CDDRFSO_E_INTERRUPT_CONTROLLER_FAILURE' in the container 'CddRfsoDemEventParameterRefs' should not be configured since the value of the parameter 'CddRfsoUnintendedInterruptCheck' of 'CddGeneral' container is configured as <FALSE>.
This warning occurs, if the value of parameter CDDRFSO_E_INTERRUPT_CONTROLLER_FAILURE is configured while the parameter CddRfsoUnintendedInterruptCheck is configured as FALSE.

WRN255002: The configured value for the parameter 'CDDRFSO_E_WRITE_VERIFY' in the container 'CddRfsoDemEventParameterRefs' should not be configured since the value of the parameter 'CddRfsoWriteVerify' of 'CddGeneral' container is configured as <FALSE>.
This warning occurs, if the value of parameter CDDRFSO_E_WRITE_VERIFY is configured while the parameter CddRfsoWriteVerify is configured as FALSE.

WRN255003: The parameter '<parameter_name>' in the container '<container_name>' is not presented.
This warning occurs, if the parameter '<parameter_name>' is not belong to the container '<container_name>'.

17.6.3 Specific Information Messages

None

17.6.4 Specific Deviation List

None.

17.7 Common 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 errors, warning and information messages displayed by the Generation Tool.

17.7.1 Common Error Messages

ERR255003: <Module> component is not present in the input file(s) or not provided in <STUBS> tag of configuration XML file.

This error will occur, if <Module> Driver component is not present in the input ECU Configuration Description File(s) or not provided in <STUBS> tag of configuration XML file.

Table 17.7 Parameter List for ERR255003

Module
CDD RFSO Driver

ERR255004: The parameter <parameter_name> in the container <container_name> should be configured.

This error will occur, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File. The list of mandatory parameters with respect to container is listed below:

Table 17.8 Parameter List for ERR255004

Container	Parameters
CddGeneral	CddRfsoDevErrorDetect
	CddRfsoVersionInfoApi
	CddRfsoDeviceName
	CddRfsoAlreadyInitDetCheck
	CddRfsoRegisterWriteVerify
	CddRfsoCriticalSectionProtection
	CddRfsoUnintendedInterruptCheck
	CddRfsoVersionCheckExternalModules
	CddInstanceId
CddRfsoChannel	CddRfsoChannelSelection
	CddRfsoChannelId
	CddRfsoFrequencyDivision
	CddRfsoIntervalTimerDuration
	CddRfsoIndicateIntervalUnit
	CddRfsoIntervalTimerCallbackFunction
	CddRfsoTimeoutTimerMaxDuration
	CddRfsoTimeoutTimerMinDuration
	CddRfsoIndicateTimeoutUnit
	CddRfsoIntervalTimerOneShot
	CddRfsoIntervalTimerInterruptEnable

ERR255800: The container '<container name>' should be configured.

This error will occur, if any of the mandatory configuration container(s) mentioned below is (are) not configured in ECU Configuration Description File. The list of mandatory container with respect to container is listed below.

Table 17.9 Parameter List for ERR255800

Container Name
CddRfsoChannel
CddRfsoDemEventParameterRefs

ERR255801: The number of container '<container_name>' should be in valid range from <Min value> to <Max value>.

This error occurs, if the number of container is configured out of valid range.

Table 17.10 Parameter List for ERR255801

Container name	Valid Range	
	Minimum value	Maximum value
	V4M	V4M
CddGeneral	1	1
CddRfsoDemEventParameterRefs	1	1
CddRfsoChannel	1	11

ERR255802: The value of parameter '<parameter name>' in the container '<container name>' should be in valid range from <Min value> to <Max value>.

This error occurs, if the configured value of parameter '<parameter name>' is not within valid range from <Min value> to <Max value>.

Table 17.11 Parameter List for ERR255802

Container Name	Parameter Name	Valid Range	
		Minimum value	Maximum value
CddRfsoChannel	CddRfsoChannelId	0	10
	CddRfsoFrequencyDivision	1	4294967295

ERR255803: Value of parameter '<parameter name>' in container '<container name>' should be a valid enumeration literal value.

This error occurs, if Enumeration type parameter is configured in invalid Enumeration type range.

Table 17.12 Parameter List for ERR255803

Container name	Parameter name	Enumeration
		V4M
CddRfsoChannel	CddRfsoChannelSelection	CHANNEL0 CHANNEL1 CHANNEL2 CHANNEL3 CHANNEL4 CHANNEL5 CHANNEL6 CHANNEL7 CHANNEL8

		CHANNEL9 CHANNEL10
	CddRfsoIndicateIntervalUnit	CDDRFSO_MICROSECOND CDDRFSO_CYCLE
	CddRfsoIndicateTimeoutUnit	CDDRFSO_MICROSECOND CDDRFSO_CYCLE

17.7.2 Common Warning Messages

None

17.7.3 Common Information Messages

None

17.7.4 Common Deviation List

None

18.IIC

18.1 Overview

The IIC Complex Driver module provides the service for initializing the whole CDD IIC structure of the microcontroller.

