HYUNDAI AUTOEVER

CanIf User Manual

DOC. NO

SCOPE OF APPLICATION All Project/Engineering
Responsibility : Classic AUTOSAR Team

File Name CanIf_UM.pdf

Creation Jaeho Yang 2025/01/24
Check Hyungmin Shin 2025/01/24
Approval DuckJoo Lee 2025/01/24
Edition Date: 2025/01/24
Document Management System

HYUNDAI
AutoEver

Document Change Histroy

Date (YYYY-MM-DD)	Ver.	Editor	Chap	Description (before -> after revision)
2016-04-08	1.5.3	Jongsun Lim	All	• CanIf UM separated
2016-06-03	1.5.4	Kyungtae Kim	All	#5123
2016-06-13	1.5.5	Kyungtae Kim	All	#5139
2016-06-28	1.5.6	Kyungtae Kim	All	#5334
2016-08-21	1.5.7	Kyungtae Kim	4.2	• CanIf module version updated
2016-11-15	1.5.8	Jongsun Lim	4.2 4.3.1	• CanIf module version updated to 1.5.8 • 4.3.1 Change Log updated
2016-11-24	1.5.9	Jongsun Lim	4.2 4.3.1 5.5	• CanIf module version updated to 1.5.9 • 4.3.1 Change Log updated • Added comments in the DET section
2016-01-03	2.5.0	Jongsun Lim	4.2 4.3.1 4.4.1 4.4.2 5.5 5.9	• CanIf module version updated to 2.5.0 • Change Log updated • Limitations items edited • Deleted unnecessary items in the Deviations section • Added necessary items in CanIfPublicCfg • Added items in CanIfTrcvDrvCfg
2017-03-30	2.5.1	Jongsun Lim	4.2 4.3.1 5.8	• CanIf module version updated to 2.5.1 • Change Log updated • CanIfBufferSize Category changed
2017-04-20	2.5.2	Jongsun Lim	4.2 4.3.1 5.5	• CanIf module version updated to 2.5.2 • Change Log updated • CanIfCANFDID16BitSupport,CanIfCANFDDiscreteDlcSupport added
2017-09-11	2.5.3	Jongsun Lim	5.5 4.2 4.3.1 6.3.5	• CanIfPublicVersionInfoApi edited • CanIf module version updated to 2.5.3 • Change Log updated • Add CanIf_GetTxConfirmationState Function Description
2017-10-12	2.5.4	Chan Kim	4.2 4.3.1 4.4.2 5.6 8.1	• CanIf module version updated to 2.5.4 • Change Log updated • Items added to the Deviations section • Description added for CanIfPduReportIdsMEnable • Appendix added
2018-01-12	2.5.5	Jongsun Lim	4.2 4.3.1 5.6	• CanIf module version updated to 2.5.5 • Change Log updated • Category fixed and description added for CanIfRxPduDlc
2018-05-24	2.5.6	Jongsun Lim	4.2 4.3.1 5.4	• CanIf module version updated to 2.5.6 • Change Log updated • CanIfPrivateDlcCheck attribute changed
2018-12-28	2.5.7	Jongsun Lim	4.2 4.3.1	• CanIf module version updated to 2.5.7 • Change Log updated

			5.1 4.4.1	• CanIfCtrlDrvTxCancellation attribute changed • Limitations items deleted
2019-08-22	2.6.0	Jongsun Lim	4.2 4.3 5.5	• CanIf module version updated to 2.6.0 • Change Log updated • Category fixed for CanIfPublicDevErrorDetect Category and content added for CanIfBusLoadDetectingSupport
2019-10-07	2.6.0.0	Jongsun Lim	4.2 4.3	• CanIf version updated • Change Log updated
2019-12-17	2.6.1.0	Jongsun Lim	4.2 4.3	• CanIf version updated • Change Log updated
2020-01-06	2.6.1.1	Jongsun Lim	4.3	• Change Log updated
2020-05-15	2.6.2.0	Seungjae Kim	4.2 4.3	• Scope of the release changed • Change Log updated
2020-10-20	2.7.0.0	Seungjae Kim	4.2 4.3 4.4.1 5.5	• Scope of the release changed • Change Log updated • Limitations items edited and deleted • CanIfCANFDID16BitSupport fixed; and content added for CanIfPublicCanDrvVersion
2020-12-16	2.7.1.0	Seungjae Kim	4.2 4.3	• Scope of the release changed • Change Log updated
2021-01-12	2.7.2.0	Seungjae Kim	4.2 4.3	• Scope of the release changed • Change Log updated
2021-01-18	2.7.3.0	Seungjae Kim	4.2 4.3 4.4.1 5.1 5.5 7.2	• Scope of the release changed • Change Log updated • Limitations of CanIfPublicTxConfirmPollingSupport added • CanIfCtrlDrvTxCancellation attribute changed • CanIfPublicTxConfirmPollingSupport attribute changed • Generator Error Message added
2021-04-21	2.7.4.0	Seungjae Kim	4.2 4.3	• Scope of the release changed • Change Log updated
2021-05-27	2.7.5.0	Seungjae Kim	4.2 4.3	• Scope of the release changed • Change Log updated
2021-09-07	2.7.6.0	Seungjae Kim	4.2 4.3 8	• Scope of the release changed • Change Log updated • Det error added
2021-12-01	3.0.0.0	Seungjae Kim	4.2 4.3 9.2	• Scope of the release changed • Change Log updated • PostBuild settings added
2021-12-28	3.0.1.0	Seungjae Kim	4.2 4.3	• Scope of the release changed • Change Log updated
2022-02-21	3.0.2.0	Seungjae Kim	4.2 4.3	• Scope of the release changed • Change Log updated
2022-03-30	3.0.3.0	Seungjae Kim	4.2 4.3	• Scope of the release changed • Change Log updated
2022-05-02	3.0.4.0	Seungjae Kim	4.2 4.3	• Scope of the release changed • Change Log updated
2022-08-19	3.0.5.0	Jaeho Yang	4.2 4.3	• Scope of the release changed • Change Log updated

2022-09-20	3.0.5.1	Jiwon Oh	4.2 4.3	• Scope of the release changed • Change Log updated
2023-05-26	3.0.6.0	Minuk Kim	4.2 4.3 5.7	• Scope of the release changed • Change Log updated • Add configuration option CanIfTxPduTruncation
2023-07-21	3.1.0.0	Jaeho Yang	4.2 4.3 5.5	• Scope of the release changed • Change Log updated • Add configuration option CanIfTxRxMonitoring
2023-09-08	3.1.1.0	Jaeho Yang	4.2 4.3 4.4.2 5.7	• Scope of the release changed • Change Log updated • Add Deviations item • CanIfTxPduTruncation option content changed
2023-11-30	3.1.2.0	Jaeho Yang	4.2 4.3 7.2	• Scope of the release changed • Change Log updated • Generator Error Message added
2025-01-24	3.2.0.0	Jaeho Yang	4.2 4.3 5.5	• Scope of the release changed • Change Log updated • Add configuration option CanIfOptimizeQueueBufferSize • Modify CanIfAutronTrcvDrvSupport option to CanIfAutoeverTrcvDrvSupport option

Table of Contents

1 Overview

This document is created based on AUTOSAR standard SRS/SWS. For more detailed functional description, please refer to the below reference documents.

The following terms on configuration category mean:

- Changeable (C) : Items that can be configured by user
- Fixed (F) : Items that cannot be changed by user
- NotSupported (N) : Items that are not used

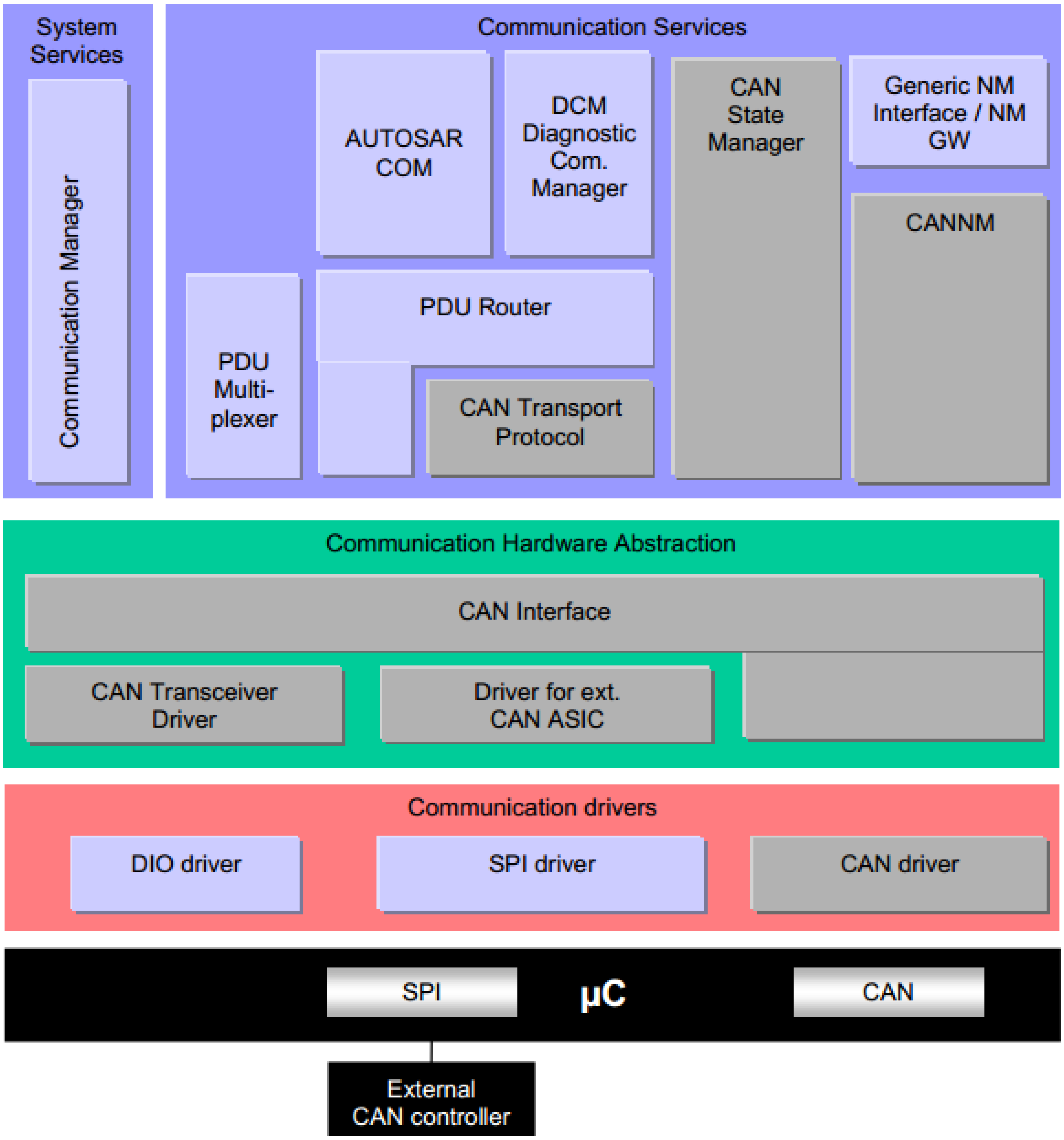
2 Reference

Sl. No.	Title	Version
1	AUTOSAR_SWS_CanInterface.pdf	5.0.0

3 AUTOSAR System

3.1 CanIf Module

CanIf is an interface module for CAN communication.



4 Product Release Notes

4.1 Overview

This chapter provides the release information of the Hyundai AutoEver CanIf module, describing the features and restrictions of different versions of CanIf software product.

4.2 Scope of the Release

All content in this document is limited to the following Hyundai Autoever CanIf modules.

Module name	AUTOSAR version	SWS version	Module version
CanIf	4.0.3	5.0.0	3.1.2

※ Module version refers to the SW version of the BswModule Description (Bswmd) file of each module.

4.3 Change Log

4.3.1 Version 3.2.0.0

- Feature
  - Support for memory optimization when using CanIf Tx Basic Can Buffer

Rationale	When using CanIf Tx Basic Can Buffer, RAM memory usage is high, so It developed a RAM memory optimization function.
Impact on behavior	About the CanIfOptimizeQueueBufferSize setting, When set to True, the Data Field size of the Basic Can Buffer is allocated as the largest CanIfTxPduDlc value among the CanIfTxPdus that use the same Basic Buffer. When set to False, If the CanIfSupportCANFD setting is True, the Data Field size of all Basic Buffer is allocated as 64 bytes, and if it is False, it is allocated as 8 bytes.
Impact on settings	CanIfPublicCfg > CanIfOptimizeQueueBufferSize
Required ASW actions	When set to True, the Buffer Size is allocated as large as the largest CanIfTxPduDlc among the CanIfTxPdus that use the same Basic Buffer, so if a message with a Data Length larger than the set CanIfTxPduDlc is required to be transmitted, the CanIfTxPduDlc value needs to be changed to the required Data Length value.

- Task
  - Fix UM related to trauncation feature

Rationale	Allowable Data Length (CANFD: 64-Byte, HS/LSCAN: 8-Byte) < When the transmission request comes in Data Length When Truncation is True, it is implemented to cut and transmit with the set CanIfTxPduDlc, but it was incorrectly written to cut and transmit with the allowable Data Length, so this has been corrected.
Impact on behavior	None
Impact on settings	CanIfInitCfg > CanIfTxPduCfg > CanIfTxPduTruncation
Required ASW actions	None

- Bug
  - Compile error when using PostBuild

Rationale	PostBuild Case not considered when developing Truncation feature
Impact on behavior	None
Impact on settings	Implementation Config Variant: VARIANT-POST-BUILD-SELECTABLE
Required ASW actions	None

- Improvement
  - Fixed CanCanIfRxPduCanIdMask to always work when comparing the received CanId with the set CanId when receiving a message.

Rationale	Fixed CanIfRxPduCanIdMask to work to compare the received CanId with the set CanId when receiving a message regardless of whether CanIfMetaDataSupport is set or not.
Impact on behavior	None
Impact on settings	CanIfInitCfg/CanIfRxPduCfg/CanIfRxPduCanIdMask
Required ASW actions	None

- Improvement
  - Change AUTRON related settings to AUTOEVER settings

Rationale	Modify AUTRON (old company) related settings to AUTOEVER (current company) settings
Impact on behavior	None
Impact on settings	CanIfPublicCfg/CanIfAutoeverTrcvDrvSupport
Required ASW actions	None

- Bug
  - While Stuck problem occurs when sending message with Dlc 0

Rationale	Logic problem where the Basic Can Buffer is considered empty when Dlc in the Basic Can Buffer is 0
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.2 Version 3.1.2.0

- Improvement
  - Improved to generate a Generation Error when the Rx Pdu mapped to Hrh does not exist or the Tx Pdu mapped to Hth does not exist.

Rationale	If the Rx Pdu mapped to Hrh does not exist, a problem may occur where the message is delivered to the upper layer

	through the wrong path when receiving a message from the CanHardwareObject corresponding to Hrh. This has been improved so that a Generation Error occurs when set like this.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - Improved the problem of messages not being sent when using R44 MCAL CAN and Partial Network (Selective Wakeup)

Rationale	When using R44 MCAL CAN, the problem is not transmitted because the Partial Network Status bit is not enabled.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - When Partial Network used & unused Channels coexist, the problem of Can Message not being sent from the unused Channel has been improved.

Rationale	Improved the issue of checking whether the Partial Network Status bit is Enabled even in the case of Partial Network unused channels.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.3 Version 3.1.1.0

- Improvement
  - Improved Tx Pdu Dlc Check Logic

Rationale	If Tx Pdu's Max Length < transmission request is on Dlc, data is cut by Max Length and transmitted.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - Improvement of non-reception of extended message when using Can Hw Object in Mixed (set hw object capable of receiving both extended type and standard type) method

Rationale	When the extended pdu received from the Hw Object is applied

	in a mixed method, there is a problem of filtering with the standard filter instead of the extended filter. This has been improved (Side effect of 3.1.0.0)
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.4 Version 3.1.0.0

- Bug
  - Improved malfunction of Bus Load Detect function when using multiple CanDrivers

Rationale	Improved malfunction of Bus Load Detect function when using multiple CanDrivers
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Improved Generation Error to occur when CanIfTxPduCfg's CanIfTxPduBufferRef setting is not grouped by cluster

Rationale	If the CanIfTxPduBufferRef setting of CanIfTxPduCfg is not grouped by cluster, the Tx Queue Buffer corresponding to the other controller is cleared during controller control (FullCom/NoCom/BusOff), and messages may not be transmitted. Detected by Generator to prevent this
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - Improved Macro error generation when CANFD, Basic Rx Buffer is used and Extended Rx Message exists

Rationale	If CANFD or Basic Rx Buffer is used and Extended Rx Message exists, a macro is generated incorrectly and the received extended message is not delivered to the upper layer.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement

- Improved so that Det Report Error can be detected normally when using Multiple Can Driver and Det Report Error functions

Rationale	Improved so that Det Report Error can be detected normally when using Multiple Can Driver and Det Report Error functions
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Feature
  - Support monitoring function for transmission and reception of each Pdu

Rationale	Support monitoring function for transmission and reception of each Pdu
Impact on behavior	None
Impact on settings	CanIfPublicCfg/CanIfTxRxMonitoringSupport
Required ASW actions	None

- Bug
  - Improvement of non-reception of extended message when using Can Hw Object in Mixed (set hw object capable of receiving both extended type and standard type) method

Rationale	When the extended pdu received from the Hw Object is applied in a mixed method, there is a problem of filtering with the standard filter instead of the extended filter. This has been improved.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Improved generator execution speed for generating Hrh Range Buffer when using HRH Range

Rationale	Improved generator execution speed for generating Hrh Range Buffer when using HRH Range
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.5 Version 3.0.6.0

- Improvement
  - Modify logic for CanIf_CheckBaudrate, CanIf_ChangeBaudrate API (unsupported).

Rationale	Modify logic for CanIf_CheckBaudrate, CanIf_ChangeBaudrate API (unsupported)
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Arrange value of Pre-compile position of CanIf Template.

Rationale	Arrange value of Pre-compile position of CanIf Template.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Add Tx Pdu dlc (data length) check logic.

Rationale	Add Tx Pdu dlc (data length) check logic.(CanIfTxPduTruncation)
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Generation error if CANFDID16BitSupport setting is true when using Extended & CANFD.

Rationale	Generation error if CANFDID16BitSupport setting is true when using Extended & CANFD.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.6 Version 3.0.5.1

- Bug
  - Unable to Busoff Notification from CanIf module to parent module when R44 MCAL CAN is applied.

Rationale	CANIF_DISPATCH_USERCTRLBUSOFF_NAME() with Busoff Notification due to different types between CanIf R40 module and R44 MCAL CAN module for Controller START during Busoff
Impact on behavior	None
Impact on settings	None

Required ASW actions	None

4.3.7 Version 3.0.5.0

- Improvement
  - Improved Undefined rom/ram data.

Rationale	Improved Undefined rom/ram data (allocated memory section).
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Task
  - Improved security coding to respond for UNECE Cyber Security regulations.

Rationale	Improved security coding to respond for UNECE Cyber Security regulations.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - When Postbuild variant is applied and rx multiple callback is used, abnormal behavior occurs when receiving rx msg.

Rationale	Improved the problem that ucCallbackIndex item is not created when Post Build Variant is applied and RX_MULTIPLE_CALLBACK is set to STD_ON.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.8 Version 3.0.4.0

- Bug
  - Fixed an issue where some Basic CAN messages were not processed as upper layer due to an index mapping error between HrH – RxLpdu.

Rationale	An error occurred that some Basic CAN messages were not processed as upper layer due to an index mapping error between HrH – RxLpdu.
Impact on behavior	None
Impact on settings	None

Required ASW actions	None

4.3.9 Version 3.0.3.0

- Bug
  - Fix compile error when PduR buffer is set.

Rationale	Due to security coding, the naming of the referenced part when setting the PduR buffer is changed and a compile error occurs..
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Modified to sort Generator's Input File List.

Rationale	The input list of the generated file may change every generation, so it needs to be corrected so that it can be sorted.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.10 Version 3.0.2.0

- Bug
  - Fixed a compile error that occurs due to a function creation issue in the Generated files when PN and MultipleTrcvDriver were both used.

Rationale	An issue where wakeup-related functions were not created in Generated files when PN and MultipleTrcvDriver were both used.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - Fixed an issue where the callback setting defaults to CDD_RXINDICATION despite user input when the upper layer of a CanTp message is set to CDD.

Rationale	When a CanTp message is created, the name of callback function was fixed to CDD_RxIndication even when user input is provided.
Impact on behavior	None
Impact on settings	None
Required ASW	None

actions	

- Improvement
  - Fixed an issue that prevented message reception because of the mistakenly imported size value of the Rx Basic message (LpdusList) when using PostBuild.

Rationale	An issue the size value of the Rx Basic message (LpdusList) was mistakenly imported and prevented the message from being found when using PostBuild.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Fixes related to MISRA

Rationale	A violation of MISRA
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.11 Version 3.0.1.0

- Task
  - Code improvements to comply with the UNECE Cyber Security regulations.

Rationale	Needed to improve the codes to comply with the UNECE Cyber Security regulations.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.12 Version 3.0.0.0

- New feature
  - PostBuild functionality support

Rationale	PostBuild functionality support
Impact on behavior	None
Impact on settings	Can ID should be modified and Apply Variant should be set (refer to 9.2. settings).
Required ASW actions	Code should be added to EcuM_Callout so that the variant setting can be done before initiating EcuM.

- Bug
  - Fixed an error that resulted in a version mismatch warning in Candrv in Generate.

Rationale	Invalid string comparison resulted in a warning during Generate
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - Syntax error fixed in ID comparison during a binary search (when metadata support is set to ON and Rx message is created as Basic).