The IIC Complex Driver module comprises of two sections as Embedded Software and the Generation Tool to achieve scalability and configurability.

The document describes the features of the IIC Complex Driver Generation Tool. IIC Complex Driver Generation Tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File and generates IIC Complex Driver C Source and C Header files (CDD_Iic_PBcfg.c, CDD_Iic_Cfg.h and CDD_Iic_Cbk.h).

This document contains information on the options, input and output files of the IIC Complex Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

ECU Configuration Description File contains information about IIC CDD Configuration.

18.2 IIC Complex Device Driver Generation Tool Overview

IIC Complex Driver Generation Tool overview is shown below.

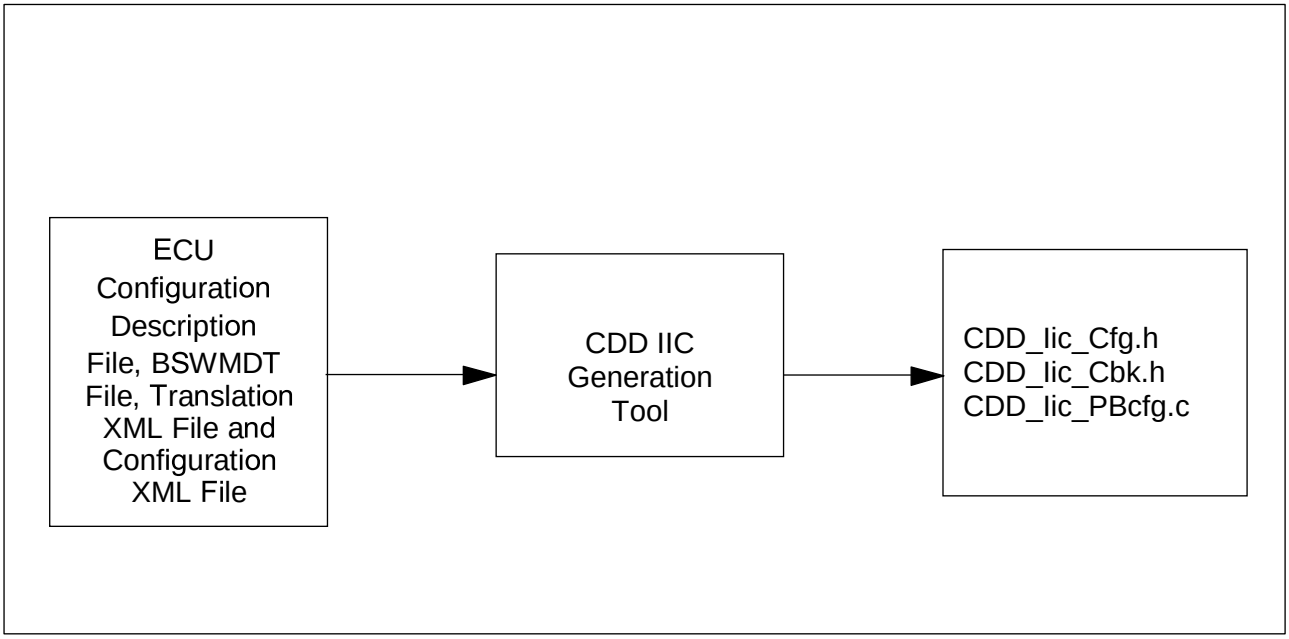


Figure 18-1 Overview of IIC Complex Driver Generation Tool

IIC Complex Driver Generation Tool is a command line tool that extracts, analyzes the configuration details provided in the input file and validates correctness of the data and provides scalability and configurability for IIC Complex Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input and displays appropriate context sensitive error messages for wrong input and exits. Tool creates the Log file (CDD_Iic.log) that contains the list of Error/Warning/Information messages in the output directory.CDD_Iic_Cfg.h and CDD_Iic_Cbk.h will be compiled and linked with IIC Complex Driver

module. CDD_Iic_PBcfg.c will be compiled and linked separately from the other C Source files and placed in flash.

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

For the error free input file, the tool generates the following output files: C Header CDD_Iic_Cfg.h and CDD_Iic_Cbk.h, C source CDD_Iic_PBcfg.c.

Note:

- In case of errors the generation tool returns a 1, in case of no errors the generation tool returns a 0.

IIC Complex Driver Generation Tool uses “Common Published Information” from CDD IIC module specific BSWMDT File. CDD IIC module specific BSWMDT File should not be updated manually since it is “Static Configuration” file.

18.3 Input Files

IIC Complex Driver Generation Tool accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input. IIC Complex Driver Generation Tool needs information about IIC Complex Driver module. Hence ECU Configuration Description File should contain configuration of IIC Complex Driver module. Generation 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.

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

18.4 Output Files

IIC Complex Driver Generation Tool generates configuration details in C Header and C Source files (CDD_Iic_PBcfg.c, CDD_Iic_Cfg.h, CDD_Iic_Cbk.h).