Rationale	During comparing IDs, an incorrect bit operation with the mask value may result in selecting an incorrect ID value.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.13 Version 2.7.5.0

- Improvement
  - Fixes to support mode switching when using Mcal CAN 4.4.0

Rationale	Modification was required due to the ControllerMode value change between AR403 and AR440.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Fixed the default value of CanIfPublicCanDrvVersion.

Rationale	The default value was not set to AR_403.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.14 Version 2.7.4.0

- Bug
  - Fixed the DLC error check routine of CAN FD.

Rationale	The DLC error check routine didn't distinguish between general and FD messages.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - Fixed the invalid schema location.

Rationale	An invalid schema had been reflected to PDF.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Fixes that enables the application of Can_IdType when using CAN Extended options.

Rationale	In some cases, when Can_GeneralTypes.h is included, Can_IdType was not set to uint16.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Fixed the variable definition of CanIf_GaaHrhInit.

Rationale	Added the omitted pre-compile options to the variable definition of CanIf_GaaHrhInit.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Fixes related to MISRA

Rationale	MISRA rule violations
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.15 Version 2.7.3.0

- Improvement
  - Fixed CanIfCtrlDrvTxCancellation settings (False -> Automated), added validations.

Rationale	Modification of Cancellation setting guide
Impact on behavior	None
Impact on settings	Hardware Cancellation functionality can’t be set in RH850.
Required ASW actions	None

- Improvement
  - CanIfPublicTxConfirmPollingSupport settings changed (Fixed -> Not Supported)

Rationale	UM settings modification
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Changed the default value of CanIfPublicVersionInfoApi (true -> false)

Rationale	Needed to change the default value of CanIf PDF.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - Added extern declarations.

Rationale	Missing extern declarations for CanIf_TriggerTransmit.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.16 Version 2.7.2.0

- Improvement
  - Module MISRA-C Verification

Rationale	Violation of MISRA-C
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.17 Version 2.7.1.0

- Improvement
  - Module MISRA-C Verification

Rationale	Violation of MISRA-C
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.18 Version 2.7.0.0

- New feature
  - S32G LIce Multi-Driver support

Rationale	Fixes to support the LIce feature.
Impact on behavior	None
Impact on settings	PublicCanDrvVersion/CANFDID16BitSupport should be set.
Required ASW actions	None

4.3.19 Version 2.6.2.0

- Improvement
  - Fixed the Multiplicity value of CanIfDispatchUserConfirmPnAvailabilityUL

Rationale	Changed due to the modification of AUTOSAR specification (1 -> 0..1)
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Improved an issue where BusLoad was not measured when DET was OFF.

Rationale	Fix required to eliminate DET influence in BusLoad measurement
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - Fixed an issue where invalid references to HrH were made.

Rationale	Flawed exception handling in the HrH check logic.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.20 Version 2.6.1.1

- Bug
  - Documentation changes due to the errors in the updates of UM and DeliveryBoxHistory documents

Rationale	Documentation changes due to the errors in the updates of UM and DeliveryBoxHistory documents
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.21 Version 2.6.1.0

- Improvement
  - CAN-FD maximum data size fixed to 64 bytes due to changes in ES95480-02 specifications.

Rationale	The maximum size of transferrable data was changed from 32 bytes to 64.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Bug
  - Fixed the CanIf_TriggerTransmit symbol error when DET was set to off.

Rationale	CanIf_Triggertransmit API was implemented in a way that it was only usable when DET was ON, even if it doesn’t have anything to do with the DET functionality.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.22 Version 2.6.0.0

- Improvement
  - Structural and documentation changes to publish the source code

Rationale	Structural and documentation changes to publish the source code
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.23 Version 2.6.0

- New feature
  - Developed an RX/TX message counting API to measure the bus load of each channel.

Rationale	Developed an RX/TX message counting API to measure the bus load of each channel.
Impact on behavior	None
Impact on settings	Added CanIf 〉 CanIfPublicCfg 〉 CanIfBusloadDetectingSupport settings
Required ASW actions	None

- Bug
  - CanIfPrivateDlcCheck attribute changed

Rationale	Fixed an error where the CanIfPrivateDlcCheck attribute was Changeable in the code but was set to Fixed in the actual arxml definition.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.24 Version 2.5.7

- Improvement
  - Fixed so that the Can Controller Cancellation On/Off behavior is not influenced by libraries.

Rationale	The Can Controller Cancellation On/Off behavior should be fixed so that it is not influenced by libraries.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.25 Version 2.5.6

- Improvement
  - Fixed so that the DLC On/Off behavior is not influenced by libraries.

Rationale	The DLC On/Off behavior should be fixed so that it is not influenced by libraries.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- New feature
  - Partly modified CanIf module API to support MCAL CAN of AUTOSAR 4.2.2

Rationale	Partly modified CanIf module API due to a request for MCAL CAN support under AUTOSAR 4.2.2
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.26 Version 2.5.5

- Improvement
  - Fixed the Category attribute of the CanIfRxPduDlc setting (FIXED → CHANGEABLE)

Rationale	Category attribute modified to enable the user to change Rx Pdu Dlc on the Odin setting screen.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.27 Version 2.5.4

- Improvement
  - Added User Manual content for external CAN transceiver development to the CanIf User Manual

Rationale	The CanTrcv module is not deployed in some project that use an external CAN transceiver. Therefore it is required to add the guide for development of external CAN transceiver, included in the User Manual of CanTrcv module, to the CanIf User Manual.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- New feature
  - Development work for IdsM module support

Rationale	Additional requirements related to the IdsM module
Impact on behavior	Reports information about CAN Controller and messages received to the IdsM module, when: - the IdsM module is integrated and related settings are enabled, or - an individual message has its IdsM reporting setting enabled
Impact on settings	CanIf -> CanIfInitCfg -> CanIfRxPduCfg -> CanIfPduReportIdsMEnable item added
Required ASW actions	None

4.3.28 Version 2.5.3

- Improvement
  - Fixed the logic saving messages that have been cancelled by CanIf_CancelTxConfirmation after MCAL CAN cancels it

Rationale	When a request is made to send a BASIC CAN message after CAN is disconnected, the message is cancelled and then saved by CanIf_CancelTxConfirmation. The outdated messages that are stored may cause problems when the connection is restored as they are sent after new messages.

Impact on behavior	Cancelled messages are not stored in the queue.
Impact on settings	None
Required ASW actions	None

- Notation fixed for CanIfPublicCfg/CanIfPublicVersionInfoApi in the User Manual

Rationale	CanIfPublicVersionInfo was mistakenly written as CanIfPublicVersion.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.29 Version 2.5.2

- Improvement
  - A DET error occurring in CanIf when a CAN-FD message is received.

Rationale	As AUTOSAR 4.1.2 FD-related specifications have been applied, the bit masking area was modified (0x4000 → 0x40000000) in Can Id of Renesas MCAL accordingly.
Impact on behavior	None
Impact on settings	Settings added for McalCANFD32BitSupport and McalCANFDDlcDiscreteSupport.
Required ASW actions	None

4.3.30 Version 2.5.1

- Improvement
  - Corrected compile warnings

Rationale	Warnings occur while building the module
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Modified Can_IdType and the Tx Buffer size due to the addition of CAN-FD functionalities

Rationale	Modified the Can_IdType range and the Tx Buffer size due to the addition of CAN-FD functionalities
Impact on behavior	Should consider expanding memory size to accommodate the changed Tx Buffer size when applying CAN-FD functionalities
Impact on settings	None
Required ASW actions	None

- CanIfBufferSize attribute changed

Rationale	PDF modified to allow the user to change the value of CanIfBufferSize
Impact on behavior	None
Impact on settings	None
Required ASW actions	Can change the size value of CanIfBuffer, if necessary, after CanIf harmonization

4.3.31 Version 2.5.0

- New feature
  - Modification to support a new CanTrv module developed by a partner

Rationale	To support a new CanTrv module developed by a partner
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Fixed the versions of module source code

Rationale	To match the version of the generator with that of the source code by fixing the source versions
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.3.32 Version 1.5.9

- New feature
  - N/A

- Improvement
  - Improve the generative logic of CanIf that refers to the upper modules for deciding the routing path.

Rationale	An error occurred during the creation of a callback function while establishing a PDU connection between modules, because a PDU going from CDD to PduR was mistaken as going from CanIf to PduR when CanIf tries to check the upper layer.
Impact on behavior	None
Impact on settings	None

Required ASW actions	None

4.3.33 Version 1.5.8

- New feature
  - N/A

- Improvement
  - Fixed an overflow error in constant value referring to the size of transfer queue created by CanIf generator.

Rationale	An overflow occurs when the queue size exceeds 255 as the queue was typecast as uint8.
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

- Improvement
  - Fixed DLC length to enable Tx/Rx settings for CAN-FD data (Max: 8 → 64 bytes).

Rationale	It wasn't possible to set the length of CAN-FD data to 64 bytes (the maximum length of Tx/Rx in CanIf was 8 byte).
Impact on behavior	None
Impact on settings	None
Required ASW actions	None

4.4 Module Release Notes

4.4.1 Limitations

- CAN Internal Wake-up Support not supported
  The following items are all unsupported and should be set to false.
  CanIfPublicWakeupCheckValidByNM
  CanIfPublicWakeupCheckValidSupport
  CanIfCtrlWakeupSupport

- CAN Controller ID
  The CanIfCtrlId value should be set to the same value as CanControllerId of the connected MCAL CAN controller. (In case of S32G MCU)

- CAN Transceiver ID
  The CanIfTrcvId value should be set to the same value as CanTrcvChannelId of the connected Can Transceiver module.

- Transmit cancellation unsupported
  (Unsupported) the feature cancels the transmission of a message in the current transmit

buffer.

- Change baud rate unsupported
  (Unsupported) the feature changes the baud rate of CAN communication in runtime.

- Software Filter Type
  Among various filters that selects valid messages among many received ones when CAN messages are received via Basic CAN method, table method and index method filters are not supported.

- Dynamic Tx Id unsupported
  (Unsupported) the feature changes in runtime the ID of a CAN message transmitted.

- TTCAN unsupported
  Time-triggered CAN is unsupported.

- Multiple CAN Driver Support (only for S32G LIce)
  the feature enables the support of multiple CAN MCAL drivers.
  It is only available when the Mcal versions of all MCAL CAN drivers are identical.
  CanIfCANFDID16BitSupport setting should be set to the same value among the drivers.
  Callback functions per individual driver are not supported.

- Wake-up Validation By Nm unsupported
  (Unsupported) the feature takes into account whether an Nm message is received when determining the validity of a wake-up.

- Polling for Tx Confirmation state unsupported
  (Unsupported) the feature enables the polling of Tx confirmation state.

4.4.2 Deviations

- CanCM Support
  An interface is supported for the CanCM module to call when transmission should be limited based on voltage.

- IdsM Support
  Reports information about the reception of messages and the status of CAN Controller to the IdsM module.