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

Table 18-1 Output Files Description

Output File	Details
CDD_Iic_Cfg.h	This file contains pre-compile time parameters and handles.
CDD_Iic_Cbk.h	This file contains notification callback function prototypes.
CDD_Iic_PBcfg.c	This file contains post-build time parameters.

Note: Output files generated by IIC Complex Driver Generation Tool should not be modified or edited manually.

18.5 Precautions

- ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.
- The input file must contain IIC Complex Driver module.
- If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddIic.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File.

- If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddIic.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File .
- Translation XML File should contain the file extension '.trxml'.
- Configuration XML File should contain the file extension '.cfgxml'
- 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.
- If the output files generated by IIC Complex Driver Generation 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 IIC Complex Driver Generation Tool. Otherwise Tool may not produce the expected results or may lead to error/warning/information.
- The configuration <FILTER-RENESAS> should be configured as ON in Configuration XML File (<Module>.cfgxml) if same module of different vendor is used with this module of Renesas. If <FILTER-RENESAS> is ON, the configuration <FILTER-NAME> will contain the name of vendors which will be processed by <Module name> Generation Tool.
- Example: CDF file contains configuration of Renesas, AUTOSAR and VendorX -> Renesas <Module name> Generation Tool will only process the configuration of Renesas and AUTOSAR if <FILTER-RENESAS> and <FILTER-NAME> are configured as below:

  <FILTER-RENESAS>ON</FILTER-RENESAS>

  <FILTER-NAME>Renesas</FILTER-NAME>

  <FILTER-NAME>AUTOSAR</FILTER-NAME>

Note: Please refer the CDD IIC Component User Manual for deviations from AUTOSAR specifications, if any.

18.6 Specific 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 Generation Tool.

18.6.1 Specific Error Messages

ERR255004: The parameter 'parameter name' in the container 'container name' is not present.

This error occurs, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File. The list of mandatory parameters with respect to container is listed below:

Table 18-2 Parameter List for ERR255004

Container	Parameter
CddGeneral	CddInstanceId
	CddIicVersionInfoApi
	CddIicDevErrorDetect
	CddIicAlreadyInitDetCheck
	CddIicRegisterWriteVerify
	CddIicCriticalSectionProtection
	CddIicVersionCheckExternalModules
	CddIicUnintendedInterruptCheck
CddIicChannel	CddIicChannelId
	CddIicHwChannelSelect

Container	Parameter
	CddIicCommuncationInterface
	CddIicDmaEnable
CddIicDma	CddIicDmaTxChannel
	CddIicDmaRxChannel
CddIicSlave	CddIicSlaveID
	CddIicClockModeSelection
	CddIicSclHighPeriod
	CddIicSclLowPeriod
	CddIicClockFrequency
	CddIicRisingTime
	CddIicFallingTime
	CddIicIntDelay
	CddIicSlaveAddress
	CddIicAddressModeSelect
	CddIicFirstBitSetupCycle
CddIicDemEventParameterRefs	CDDIIC_E_NON_ACKNOWLEGEMENT
	CDDIIC_E_WRITE_VERIFY
	CDDIIC_E_INTERRUPT_CONTROLLER_FAILURE

ERR255011: The value configured for the parameter 'CddIicHWChannelSelect' is duplicated.
This error occurs, if the more than 1 channel select the same IIC hardware channel.

ERR255013: The reference path 'reference_path' provided for the parameter 'parameter_name' within the container 'container name' is incorrect.
This error occurs, if the path provided for the following parameters in the respective container is incorrect.

Table 18-3 Parameter List for ERR255013

Container	Parameter
CddIicDemEventParameterRefs	CDDIIC_E_NON_ACKNOWLEGEMENT
	CDDIIC_E_WRITE_VERIFY
	CDDIIC_E_INTERRUPT_CONTROLLER_FAILURE

ERR255802: The value of parameter ' parameter_name ' in container 'container_name ' should be in valid range from <min_value> to <max_value>.
This error occur, if the value of the parameters is configured out of range <min_vale> and <max_value>

Table 18-4 Parameter List for ERR255802

Container	Parameters	Range		Device
		Min Value	Max Value	
CddIicChannel	CddIicChannelId	0	3	V4M
CddIicSlave	CddIicSlaveID	0	1023	V4M
	CddIicSclHighPeriod	0.26	491.56	V4M
	CddIicSclLowPeriod	0.5	491.56	V4M

	CddIicClockFrequency	1	1000000	V4M
	CddIicRisingTime	0	120	V4M
	CddIicFallingTime	0	120	V4M
	CddIicIntDelay	0	110	V4M
	CddIicSlaveAddress	0	1023	V4M
	CddIicFirstBitSetupCycle	2	33	V4M

ERR255803: Value of parameter 'parameter_name', in container 'container_name' should be a valid enumeration literal value.

This error occur, if the enumeration parameter is not configured in the valid range.