- CanIfTxPduTruncation
  In Autosar 4.2.2 or later specifications, if a dlc larger than the Max Length (EcuC PduLength) of the set Tx Pdu is requested for transmission, a function is written to request transmission by cutting it by the Max Length of the configured Tx Pdu.
  In CanIf, the above Autosar specifications are not satisfied and
  If a transmission request is made for a dlc larger than the Max Length (FD: 64 bytes, HS: 8 bytes) allowed in the CanIdType of the configured Tx Pdu, a function is provided to cut the transmission request by the configured PduLength (CanIfTxPduDlc)

5 Configuration Guide

The CanIf configuration for the AUTOSAR platform distributed by Hyundai Autoever reflects the policies of Hyundai Autoever and therefore any changes require consultation with Hyundai Autoever.

5.1 CanIfCtrlDrvCfg configuration

Parameter Name	Value	Category
[1)]CanIfCtrlDrvTxCancellation	Automated	C
CanIfCtrlDrvInitHohConfigRef	Automated	F
CanIfCtrlDrvNameRef	Automated	F

1) As for the CAN Controller cancellation settings, it must be checked whether the hardware supports them (ask the MCU vendor).
   When Basic CAN and CAN Controller Cancellation settings are both used, messages may not be transmitted properly in some cases. (Due to the characteristics of RH850 (F1x), the CAN controller cancellation feature only works with the Basic CAN; however, Hyundai Autoever policies prohibit the setting of CAN Controller cancellation with Basic CAN because there's a possibility of dropping messages.)

5.2 CanIfCtrlCfg configuration

Parameter Name	Value	Category
CanIfCtrlId	Automated	F
[1)]CanIfCtrlWakeupSupport	False	N
CanIfCtrlCanCtrlRef	Automated	F

1) For each channel, the CanIfCtrlId value should be set to the same value as CanControllerId of the connected MCAL CAN controller.

5.3 CanIfDispatchCfg configuration

Parameter Name	Value	Category
CanIfDispatchUserCheckTrcvWakeFlagIndicationUL	CAN_SM	C
[1)]CanIfDispatchUserCheckTrcvWakeFlagIndicationName	-	C
CanIfDispatchUserClearTrcvWufFlagIndicationUL	CAN_SM	C
[1)]CanIfDispatchUserClearTrcvWufFlagIndicationName	-	C
CanIfDispatchUserConfirmPnAvailabilityUL	CAN_SM	C
[1)]CanIfDispatchUserConfirmPnAvailabilityName	-	C
CanIfDispatchUserCtrlBusOffUL	CAN_SM	C
[1)]CanIfDispatchUserCtrlBusOffName	-	C
CanIfDispatchUserCtrlModeIndicationUL	CAN_SM	C
[1)]CanIfDispatchUserCtrlModeIndicationName	-	C
CanIfDispatchUserTrcvModeIndicationUL	CAN_SM	C

Parameter Name	Value	Category
[1]CanIfDispatchUserTrcvModeIndicationName	-	C
CanIfDispatchUserValidateWakeupEventUL	ECUM	C
[1]CanIfDispatchUserValidateWakeupEventName	-	C

1) Enter the name of the function to be called in case the user wants to develop a separate CDD to receive Dispatch events instead of using CanSM or EcuM modules. After processing a Dispatch event, the CDD should call the indication function of a upper module (CanSM or EcuM) to pass the event. The default values at the time of deployment should not be changed unless the user develops and implements a CDD.

5.4 CanIfPrivateCfg configuration

Parameter Name	Value	Category
[1]CanIfPrivateDlcCheck	True	C
[2]CanIfPrivateSoftwareFilterType	BINARY	F
CanIfSupportTTCAN	False	N
[3]CanIfSupportCanCM	False	C
[4]CanIfSupportCANFD	False	C

1) Set whether DLC check is performed on the received CAN frames (CanIfPublicDevErrorDetect function should be used)
2) Set the kind of the filter when the BASIC CAN method is used to receive CAN frames.
3) Set to true if the CanCM module is used to control TX based on the battery voltage.
4) Set to true if it is required to receive and transmit CAN-FD frames

5.5 CanIfPublicCfg configuration

Parameter Name	Value	Category
CanIfPublicCancelTransmitSupport	False	N
[1]CanIfPublicCddHeaderFile	-	C
CanIfPublicChangeBaudrateSupport	False	N
[8]CanIfPublicDevErrorDetect	True	F
CanIfPublicHandleTypeEnum	UINT16	F
CanIfPublicMultipleDrvSupport	False	N
CanIfPublicNumberOfCanHwUnits	1	F
CanIfPublicPnSupport	False	C
[3]CanIfPublicReadRxPduDataApi	False	F
[4]CanIfPublicReadRxPduNotifyStatusApi	False	F
[5]CanIfPublicReadTxPduNotifyStatusApi	False	F
CanIfPublicSetDynamicTxIdApi	False	F
[6]CanIfPublicTxBuffering	True	F
CanIfPublicTxConfirmPollingSupport	False	N
CanIfPublicVersionInfoApi	False	F
CanIfPublicWakeupCheckValidByNM	False	N
[2]CanIfPublicWakeupCheckValidSupport	False	N
[7]CanIfMetaDataSupport	False	C
CanIfAutoeverTrcvDrvSupport	True	C
CanIfExternalTrcvDrvSupport	False	C
[9]CanIfCANFDID16BitSupport	False	C
[10]CanIfCANFDDiscreteDlcSupport	False	C
[11]CanIfBusLoadDetectingSupport	False	C

Parameter Name	Value	Category
[12]CanIfPublicCanDrvVersion	AR_403	C
[13]CanIfTxRxMonitoringSupport	False	C
[14]CanIfOptimizeQueueBufferSize	False	C

1) If the user develops a CDD that directly communicates with the CanIf module, the CDD should provide an indication function to CanIf, and therefore a header file with the declaration of that function should be added.
2) Set whether the CAN should perform a validity check on a wakeup.
3) Set in case the user-developed CDD directly reads the CAN frame data received by the CanIf module.
4) Provides an API that informs the user-developed CDD whether the CanIf module received CAN frames.
5) Provides an API that informs the user-developed CDD whether the CanIf module transmitted CAN frames.
6) Used when the number of CAN hardware buffers available for TX is smaller than that of messages to be transmitted.
7) Set whether to use metadata when using a J1939 CAN communication method.
8) Switches DET functionality on and off. Set to true by default; when optimization is required due to the use of gateway functionality, may be set to false.
9) Set by the platform distributor; should be set after checking whether the platform's MCAL uses the 31st bit as the CAN-FD mask bit in CAN-FD processing (true if MCAL supports 16 bit; false otherwise).
   Should be set to false when CanIfPublicCanDrvVersion is set to AR_440.
10) Set by the platform distributor; should be set after checking whether the platform's MCAL does the DLC processing in advance for CAN-FD.
   (False if the MCAL does DLC proc'essing because CanIF doesn't have to; True if the MCAL doesn't because CanIF should.)
11) Set by the platform distributor; should be set after checking whether the bus load detecting functionality is used.
   (ComMBusloadDetectingApi of the ComM module should be set in order to enable bus load detecting.)
12) Set to AR_440 when using the version 4.4.0 or later of MCAL CAN (should be set to AR_440 to enable S32G LIce feature)
13) Settings to support the monitoring function to see if each Pdu is transmitting and receiving normally
14) Settings to support memory optimization when using Basic Tx Buffer
   If checked as True, the Data Field size of the Basic Can Tx Buffer is allocated as the largest CanIfTxPduDlc value among the CanIfTxPdus that use the same Basic Tx Buffer.
   If checked as False, if the CanIfSupportCANFD setting is True, the Data Field size of all Basic Tx Buffer is allocated as 64 bytes, and if it is False, it is allocated as 8 bytes.
   ※ When using this function, the Buffer Size is allocated as large as the largest CanIfTxPduDlc among the CanIfTxPdus that use the same Basic Buffer. Therefore, if a message with a Data Length larger than the set CanIfTxPduDlc is required to be transmitted, the CanIfTxPduDlc value must be changed to the required Data Length value.

5.6 CanIfRxPduCfg configuration

Parameter Name	Value	Category
CanIfRxPduCanId	Automated	F
CanIfRxPduCanIdType	Automated	F
CanIfRxPduDlc	Automated	C
CanIfRxPduId	Automated	F
[1]CanIfRxPduReadData	False	F

Parameter Name	Value	Category
[2]CanIfRxPduReadNotifyStatus	False	F
[3]CanIfRxPduUserRxIndicationUL	Automated	C
[4]CanIfRxPduUserRxIndicationName	Automated	C
CanIfRxPduHrhIdRef	Automated	F
CanIfRxPduRef	Automated	F
CanIfRxPduCanIdMask	Automated	F
[5]CanIfPduReportIdsMEnable	Automated	C

1) Whether to use the API that reads PDU data
2) Whether to use the API that check the PDU reception status
3) Designate an upper module to which the PDU reception notice is passed.
   This is automatically set when importing CANDB; if the user develops a CDD that needs to receive PDUs, can be set to the CDD.
4) Designate a function provided by the CDD.
5) Determines whether the message reception information is delivered to the IdsM module.

5.7 CanIfTxPduCfg configuration

Parameter Name	Value	Category
CanIfTxPduCanId	Automated	F
CanIfTxPduCanIdType	Automated	F
CanIfTxPduDlc	Automated	F
CanIfTxPduId	Automated	F
CanIfTxPduReadNotifyStatus	False	C
CanIfTxPduType	Automated	F
[1]CanIfTxPduUserTxConfirmationUL	-	C
[2]CanIfTxPduUserTxConfirmationName	-	C
CanIfTxPduBufferRef	Automated	F
CanIfTxPduRef	Automated	F
CanIfTxPduPnFilterPdu	False	N
CanIfTxPduCanIdMask	0	N
[3]CanIfTxPduTruncation	-	C

1) Designate an upper module to which the PDU transmission complete notice is passed.
   This is automatically set when importing CANDB; if the user develops a CDD that needs to transmit PDUs, can be set to the CDD.
2) Designate a function provided by the CDD.
3) Check whether to use truncation for the pdu.
   For pdus transmitted exceeding the maximum dlc,
   If checked as true, CanIf shall the Det error, transmits only the set amount of Pdu Length (CanIfTxPduDlc) and discards the rest.
   If checked as false, CanIf shall report the Det error and does not transmit.
   ※ Maximum dlc: 64 bytes if CanIfTxPduType is Fd Can Id Type, otherwise 8 bytes

5.8 CanIfBufferCfg configuration

Parameter Name	Value	Category
CanIfBufferSize	Automated	C

- The user may change the value as necessary after running in an auto-configuration mode.

5.9 CanIfTrcvDrvCfg 설정

Parameter Name	Value	Category
CanIfTrcvId	-	C
CanIfTrcvWakeupSupport	False	N
CanIfTrcvCanTrcvRef	-	C

6 Application Programming Interface (API)

6.1 Type Definitions

None

6.2 Macro Constants

None

6.3 Functions

6.3.1 PDU Channel Mode Control

Function Name	CanIf_SetPduMode	
Syntax	FUNC(Std_ReturnType, CANIF_CODE) CanIf_SetPduMode(uint8 ControllerId, CanIf_PduSetModeType PduModeRequest)	
Service ID	0x09	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	ControllerId	All PDUs of the own ECU connected to the corresponding CanIf ControllerId, which is assigned to a physical CAN controller are addressed
	PduModeRequest	Requested PDU mode change
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType E_OK: Request for mode transition has been accepted.	