Table 18-5 Parameter List for ERR255803

Container	Parameter	Value	Device
CddIicChannel	CddIicHwChannelSelect	IIC0	V4M
		IIC1	V4M
		IIC2	V4M
		IIC3	V4M
	CddIicCommunicationInterface	MASTER_INTERFACE	V4M
		SLAVE_INTERFACE	V4M
CddIicDma	CddIicDmaTxChannel	DMAn , n= 00 .. 23	V4M
	CddIicDmaRxChannel	DMAn , n= 00 .. 23	V4M
CddIicSlave	CddIicClockModeSelection	FIXED_DUTY	V4M
		VARIABLE_DUTY	V4M
	CddIicAddressModeSelect	SEVEN_BIT_ADDR	V4M
		TEN_BIT_ADDR	V4M

ERR255023: To check The reference path provided for the parameters within the container CddIicDemEventParameterRefs is duplicated

This error occur, if there is a parameter in container CddIicDemEventParameterRefs have reference path as same as other parameter in container CddIicDemEventParameterRefs

Table 18-6 Parameter List for ERR255023

Container	Parameter
CddIicDemEventParameterRefs	CDDIIC_E_NON_ACKNOWLEGEMENT
	CDDIIC_E_WRITE_VERIFY
	CDDIIC_E_INTERRUPT_CONTROLLER_FAILURE

18.6.2 Specific Warning Messages

WRN255001: The value configured for the parameter <param_name> in the container <container_name> should not be configured since the value of the parameter <relative_param_name> of 'CddGeneral' container is configured as <false>.

This warning occurs, if the value of a parameter is configured (below table) while the parameter its relative parameter is configured as 'false'.

Table 18-7 Parameter List for WRN255001

Parameter	Relative Parameter
CDDIIC_E_REGISTER_WRITE_VERIFY	CddIicRegisterWriteVerify = <false>
CDDIIC_E_INTERRUPT_CONTROLLER_FAILURE	CddIicUnintendedInterruptCheck = <false>

18.6.3 Specific Information Messages

INF255001: The value of parameter 'CddIicClockModeSelection' is configured as 'VARIABLE_DUTY' The actual clock frequency is 'actual_clock_value' in comparision with the target clock frequency 'target_clock_value'

This information message when the parameter 'CddIicClockModeSelection' is configured as 'VARIABLE_DUTY'

18.6.4 Specific Deviation List

None.

18.7 Common 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 common errors, warning and information messages displayed by the Generation Tool.

18.7.1 Common Error Messages

ERR255003: 'Module' component is not present in the input file(s).

This error will occur, if <Module> Driver components mentioned below are not present in the input ECU Configuration Description File(s).

Table 18-8 Modules for ERR255003

Module
Cddlic
Dem

ERR255800: The container 'container_name' should be configured.
This error occur, if the container is not configured when its lower multiplicity is not '0'.

Table 18-9 List of containers should be configured

Container	Device
CddGeneral	V4M
CddIicChannel	V4M
CddIicDemEventParameterRefs	V4M

ERR255801: The number of container ‘container_name’, should be in valid range from <min_value> to < max_value >.
This error occur, if the number of the container in not configured in the valid range.

Table 18-10 List of containers and its valid range

Container	Range		Device
	Min	Max	
CddGeneral	1	1	V4M
CddIicChannel	1	4	V4M
CddIicDemEventParameterRefs	1	1	V4M
CddIicSlave	0	1024	V4M

ERR255006: The value configured for the parameter ‘CddIicChannelId’ is not unique.
This error occurs, if parameter CddIicChannelID is configured the same value across multiple channels

ERR255007: The value of the parameter ‘CddIicChannelID’ configured in container ‘CddIicChannel' should be ‘index’. In general per configuration set, the value of CddIicChannelID’ parameter should start with 0 and should be sequential without any gaps.
This error occur, if the channel ID of channels is not start from 0 and is not sequence.

ERR255008: The value configured for the parameter ‘CddIicSlaveId’ is not unique.
This error occurs, if parameter CddIicSlaveId is configured the same value across multiple slaves

ERR255009: The value of the parameter ‘CddIicSlaveID’ configured in container ‘CddIicSlave’ should be ‘index’. In general per configuration set, the value of ‘CddIicSlaveID’ parameter should start with 0 and should be sequential without any gaps.
This error occur, if the channel ID of channels is not start from 0 and is not sequence.

ERR255010: The value configured for the parameter ‘CddIicChannelNotification’ is duplicated.
This error occurs, if the channel notification function was configured in the same function name across multiple channels

ERR255014: The value configured for the parameter ‘CddIicChannelNotification’ should follow C syntax [a-zA-Z][a-zA-Z0-9_].
This error occurs, if the configured value of CddIicChannelNotification is not a valid C function name.

ERR255015: The container ‘CddIicDma’ shall be configured since the value of parameter ‘CddIicDmaEnable’ is configured as <true>
This error occurs, if the container CddIicDma is not configured while the parameter CddIicDmaEnable is configured as <true>

ERR255016: The container ‘CddIicSlave’ shall be configured at least once since the value parameter ‘parameter_name’ in container ‘container_name’ is configured as ‘MASTER_INTERFACE’
This error occurs, if the container ‘CddIicSlave is not configure while there have a channel is configured as Master

ERR255018: The value of parameter ‘CddIicSlaveAddress” should be configure in 7-bit range (0…127) since the value of parameter “CddIicAddressModeSelect” is configured as “SEVEN_BIT_ADDR”
This error occur, if the parameter ‘CddIicSlaveAddress” is configure as "SEVEN_BIT_ADDR", but the value of parameter "CddIicSlaveAddress" in container "CddIicSlave" is configure more than 7-bit range (0…127).