	E_NOT_OK: Request for mode transition has not been accepted.
Description	This service sets the requested mode at the L-PDUs of a predefined logical PDU channel.
Preconditions	CAN Interface module should be initialized
Configuration Dependency	None

Function Name	CanIf_GetPduMode	
Syntax	FUNC(Std_ReturnType, CANIF_CODE)CanIf_GetPduMode (uint8 ControllerId, P2VAR(CanIf_PduGetModeType, AUTOMATIC, CANIF_APPL_DATA) PduModePtr)	
Service ID	0x0a	
Sync/Async	Synchronous	
Reentrancy	Reentrant (Not for the same channel)	
Parameters (In)	ControllerId	All PDUs of the own ECU connected to the corresponding CanIf ControllerId, which is assigned to a physical CAN controller are addressed
Parameters (Inout)	None	
Parameters (Out)	PduModePtr	Pointer to a memory loc here the current mode of the logical PDU channel will be stored.
Return Value	Std_ReturnType E_OK: PDU mode request has been accepted E_NOT_OK: PDU mode request has not been accepted	
Description	This service reports the current mode of a requested PDU channel.	
Preconditions	CAN Interface module should be initialized	

Configuration Dependency	None

6.3.2 Transceiver Mode Control

Function Name	CanIf_SetTrcvMode	
Syntax	FUNC(Std_ReturnType, CANIF_CODE) CanIf_SetTrcvMode (uint8 TransceiverId, CanTrcv_TrcvModeType TransceiverMode)	
Service ID	0x0d	
Sync/Async	Asynchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	TransceiverId	Abstracted CanIf TransceiverId, which is assigned to a CAN transceiver, which is requested for mode transition
	TransceiverMode	Requested mode transition
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType E_OK: Transceiver mode request has been accepted. E_NOT_OK: Transceiver mode request has not been accepted	
Description	This service changes the operation mode of the tansceiver TransceiverId, via calling the corresponding CAN Transceiver Driver service.	
Preconditions	CAN Interface module should be initialized	
Configuration Dependency	At least one Transceiver should be configured	

Function Name	CanIf_GetTrcvMode

Syntax	FUNC(Std_ReturnType, CANIF_CODE) CanIf_GetTrcvMode (P2VAR(CanTrcv_TrcvModeType, AUTOMATIC, CANIF_APPL_DATA)TransceiverModePtr, uint8 TransceiverId)	
Service ID	0x0e	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	TransceiverId	Abstracted CanIf TransceiverId, which is assigned to a CAN transceiver, which is requested for current operation mode.
Parameters (Inout)	None	
Parameters (Out)	TransceiverModePtr	Requested mode transition.
Return Value	Std_ReturnType E_OK: Transceiver mode request has been accepted. E_NOT_OK: Transceiver mode request has not been accepted	
Description	This function invokes CanTrcv_GetOpMode and updates the parameter TransceiverModePtr with the value OpMode provided by CanTrcv.	
Preconditions	CAN Interface module should be initialized	
Configuration Dependency	At least one Transceiver should be configured	

Function Name	CanIf_SetTrcvWakeupMode
Syntax	FUNC(Std_ReturnType, CANIF_CODE) CanIf_SetTrcvWakeupMode (uint8 TransceiverId, CanTrcv_TrcvWakeupModeType TrcvWakeupMode)
Service ID	0x10
Sync/Async	Synchronous

Reentrancy	Non-Reentrant	
Parameters (In)	TransceiverId	Abstracted CanIf TransceiverId, which is assigned to a CAN transceiver, which is requested for current operation mode.
	TrcvWakeupMode	Requested transceiver wakeup notification mode
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType E_OK: Will be returned, if the wakeup notifications state has been changed to the requested mode. E_NOT_OK: Will be returned, if the wakeup notifications state change has failed or the parameter is out of the allowed range. The previous state has not been changed.	
Description	This function shall call CanTrcv_SetTrcvWakeupMode.	
Preconditions	CAN Interface module should be initialized	
Configuration Dependency	At least one Transceiver should be configured	

6.3.3 Transmit Processing

Function Name	CanIf_Transmit
Syntax	FUNC(Std_ReturnType, CANIF_CODE) CanIf_Transmit (PduIdType CanTxPduId, P2CONST(PduInfoType, AUTOMATIC, CANIF_APPL_CONST) PduInfoPtr)
Service ID	0x05
Sync/Async	Synchronous
Reentrancy	Reentrant

Parameters (In)	CanTxPduId	L-PDU handle of CAN L-PDU to be transmitted. This handle specifies the corresponding CAN L-PDU ID and implicitly the CAN Driver instance as well as the corresponding CAN controller device
	PduInfoPtr	Pointer to a structure with CAN L-PDU related data: DLC and pointer to CAN L-SDU buffer
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType E_OK: Will be returned, if the check wakeup request has been accepted E_NOT_OK: Will be returned, if the check wakeup request has not been accepted	
Description	This service initiates a request for transmission of the CAN L-PDU specified by the CanTxPduId and CAN related data in the L-PDU structure.	
Preconditions	CAN Interface module should be initialized	
Configuration Dependency	At least one TxPdu should be configured.	

6.3.4 RX Indication

Function Name	CanIf_ReadRxPduData
Syntax	FUNC(Std_ReturnType, CANIF_CODE)CanIf_ReadRxPduData (PduIdType CanRxPduId, P2VAR(PduInfoType, AUTOMATIC, CANIF_APPL_DATA) PduInfoPtr)
Service ID	0x06
Sync/Async	Synchronous

Reentrancy	Non-Reentrant	
Parameters (In)	CanRxPduId	Receive L-PDU handle of CAN L-PDU. This handle specifies the corresponding CAN L-PDU ID and implicitly the CAN Driver instance as well as the corresponding CAN controller device.
Parameters (Inout)	None	
Parameters (Out)	PduInfoPtr	Standard/Extended CAN ID of CAN L-PDU that has been successfully received.
Return Value	Std_ReturnType	E_OK: Request for L-PDU data has been accepted E_NOT_OK: No valid data has been received
Description	This service provides the CAN DLC and the received data of the requested CanRxPduId to the calling upper layer.	
Preconditions	CAN Interface module should be initialized	
Configuration Dependency	At least one RxPdu should be configured and CANIF_READRXPDU_DATA_API should be configured as STD_ON.	

Function Name	CanIf_ReadTxNotifStatus	
Syntax	FUNC (CanIf_NotifStatusType, CANIF_CODE) CanIf_ReadTxNotifStatus (PduIdType CanTxPduId)	
Service ID	0x07	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	CanTxPduId	L-PDU handle of CAN L-PDU to be transmitted. This handle specifies the corresponding CAN L-PDU ID and implicitly the CAN Driver

		instance as well as the CAN controller device.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	CanIf_NotifStatusType	Current confirmation status of the corresponding CAN Tx L-PDU.
Description	This service returns the confirmation status (confirmation occurred of not) of a specific static or dynamic CAN Tx L-PDU, requested by the CanTxPduId.	
Preconditions	CAN Interface module should be initialized	
Configuration Dependency	CANIF_READTXPDU_NOTIFY_STATUS_API should be configured as STD_ON.	

Function Name	CanIf_ReadRxNotifStatus	
Syntax	FUNC(CanIf_NotifStatusType, CANIF_CODE) CanIf_ReadRxNotifStatus (PduIdType CanRxPduId)	
Service ID	0x08	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	CanRxPduId	L-PDU handle of CAN L-PDU to be received. This handle specifies the corresponding CAN L-PDU ID and implicitly the CAN Driver instance as well as the CAN controller device.
Parameters (Inout)	None	
Parameters (Out)	None	

Return Value	CanIf_NotifStatusType	Current confirmation status of the corresponding CAN Rx L-PDU.
Description	This service returns the indication status (indication occurred or not) of a specific CAN Rx L-PDU, requested by the CanRxPduId.	
Preconditions	CAN Interface module should be initialized	
Configuration Dependency	CANIF_READRXPDU_NOTIFY_STATUS_API should be configured as STD_ON.	

6.3.5 Get Status

Function Name	CanIf_GetTxConfirmationState	
Syntax	FUNC(CanIf_NotifStatusType,CANIF_CODE)CanIf_GetTxConfirmationState (VAR(uint8, CANIF_VAR) ControllerId)	
Service ID	0x19	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (In)	CanControllerId	Can controller for which the status shall be changed - based on configuration order list
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	CanIf_NotifStatusType	Current confirmation status of the corresponding CAN Rx L-PDU.
Description	This service reports, if any TX confirmation has been done for the whole CAN controller since the last CAN controller start.	
Preconditions	CAN Interface module should be initialized	

Configuration Dependency	CANIF_TXCONFIRM_POLLING_SUPPORT should be configured as STD_ON.

6.3.6 Notes

None

7 Generator

7.1 Generator Option

Options	Description
-H/-Help	To display help regarding usage of the tool.
-O/-Output	To generate the output files in the specified directory location.
-V/-Version	To display the copyright information and the tool version.
-L/-Log	To generate "$BswConfig::Lis_File_Name" file.

7.2 Generator Error Message

ERR060059: 'CanIfRxPduCanId' configured in the container 'CanIfRxPduCfg' should be within the Rx Pdu Range configured in container 'CanIfRxPduCanIdRange'.

This Error message is displayed, 'CanIfRxPduCanId' configured in the container 'CanIfRxPduCfg' is not within the Rx Pdu Range configured in container 'CanIfRxPduCanIdRange'

ERR060021: Multiplicity of the container 'CanIfCtrlDrvCfg' should be same as Parameter 'CanIfPublicNumberOfCanHwUnits' of 'CanIfPublicCfg'.

This Error message is displayed, if (Multiplicity of the container 'CanIfCtrlDrvCfg' != parameter 'CanIfPublicNumberOfCanHwUnits' of 'CanIfPublicCfg' container)

ERR060060: Lower HRH Range configured in the parameter 'parameter name' should be less than upper HRH Range configured in the parameter 'parameter name' of container 'container name'.

This Error message is displayed, if (Lower Rx Pdu Range configured in the parameter 'CanIfRxPduCanIdRangeLowerCanId' should be less than upper Rx pdu Range configured in the parameter 'CanIfRxPduCanIdRangeUpperCanId' of container ' CanIfRxPduCanIdRange')
and
if (Lower HRH Range configured in the parameter 'CanIfHRHRangeRxPduLowerCanId' should be less than upper HRH Range configured in the parameter 'CanIfHRHRangeRxPduUpperCanId' of container ' CanIfRxPduCanIdRange')

ERR060058: At least one Tx PDU or Rx PDU should be configured.

This Error message is displayed, if (At least one Tx PDU or Rx PDU is not configured)

ERR060051: The reference path <Reference Path> provided for the parameter 'CanIfRxPduRef' in the container 'CanIfRxPduConfig' is not found in <CAN_NM/CAN_TP/J1939TP/PDUR/CDD>.