ERR255021: The value of parameter 'dma_parameter_name' configured in container ‘channel_name’ is duplicated. For each channel configured, the value of ‘dma_parameter_name’ should be unique.
This error occur, if the value of dma parameter is duplicate according to multiple channel configuration.

ERR255022: The value of Scl Clock Generation Divider is 'scgd_value' and it shall be configured in range of 0...63.

This error occur, if the calculation of SCGD according to equation in section “Preconditions” of IIC in Component User Manual” is out of range.

18.7.2 Common Warning Messages

None.

18.7.3 Common Information Messages

INF255002: The value of parameter ‘parameter_name’ in container ‘container_name’ will be ignore while the value of parameter ‘CddIicClockModeSelection’ in container ‘CddIicSlave’ is configured as ‘FIXED_DUTY”

This information message inform that the parameter “CddIicSclHighPeriod” or “CddIicSclLowPeriod” will be ignore if the parameter ‘CddIicClockModeSelection’ in container ‘CddIicSlave’ is configured as ‘FIXED_DUTY’

18.7.4 Common Deviation List

None.

19.CRC

19.1 Overview

The CRC Complex Driver module provides the service for initializing the whole CDD CRC structure of the microcontroller.

The CRC Complex Driver module comprises of two sections as Embedded Software and the Generation Tool to achieve scalability and configurability.

The document describes the features of the CRC Complex Driver Generation Tool. CRC Complex Driver Generation Tool is a command line tool that extracts information from ECU Configuration Description File and BSWMDT File and generates CRC Complex Driver C Header and C Source files (CDD_Crc_Cfg.h, CDD_Crc_PBcfg.c and CDD_Crc_Cbk.h).

This document contains information on the options, input and output files of the CRC Complex Driver Generation Tool. In addition, this manual covers a step-by-step procedure for the usage of tool.

ECU Configuration Description File contains information about CRC CDD configuration.

19.2 CRC Driver Generation Tool Overview

CRC Complex Driver Generation Tool overview is shown in **Figure 19.1**.

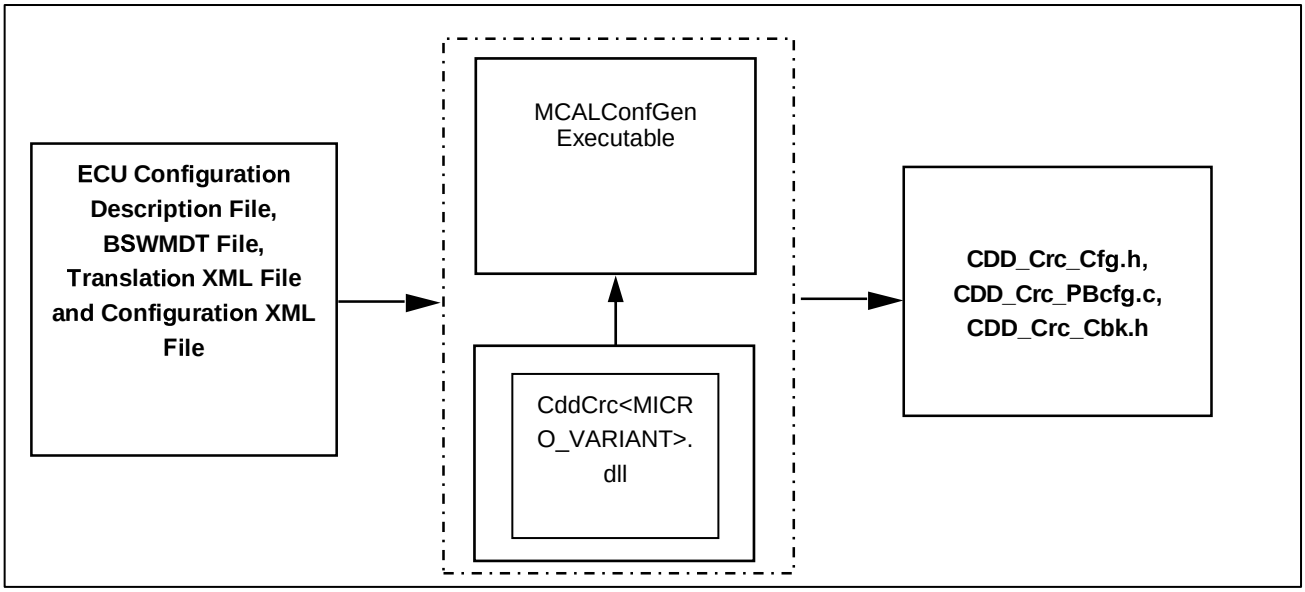


Figure 19.1 Overview of CRC Complex Driver Generation

CRC Complex Driver Generation Tool is a command line tool that extracts, analyzes the configuration details provided in the input file and validates correctness of the data and provides scalability and configurability for CRC Complex Driver module. It accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input and displays appropriate context sensitive error messages for wrong input and exits. Tool creates the Log file (CddCrc.log) that contains the list of Error/Warning/Information messages in the output directory.