This Error message is displayed, if (the reference path configured for the parameter 'CanIfRxPduRef' in the container 'CanIfRxPduCfg' is not found in CAN_NM/CAN_TP/J1939TP/PDUR/CDD) AND
if (the reference path configured for the parameter 'CanIfTxPduRef' in the container 'CanIfTxPduCfg' is not found in CAN_NM/CAN_TP/J1939TP/PDUR/CDD)

ERR050020: UThe value <Value> of the structure element 'Structure Element Name' in structure 'Structure Name' is not within the range. The value <Value> should be within the range of <Min Value> - <Max Value>, as its data type is <Type>.

This Error message is displayed, if ((Value of the structure element < Min value) || (Value of the structure element > Max value))

ERR060064: Value(s) <IDs> is(are) not configured for the parameter 'Parameter Name' in 'Dependent Container Name' of the container 'Container Name'.

This Error message is displayed, if (the values of the CanIfRxPduId parameters are not sequential in their respective containers. Values of CanIfRxPduId parameters should start from 0 and should be sequential)

ERR060055: Value 'Parameter' of the reference parameter 'Parameter Name' is repeated within the container 'Container Name' in the configuration set <Configset Name>.

This Error message is displayed, if (value of the refernce parameter 'CanIfHthIdSymRef' is repeated within the container 'CanIfHthCfg' within the configuration set.)
Or
If (value of the reference parameter 'CanIfHRHIdSymRef' is repeated within the container 'CanIfHRHCfg' within the configuration set.)

ERR060052: Lower CAN ID <ID> should be lesser than the higher CAN ID <ID> in the HRH <HRH ID>.

This Error message is displayed, if (value of the parameter CanIfHRHRangeRxPduLowerCanId >= value of the parameter CanIfHRHRangeRxPduUpperCanId of the container CanIfHRHRangeCfg)

ERR060065: Valid values of the parameter 'Parameter Name' of the container 'Container Name' <PDU name> are <1 – 2047/536870911> for <STANDARD/EXTENDED> type.

This Error message is displayed, if((CanIfRxPduCanId < 1 or > 2047, when CanIfRxPduCanIdType == "STANDARD")OR
(CanIfRxPduCanId < 1 or > 2415919103, when CanIfRxPduCanIdType == "EXTENDED")OR
(CanIfTxPduCanId < 1 or > 2047, when CanIfTxPduCanIdType == "STANDARD")OR
(CanIfTxPduCanId is < 1 or > 2415919103, when CanIfTxPduCanIdType == "EXTENDED"))

ERR060062: 'Parameter Name' <ID> is repeated in the <Dependent Container Name> of the container 'Container Name'.

This Error message is displayed, if (the values of the following parameters are not unique in their respective containers)

Parameter name	Container Name	Dependent Container Name
CanIfTxPduCanId	CanIfTxPduCfg	‘Value configured for the parameter CanIfInitCfgSet’
CanIfCtrlCanCtrlRef	CanIfCtrlCfg	CAN Driver ‘Value configured for the container CanIfCtrlDrvCfg’
CanIfRxPduCanId	CanIfRxPduCfg	‘Value configured for the parameter CanIfInitCfgSet’

ERR060061: Value of the parameter 'Parameter Name' <ID> is repeated in the container 'container name'.

This Error message is displayed if (the value of the following parameter is configured for the following parameters are repeated in their respective containers)

Parameter Name	Container Name
CanIfCtrlId	CanIfCtrlCfg
CanIfTrcvId	CanIfTrcvCfg

ERR060063: Value(s) <0> is (are) not configured for the parameter 'Parameter Name' in the container 'Container Name'.

This Error message is displayed if (value of the parameter ‘CanIfCtrlId’ does not start from 0 and is not sequential across all the Drivers)

ERR060056: Reference to CAN Hardware object <Reference Path> is repeated across HRH and HTH in the configuration set <CanIfInitCfg0>.

This Error message is displayed if (Reference to CAN Hardware object <Reference Path> is repeated across HRH and HTH in the configuration set)

ERR060053: Atleast one HRH or HTH should be configured within the controller <controller ID> of the CAN Driver <Driver name>.

This Error message is displayed if (Controller is not configured with HRH or HTH)

ERR060054: Value <Reference Path> configured for the parameter 'CanIfCanTrcvChannelRef' of the container CanTrcvChannel is repeated

This Error message is displayed if (the reference path configured for the parameter ‘CanIfTrcvCanTrcvRef‘ is not unique across all the Channels of the container ‘CanTrcvChannel‘)

ERR050026: The value configured for the element ‘Structure Element Name’ in structure ‘Structure Name’ should follow C syntax <[a-zA-Z][a-zA-Z0-9₩_]>.

This Error message is displayed if (((Value of the structure element !~ /^[a-zA-Z][a-zA-Z0-9₩_]*$/)) && (Value of the structure element!~ /^₩s*$/))
ERR050068: The reference path provided for the parameter 'Parameter Name' in the container ‘Container Name’ is not found in <CDD>.

This Error message is displayed if (The reference path provided for the following parameter is not found in CDD)

Parameter Name	Container Name
CanIfTxPduRef	CanIfTxPduCfg
CanIfRxPduRef	CanIfRxPduCfg

ERR060001: Unexpected Error Found. Please contact AUTOEVER Support Team.

This Error message is displayed if (Incorrect Configuration in input file(s))

ERR060002: Unexpected Error Found. This error may be due to the incorrect configuration of the element(s) 'Element Name'. If the error is not resolved, then please contact AUTOEVER Support Team.

This Error message is displayed if (incorrect Configuration of below parameters and conatiners in input file)

Parameter Name	Container Name	Valid Value
CanIfPrivateDlcCheck	CanIfPrivateCfg	true / false
CanIfPublicDevErrorDetect	CanIfPublicCfg	true / false
CanIfPublicReadRxPduDataApi	CanIfPublicCfg	true / false
CanIfPublicReadRxPduNotifyStatusApi	CanIfPublicCfg	true / false
CanIfRxPduReadData	CanIfRxPduCfg	true / false
CanIfRxPduReadNotifyStatus	CanIfRxPduCfg	true / false
CanIfTxPduReadNotifyStatus	CanIfTxPduCfg	true / false
CanIfCtrlDrvTxCancellation	CanIfCtrlDrvCfg	true / false
CanIfHRHSoftwareFilter	CanIfHRHConfig	true / false
CanIfPublicReadTxPduNotifyStatusApi	CanIfPublicCfg	true / false
CanIfPublicSetDynamicTxIdApi	CanIfPublicCfg	true / false
CanIfPublicVersionInfoApi	CanIfPublicCfg	true / false
CanIfPublicMultipleDrvSupport	CanIfPublicCfg	true / false

Table 2: List of parameters

ERR060003: CanIf Component is not present in the input file(s).

This Error message is displayed if (Module CanIf or CAN Driver component are not present in input file(s))

ERR060004: The reference path is empty for the parameter 'Parameter Name' in the container 'Container Name', having short name 'Container Short Name'.

This Error message is displayed if (no reference path is provided for any of the below mentioned parameters)

Parameter Name	Container Name
CanIfCtrlCanCtrlRef	CanIfCtrlCfg
CanIfCtrlDrvNameRef	CanIfCtrlDrvCfg

CanIfCtrlDrvInitHohConfigRef	CanIfCtrlDrvCfg
CanIfHRHCanCtrlIdRef	CanIfHRHCfg
CanIfHRHIdSymRef	CanIfHRHCfg
CanIfHthCanCtrlIdRef	CanIfHthCfg
CanIfHthIdSymRef	CanIfHthCfg
CanIfRxPduHRHIdRef	CanIfRxPduCfg
CanIfTxPduHthIdRef	CanIfTxPduCfg
CanIfTrcvId	CanIfTrcvCfg

ERR060013: The incorrect reference path for the parameter 'Parameter Name' in the container 'Container Name', has short name 'Container Short Name'.

This Error message is displayed if (Incorrect reference path is configured for the parameter: CanIfHRHIdSymRef of CanIfHRHCfg container, parameter: CanIfHthIdSymRef of CanIfHthCfg container, parameter: CanIfRxPduHRHIdRef of CanIfRxPduCfg container , parameter: CanIfTxPduHthIdRef of CanIfTxPduCfg container)

ERR060006: The value configured for the parameter 'AR-RELEASE-VERSION' in the container 'BSW-IMPLEMENTATION' should follow the pattern: <4.[0-9]+.[0-9]+>.

This Error message is displayed if (AR-RELEASE-VERSION in BSW-IMPLEMENTATION does not follow the pattern: <4.[0-9]+.[0-9]+>)

ERR060057: PDU's lower Can id should be greater than HRH's lower Can id and PDU's lower Can id should be less than HRH's upper Can id. PDU's upper Can id should be greater HRH's Lower CanId and PDU's upper Can id should be less than HRH's Upper Can id.

This Error message is displayed if (the value configured for parameter CanIfHRHRangeRxPduLowerCanId > CanIfRxPduCanIdRangeLowerCanId && CanIfRxPduCanIdRangeLowerCanId > CanIfHRHRangeRxPduUpperCanId and CanIfHRHRangeRxPduUpperCanId < CanIfRxPduCanIdRangeUpperCanId && CanIfRxPduCanIdRangeUpperCanId < CanIfHRHRangeRxPduLowerCanId)

ERR060066: Number of HTHs configured should be within the range of <UINT8>/<UINT16>. Number of container 'Container Name' configured = <Container count>. Value of the parameter 'Parameter Name' = <Value>.

This Error message is displayed if (parameter 'CanIfPublicHandleTypeEnum' configured as UINT8 & number of HTHs configured are more than 255 Or if parameter 'CanIfPublicHandleTypeEnum' configured as UINT16 & number of HTHs configured are more than 65535)

ERR060067: Number of HRHs configured should be within the range of <UINT8>/<UINT16>. Number of container 'Container Name' configured = <Container count>. Value of the parameter 'Parameter Name' = <Value>.

This Error message is displayed if (parameter 'CanIfPublicHandleTypeEnum' configured as UINT8 & number of HRHs configured are more than 255 Or if parameter 'CanIfPublicHandleTypeEnum' configured as UINT16 & number of HRHs configured are more than 65535)

ERR060069: Value of the parameter 'CanIfRxPduUserRxIndicationName' should be conifgured when value of the parameter 'CanIfRxPduUserRxIndicationUL' is configured as <CDD> in the container 'CanIfRxPduCfg'.

This Error message is displayed if (parameter 'CanIfRxPduUserRxIndicationName' not configured when 'CanIfRxPduUserRxIndicationUL' is configured as <CDD> in the container 'CanIfRxPduCfg').

ERR060070: Value of the parameter 'CanIfRxPduCanId' must be configured since CAN RxPdu configured as 'FULL' in CAN driver.

This Error message is displayed if (Value of the parameter 'CanIfRxPduCanId' not configured when CAN RxPdu configured as 'FULL' in CAN driver).

ERR060071: HRH Range should be configured as range for Rx Pdu is configured.

This Error message is displayed if (HRH Range is not configured when range for Rx Pdu is configured).Appendix
None

ERR060080: In case of Renesas, parameter 'CanIfCtrlDrvTxCancellation' in the container 'CanIfCtrlDrvCfg'can not be configured as <true/1>, please refer to CanIf UserManual.