CDD_Crc_Cfg.h will be compiled and linked with CRC Complex Driver module. CDD_Crc_PBcfg.c will be compiled and linked separately from other C Source files and linked separately from the other C Source files and placed in flash. CDD_Crc_Cbk.h contain prototupe of Callback Functions.

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

For the error free input file, the tool generates the following output files: CDD_Crc_Cfg.h, CDD_Crc_PBcfg.c and CDD_Crc_Cbk.h.

Remark

- In case of errors the generation tool returns 1, in case of no errors the generation tool returns 0.
- CRC Complex Driver Generation Tool uses “Common Published Information” from CRC module specific BSWMDT File. CRC module specific BSWMDT File should not be updated manually since it is “Static Configuration” file.

19.3 Input Files

CRC Complex Driver Generation Tool accepts ECU Configuration Description File(s), BSWMDT File, Translation XML File and Configuration XML File as input. CRC Complex Driver Generation Tool needs information about CRC Complex Driver module.

Hence ECU Configuration Description File should contain configuration of CRC Complex Driver module. Generation 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.

Note The detailed explanations about the parameters and containers are found in Parameter Definition File.

19.4 Output Files

CRC Complex Driver Generation Tool generates configuration details in C Header and C Source files (CDD_Crc_Cfg.h, CDD_Crc_Cbk.h and CDD_Crc_PBcfg.c).

The content of each output file is given in the following table

Table 19.1 Output Files Description

Output File	Details
CDD_Crc_Cfg.h	This file contains pre-compile time parameters.
CDD_Crc_PBcfg.c	This file contains post-build time configuration data.
CDD_Crc_Cbk.h	This file contains prototype of callback functions.

Note Output files generated by CRC Complex Driver Generation Tool should not be modified or edited manually.

19.5 Precautions

ECU Configuration Description File and BSWMDT File must comply with AUTOSAR standard for R19-11 ECU Configuration Description File and BSWMDT File respectively.

The input file must contain CRC Complex Device Driver module.

If a Translation XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddCrc.trxml which is presented in the same location of MCALConfGen.exe as the input Translation XML File..

If a Configuration XML File is provided on command line, the Generation Tool will get this file as input. Otherwise, the Generation Tool will get the file CddCrc.cfgxml which is presented in the same location of MCALConfGen.exe as the input Configuration XML File.

Translation XML File should contain the file extension '.trxml'.

Configuration XML File should contain the file extension '.cfgxml'.

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.

If the output files generated by CRC Complex Device Driver Generation 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 CRC Complex Device Driver Generation Tool. Otherwise, Tool may not produce the expected results or may lead to errors/warnings/information's.

The configuration <FILTER-RENESAS> should be configured as ON in Configuration XML File (<Module>.cfgxml) if same module of different vendor is used with this module of Renesas. If <FILTER-RENESAS> is ON, the configuration <FILTER-NAME> will contain the name of vendors which will be processed by <Module name> Generation Tool.

Example: CDF file contains configuration of Renesas, AUTOSAR and VendorX -> Renesas <Module name> Generation Tool will only process the configuration of Renesas and AUTOSAR if <FILTER-RENESAS> and <FILTER-NAME> are configured as below:

```
<FILTER-RENESAS>ON</FILTER-RENESAS>

<FILTER-NAME>Renesas</FILTER-NAME>

<FILTER-NAME>AUTOSAR</FILTER-NAME>
```

Note: For more information, please refer to the following appendix file: "*ASR_MCAL_R-CarGen4_Deviation_List.xlsx*" in R-Car Gen4 AUTOSAR R19-11 MCAL User's Manual Driver Component Embedded User's Manual

19.6 Specific 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 errors, warning and information messages displayed by the Generation Tool.

19.6.1 Specific Error Messages

None.

19.6.2 Specific Warning Messages

None.

19.6.3 Specific Information Messages

None.

19.6.4 Specific Deviation List

None.

19.7 Common 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 errors, warning and information messages displayed by the Generation Tool.

19.7.1 Common Error Messages

ERR255003: <Component_name> component is not present in the input file(s).

This error occurs, if <Component_name> component is not present in the input ECU Configuration Description File(s).

ERR255004: The parameter 'parameter name' in the container 'container name' should be configured.

This error occurs, if any of the mandatory configuration parameter(s) mentioned below is (are) not configured in ECU Configuration Description File. The list of mandatory parameters with respect to container is listed below:

Container	Parameters
CddGeneral	CddCrcDevErrorDetect
	CddCrcVersionInfoApi
	CddCrcVersionCheckExternalModules
	CddCrcAlreadyInitDetCheck
	CddCrcRegisterWriteVerify
	CddCrcCriticalSectionProtection
	CddCrcUnintendedInterruptCheck
	CddCrcUnintendedModuleStopCheck
	CddCrcEdcErrorDetect
CddCrcCRCModule	CddCrcInBitSwapMode
	CddCrcInByteSwapMode
	CddCrcInXorValMode
	CddCrcOutBitSwapMode
	CddCrcOutByteSwapMode
	CddCrcOutXorValMode
	CddCrcPoly
CddCrcKCRCModule	CddCrcInReflectMode
	CddCrcOutReflectMode
	CddCrcPoly
	CddCrcShiftMode
	CddCrcXorVal
CddCrcChannelConfiguration	CddCrcDmaDataInChannel
	CddCrcDmaDataResChannel
	CddCrcDmaDataOutChannel
	CddCrcDmaCommandChannel
	CddCrcChannelID
	CddCrcHWSelection

ERR255011: The reference parameter 'reference parameter name' of the container <CddCrcDemEventParameterRefs> should be configured, since the value of the parameter 'parameter name' of the container <CddCrcGeneral> is configured as <true>.

Parameter	Reference Parameter
CddCrcRegisterWriteVerify	CDDCRC_E_WRITE_VERIFY
CddCrcUnintendedInterruptCheck	CDDCRC_E_INTERRUPT_CONTROLLER_FAILURE
CddCrcUnintendedModuleStopCheck	CDDCRC_E_UNINTENDED_MODULE_STOP_FAILURE

This error will occur, if the parameter CddCrcRegisterWriteVerify/ CddCrcUnintendedInterruptCheck/ CddCrcUnintendedModuleStopCheck was configured as 'true' while the reference parameter CDDCRC_E_WRITE_VERIFY/ CDDCRC_E_INTERRUPT_CONTROLLER_FAILURE/ CDDCRC_E_UNINTENDED_MODULE_STOP_FAILURE was not configured.

ERR255012: The sub-container <sub-container name> should not be configured since the hardware selection in container <CddCrcChannelConfiguration> is <CddCrcHWSelection>.

This error will occur, if the sub-container is not configured based on the corresponding HW selection CddCrcHWSelection of CddCrcChannelConfiguration.

ERR255013: The value configured for the parameter <Parameter Name> in <Container Name> is duplicated.

This issue will occur if a HW DMA channel is configured twice or more.

Container name	Parameter name
CddCrcChannelConfiguration	CddCrcDmaDataInChannel
	CddCrcDmaDataResChannel
	CddCrcDmaDataOutChannel
	CddCrcDmaCommandChannel

ERR255014: The value <configured value> configured for the parameter <parameter name> present in the container <container name> should be unique in all instances.

This error will occur, if the value configured for the following parameters in the respective container is not unique in CDFs.

Container name	Parameter name
CddCrcChannelConfiguration	CddCrcHWSelection

ERR255015: Exactly one sub-container is allowed in container <container name> since HW selection is CRC/KCRC.

This error will occur, if number of sub-containers configured in CddCrcChannelConfiguration that select CRC/KCRC hardware is not equal to 1.

ERR255017: In general, the value of parameter <parameter name> in container <container name> should be started at <0> and should be sequential without any gaps.

This error will occur, if the following parameters does not start with 0 and is not continuous in the corresponding container.

The channel IDs can be configured in any order but there should not be any gap in the channel Ids configured. For eg: if there are 3 channels in a configset then the possible IDs are 0, 1, 2 in any order.

Container name	Parameter name
CddCrcChannelConfiguration	CddCrcChannelID

ERR255020: The reference path <path> provided for the DEM parameter <Parameter Name> within the container <Container Name> is incorrect.

This error will occur , if path provided for the DEM parameter in "CddCrcDemEventParameterRefs" of table below is incorrect.

Container	Parameter

CddCrcDemEventParameterRefs	CDDCRC_E_WRITE_VERIFY
	CDDCRC_E_INTERRUPT_CONTROLLER_FAILURE
	CDDCRC_E_UNINTENDED_MODULE_STOP_FAILURE
	CDDCRC_E_HARDWARE_ERROR

Note - Only check for CDDCRC_E_WRITE_VERIFY if CddCrcRegisterWriteVerify is configured <TRUE> in CddGeneral.
- Only check for CDDCRC_E_INTERRUPT_CONTROLLER_FAILURE if CddCrcUnintendedInterruptCheck is configured <TRUE> in CddGeneral.
- Only check for CDDCRC_E_UNINTENDED_MODULE_STOP_FAILURE if CddCrcUnintendedModuleStopCheck is configured <TRUE> in CddGeneral

ERR255022: The reference path of dem event <reference parameter name> should be unique in all instances.

This error will occur if the reference path of any reference parameter is not unique.