This Error message is displayed if (In case of Renesas, CanIfCtrlDrvTxCancellation can not be configured as 'true'.)

ERR060087: HTH 'Hth name'connected to CAN Driver 'Can Drv name' is not connected to any Tx PDU.
This Error message is displayed if (CanIfTxPduCfg that refers to Hth does not exist.)

ERR060088: HRH 'Hrh name'connected to CAN Driver 'Can Drv name' is not connected to any Rx PDU.
This Error message is displayed if (CanIfRxPduCfg that refers to Hrh does not exist.)

8 Det Error

Detected development errors shall be reported to the Det_ReportError(uint16 ModuleId, uint8 InstanceId, uint8 ApiId, uint8 ErrorId) service of the Development Error Tracer (DET) if the pre-processor switch CANIF_DEV_ERROR_DETECT is set “on”.

8.1 Error Classification

Type of error	Relevance	Related error code	Value
Invalid CAN ID is reported	Development	CANIF_E_PARAM_CANID	0x0A
Invalid DLC is reported	Development	CANIF_E_PARAM_DLC	0x0B
Invalid HRH is reported	Development	CANIF_E_PARAM_HRH	0x0C
Invalid LPDU	Development	CANIF_E_PARAM_LPDU	0x0D
Invalid controller is reported	Development	CANIF_E_PARAM_CONTROLLER	0x0E
Invalid controller ID is reported	Development	CANIF_E_PARAM_CONTROLLERID	0x0F
Invalid wakeup source ID is reported	Development	CANIF_E_PARAM_WAKEUPSOURCE	0x10
Invalid mode request is reported	Development	CANIF_E_PARAM_TRCV	0x11
Invalid pointer input is reported	Development	CANIF_E_PARAM_TRCVMODE	0x12
This Error code is used when TrcvWakeupMode is out of range	Development	CANIF_E_PARAM_TRCVWAKEUPMODE	0x13
DET error used with parameter passed as null pointer	Development	CANIF_E_PARAM_POINTER	0x14
This Error code is used when controller Mode is invalid or out of defined values or CANIF_CS_UNINIT	Development	CANIF_E_PARAM_CTRLMODE	0x15
Invalid PDU Mode	Development	CANIF_E_PARAM_PDU_MODE	0x16
DET error reported when module is unintialized	Development	CANIF_E_UNINIT	0x1E
This Error code is used TxPduId is out of range	Development	CANIF_E_INVALID_TXPDUID	0x32
This Error code is used RxPduId is out of range	Development	CANIF_E_INVALID_RXPDUID	0x3C
Invalid DLC value	Development	CANIF_E_INVALID_DLC	0x3D
This Error code is used when Pdu Mode is Offline or controller mode is in stopped	Development	CANIF_E_STOPPED	0x46
This Error code is used when controller mode is in not in sleep	Development	CANIF_E_NOT_SLEEP	0x47

8.1.1 Service ID

CanIf function name	Service ID[hex]
CanIf_Init	0x01
CanIf_SetController	0x03
CanIf_GetControllerMode	0x04
CanIf_Transmit	0x05
CanIf_ReadRxPduData	0x06
CanIf_ReadTxNotifStatus	0x07
CanIf_ReadRxNotifStatus	0x08
CanIf_SetPduMode	0x09
CanIf_GetPduMode	0x0A
CanIf_VersionInfo	0x0B
CanIf_SetDynamicTxId	0x0C
CanIf_SetTrcvMode	0x0D
CanIf_GetTrcvMode	0x0E
CanIf_GetTrcvWakeupReason	0x0F
CanIf_SetTransceiverWakeupMode	0x10
CanIf_CheckWakeup	0x11
CanIf_CheckValidation	0x12
CanIf_TxConfirmation	0x13
CanIf_RxIndication	0x14
CanIf_CancelTxConfirmation	0x15
CanIf_ControllerBusOff	0x16
CanIf_ControllerModeIndication	0x17
CanIf_CancelTransmit	0x18
CanIf_GetTxConfirmationState	0x19
CanIf_ConfirmPnAvailability	0x1A
CanIf_ChangeBaudrate	0x1B
CanIf_CheckBaudrate	0x1C
CanIf_ClearTrcvWufFlag	0x1E
CanIf_CheckTrcvWakeFlag	0x1F
CanIf_ClearTrcvWufFlagIndication	0x20
CanIf_CheckTrcvWakeFlagIndication	0x21
CanIf_TrcvModeIndication	0x22

9 Appendix

9.1 Development of a CANTRCV module

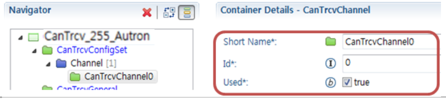
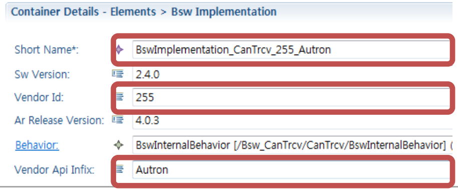
For details about building a CANTRCV module, refer to AUTOSAR 4.0.3 CANTransceiverDriver Spec.

When creating a CANTRCV module, file, or API, it must be named following the recommended naming rule.

- Use 255(0xFF) as VendorID

- Naming Rule : CanTrcv_VendorId_VendorSpecifiName

9.1.1 Files need to be created anew

Ecud_CanTrcv_255_ VendorspecificName.arxm	Module definition PDF: recommend using the prebuilt PDF provided within AUTOSAR  Navigator CanTrcv_255_Autron CanTrcvConfigSet Channel [1] CanTrcvChannel0 CanTrcvGeneral Container Details - CanTrcvChannel Short Name*: CanTrcvChannel0 Id*: 0 Used*: ☑ true	Ecud_CanTrcv_255_Autron.arxml (see deploy file)
Bswmd_CanTrcv_255_ VendorspecficName.arxml	 Container Details - Elements > Bsw Implementation Short Name*: BswImplementation_CanTrcv_255_Autron Sw Version: 2.4.0 Vendor Id: 255 Ar Release Version: 4.0.3 Behavior: BswInternalBehavior [/Bsw_CanTrcv/CanTrcv/BswInternalBehavior] (Vendor Api Infix: Autron	Bswmd_CanTrcv_255_Autron.arxml (see deploy file)
CanTrcv_255_ VendorspecificName.h		CanTrcv_255_Autron.h (see deploy file)
CanTrcv_ VenderID_VendorspecificName. c		CanTrcv_255_Autron.c (see deploy file)

9.1.2 Mandatory API

1) When not using CAN Transceiver wakeup

CanTrcv_255_VendorspecificName_SetOpMode	Changes module mode
CanTrcv_255_VendorspecificName_GetOpMode	Checks module mode information
CanTrcv_255_VendorspecificName_Init	Resets the module

2) When using CAN Transceiver wakeup

CanTrcv_255_VendorspecificName_SetOpMode	Changes module mode
CanTrcv_255_VendorspecificName_GetOpMode	Checks module mode information
CanTrcv_255_VendorspecificName_Init	Resets the module

CanTrcv_255_VendorspecificName_GetBusWuReason	Gets the wakeup reason for the Transceiver and returns it in parameter Reason
CanTrcv_255_VendorspecificName_SetWakeupMode	Enables, disables or clears wake-up events of the Transceiver according to TrcvWakeupMode.
CanTrcv_255_VendorspecificName_CheckWakeup	Service is called by underlying CANIF in case a wake up interrupt is detected
CanTrcv_255_VendorspecificName_ClearTrcvWufFlag	Clears the WUF flag in the transceiver hardware. This API shall exist only if CanTrcvHwPnSupport = TRUE.
CanTrcv_255_VendorspecificName_CheckWakeFlag	Requests to check the status of the wakeup flag from the transceiver hardware.

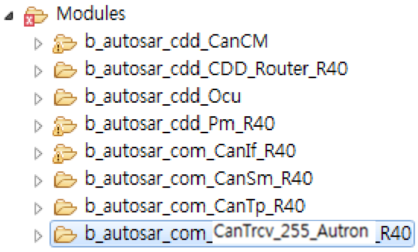
9.1.3 How to integrate CANTRCV

9.1.3.1 Add the CANTRCV module (as source code or a library) and a build configuration

9.1.3.1.1 AUTOEVER CANTRCV module + a new CANTRCV module

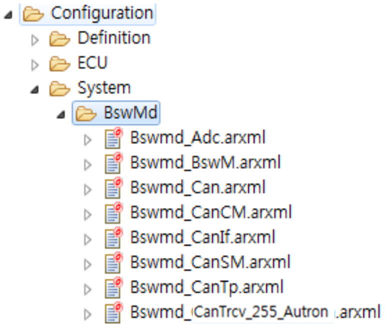
1. Copy the new CANTRCV module (as source code or a library) into the folder where the Bsw module is located.

   (The location of the module can be different.)



```
Modules
  b_autosar_cdd_CanCM
  b_autosar_cdd_CDD_Router_R40
  b_autosar_cdd_Ocu
  b_autosar_cdd_Pm_R40
  b_autosar_com_CanIf_R40
  b_autosar_com_CanSm_R40
  b_autosar_com_CanTp_R40
  b_autosar_com_CanTrcv_255_Autron_R40
```

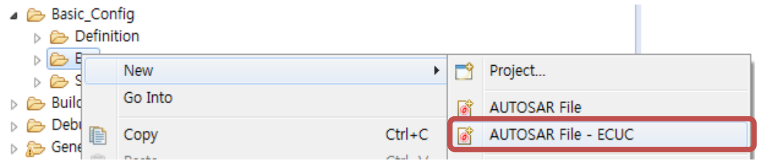
2. Copy the BswMd files of the new CANTRCV module: ECU > System > BswMd

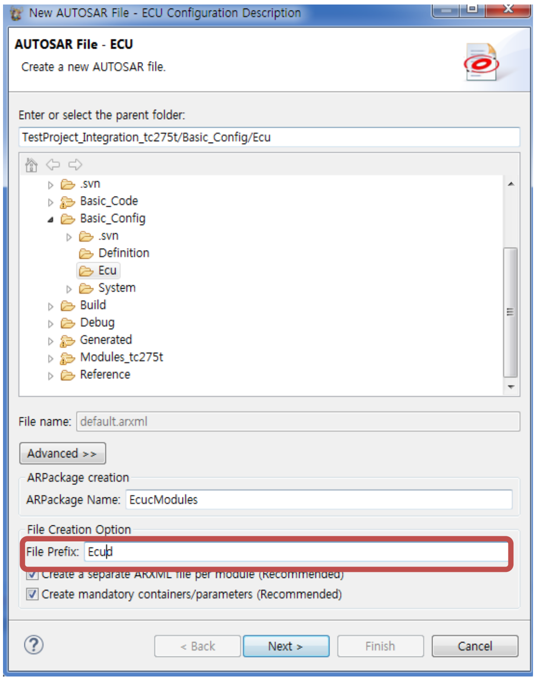


```
Configuration
  Definition
  ECU
  System
    BswMd
      Bswmd_Adc.arxml
      Bswmd_BswM.arxml
      Bswmd_Can.arxml
      Bswmd_CanCM.arxml
      Bswmd_CanIf.arxml
      Bswmd_CanSM.arxml
      Bswmd_CanTp.arxml
      Bswmd_CanTrcv_255_Autron .arxml
```

3. Project > New AUTOSAR File - EcuC > select CanTrcv under Module Selection > select

ECUConfigurationParameters.arxml





New AUTOSAR File - ECU Configuration Description

AUTOSAR File - ECU

Create a new AUTOSAR file.