Container	Parameter
CddCrcDemEventParameterRefs	CDDCRC_E_WRITE_VERIFY
	CDDCRC_E_INTERRUPT_CONTROLLER_FAILURE
	CDDCRC_E_UNINTENDED_MODULE_STOP_FAILURE
	CDDCRC_E_HARDWARE_ERROR

19.7.2 Common Warning Messages

WRN255001: The value configured for the parameter <Reference Parameter Name> in the container <CddCrcDemEventParameterRefs> should not be configured since the value of the parameter <Corresponding Parameter Name> of 'CddCrcGeneral' container is configured as <false>.

This warning occurs, if the value of parameter reference path <Reference Path> provided for the DEM event reference parameter <Parameter Name> is configured while the corresponding parameter is disabled.

Parameter	Reference Parameter
CddCrcRegisterWriteVerify	CDDCRC_E_WRITE_VERIFY
CddCrcUnintendedInterruptCheck	CDDCRC_E_INTERRUPT_CONTROLLER_FAILURE
CddCrcUnintendedModuleStopCheck	CDDCRC_E_UNINTENDED_MODULE_STOP_FAILURE

19.7.3 Common Information Messages

None

19.7.4 Common Deviation List

None.

Revision History		R-Car V4M AUTOSAR R19-11 MCAL User's Manual	
Rev.	**Date**	**Description**	
		Page	**Summary**
0.12	Jan 22. 2025	-	Cover, footer and colophon: - Update Rev and issue date. 1.Introduction: - Update Release Version in Table 1-1, 1-2 - Update Version in Table 1-2. 2.Reference Documents - Update references version at Table 2.1 Reference Documents(1/2).
0.11	Dec 23. 2024	-	Cover, footer and colophon: - Update Rev and issue date. 1.Introduction: - Update Release Version in Table 1-1, 1-2 - Update Version in Table 1-2. 2.Reference Documents - Update references version at Table 2.1 Reference Documents(1/2). 9. ICCOM: - Section 9.6: Add CDDICCOM_E_DEINIT_NEGOTIATION to parameter list of error ERR255004, ERR255012, ERR255016 10. MCU: - Section 10.6.1 Specific Error Messages: change parameter name from McuWakeUpFactor to McuPSOWakeUpFactor, add 2 parameters: McuC4PowerOn, McuTimeOutC4ModeChange.
0.10	Oct 29. 2024	-	Cover, footer and colophon: - Update Rev and issue date. 1.Introduction: - Update Release Version in Table 1-1, 1-2 - Update Version in Table 1-2. 2.Reference Documents - Update references version at Table 2.1 Reference Documents(1/2).
0.09	Sep 26. 2024	-	Cover, footer and colophon: - Update Rev and issue date. 1.Introduction: - Update Release Version in Table 1-1, 1-2 - Update Version in Table 1-2. 2.Reference Documents - Update references version at Table 2.1 Reference Documents(1/2). 4.CAN: - Section 4.6.1 Specific Error Messages: update information of message ERR080137.
0.08	Mar 27. 2024	-	Cover, footer and colophon: - Update Rev and issue date. 1.Introduction: - Update Release Version in Table 1.1, 1.2 2.Reference Documents - Update references version at Table 2.1 Reference Documents(1/2).

0.07	Feb 27, 2024	-	Cover, footer and colophon: - Update Rev and issue date. 1.Introduction: - Update Release Version in Table 1.1, 1.2 2.Reference Documents - Update references version at Table 2.1 Reference Documents(1/2).
0.06	Jan 29, 2024	-	Cover, footer and colophon: - Update Rev and issue date. 1.Introduction: - Update Release Version in Table 1.1, 1.2 2.Reference Documents - Update references version at Table 2.1 Reference Documents(1/2). 10.MCU Section 10.6.1 Specific Error Messages: - Update Table 10.5, 10.6 “The list of mandatory parameters” - Update Table 10.14 “The Parameters List of DEM Container”
0.05	Nov 29, 2023	—	Cover, footer and colophon: - Update Rev and issue date. 1.Introduction: - Update Release Version in Table 1.1 2.Reference Documents - Update references version at Table 2.1 Reference Documents(1/2). 14.THS Sections 14.6 and 14.7: - Update reference to “R-Car Gen4 AUTOSAR R19-11 MCAL User’s Manual Getting Started”. 16. EMM: Section 16.6.1: - Update number of domains in note for ERR255004.
0.04	Oct 27, 2023	—	Add V4M Abbreviation Section 1: - Add V4M device in description, Table 1-1 Supported MCAL Product Release Version, Table 1-2 Supported Generation Tool Version, Table 1-3 Document Overview Section 2: - Add V4M device in Table 2-1 Reference Documents(1/2), Table 2-2 Reference Documents(2/2)
0.03	Aug 15, 2023	—	15. THS: - Section 15.6.2 Specific Warning Messages: Update the description for WRN255001
0.02	Jun 16, 2023	—	Add ETH module
0.01	May 15, 2023	—	First Edition issued

R-Car V4M AUTOSAR R19-11 MCAL User's Manual
Driver Component
Generation Tool User's Manual

Publication Date: Rev.0.12 Jan 22, 2025

Published by: Renesas Electronics Corporation

R-Car V4M AUTOSAR R19-11 MCAL
User's Manual

Renesas Electronics Corporation