Enter or select the parent folder: TestProject_Integration_tc275t/Basic_Config/Ecu

File name: default.arxml

ARPackage creation

ARPackage Name: EcucModules

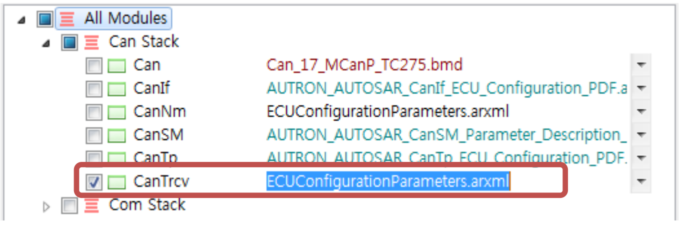
File Creation Option

File Prefix: Ecud

☑ Create a separate ARXML file per module (Recommended)

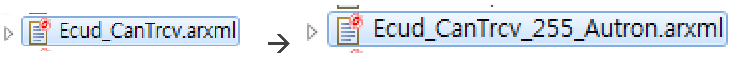
☑ Create mandatory containers/parameters (Recommended)

Change the File Prefix: Ecucd → Ecud



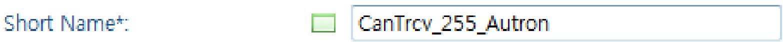
All Modules	
Can Stack	
□ Can	Can_17_MCanP_TC275.bmd
□ CanIf	AUTRON_AUTOSAR_CanIf_ECU_Configuration_PDF.a
□ CanNm	ECUConfigurationParameters.arxml
□ CanSM	AUTRON_AUTOSAR_CanSM_Parameter_Description_
□ CanTp	AUTRON_AUTOSAR_CanTp_ECU_Configuration_PDF.
☑ CanTrcv	ECUConfigurationParameters.arxml
Com Stack	

4. Rename > Ecud_CanTrcv_ 255_VendorspecificName



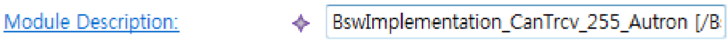
Ecud_CanTrcv.arxml → Ecud_CanTrcv_255_Autron.arxml

5. Change the module's Short Name > CanTrcv_ 255_VendorspecificName



Short Name*: CanTrcv_255_Autron

6. Select Module Description

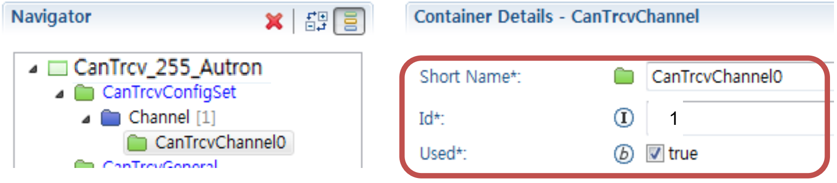


Module Description: BswImplementation_CanTrcv_255_Autron [/B

7. Create a new channel at CanTrcv > Channel

The following three items must be present in Ecud_CanTrcv_255_Autoever.arxml

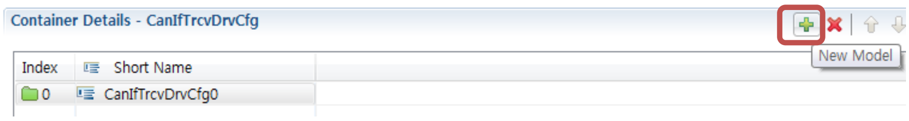
- Short Name: appropriate name
- Id: use the number that follows the CanTrcvChannelId of the AUTOEVER CANTRCV module.
- (e.g. if the last number of CanTrcvChannelId in AUTOEVER CANTRCV is 0, the Ecud_CanTrcv_255_Autoever module can be 1 or above.)
- Used: True



Container Details - CanTrcvChannel

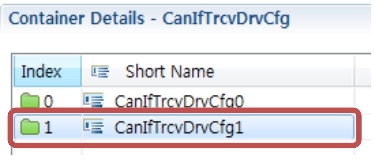
Short Name*: CanTrcvChannel0
Id*: 1
Used*: ☑ true

8. Click the plus button at CanIf > Trcv Drv Cfg to create a new CanIfTrcvDrvCfg



Container Details - CanIfTrcvDrvCfg

Index	Short Name
0	CanIfTrcvDrvCfg0



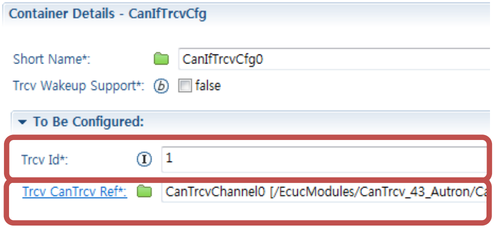
Container Details - CanIfTrcvDrvCfg

Index	Short Name
0	CanIfTrcvDrvCfg0
1	CanIfTrcvDrvCfg1

9. CanTrcvChannelId should be set to the number following CanTrcvChannelId used in the AUTOEVER CANTRCV module.

   (The example is using one transceiver for AUTOEVER CANTRCV.)

   (Other settings except Id are not used by other BSW modules.)



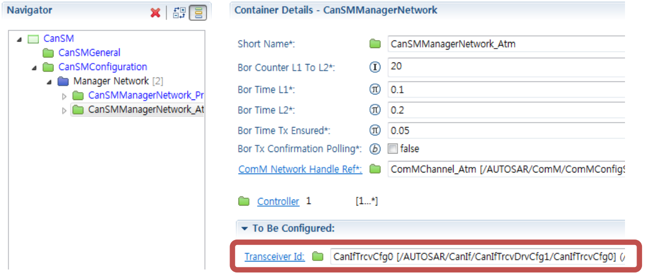
Container Details - CanIfTrcvCfg

Short Name*: CanIfTrcvCfg0
Trcv Wakeup Support*: □ false

To Be Configured:

Trcv Id*: 1
Trcv CanTrcv Ref*: CanTrcvChannel0 [/EcucModules/CanTrcv_43_Autron/Ca

10. CanSM > CanSMConfiguration > set up a new transceiver



Container Details - CanSMManagerNetwork

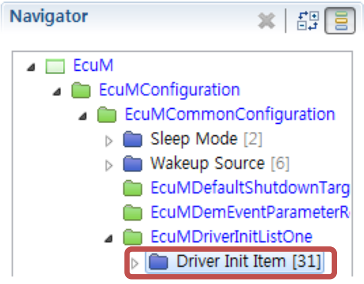
Short Name*: CanSMManagerNetwork_Atm
Bor Counter L1 To L2*: 20
Bor Time L1*: 0.1
Bor Time L2*: 0.2
Bor Time Tx Ensured*: 0.05
Bor Tx Confirmation Polling*: □ false
ComM Network Handle Ref*: ComMChannel_Atm [/AUTOSAR/ComM/ComMConfig

Controller 1 [1...*]

To Be Configured:

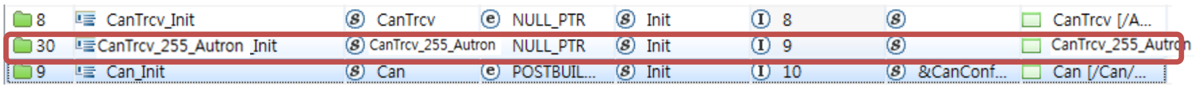
Transceiver Id: CanIfTrcvCfg0 [/AUTOSAR/CanIf/CanIfTrcvDrvCfg1/CanIfTrcvCfg0] (/

11. Add the Init function of the new CanTrcv module at EcuM 〉 EcuMConfiguration 〉 EcuMCommonConfiguration 〉 EcuMDriverInitListOne (should be one right after the existing CanTrcv)



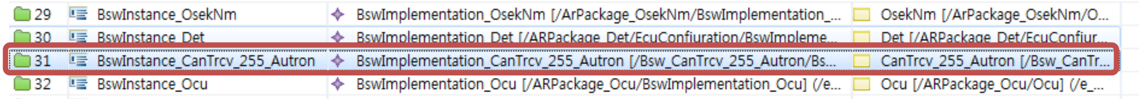
Navigator

- EcuM
  - EcuMConfiguration
    - EcuMCommonConfiguration
      - Sleep Mode [2]
      - Wakeup Source [6]
      - EcuMDefaultShutdownTarg
      - EcuMDemEventParameterR
      - EcuMDriverInitListOne
        - Driver Init Item [31]



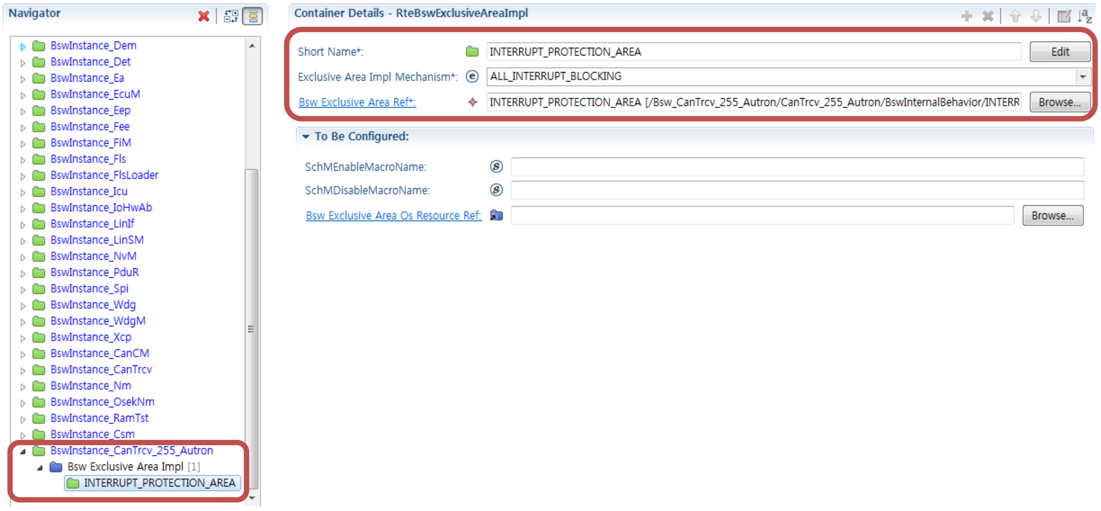
8	CanTrcv_Init	CanTrcv	NULL_PTR	Init	8		CanTrcv [/A...
30	CanTrcv_255_Autron _Init	CanTrcv_255_Autron	NULL_PTR	Init	9		CanTrcv_255_Autron
9	Can_Init	Can	POSTBUIL...	Init	10	&CanConf...	Can [/Can/...

12. Add the CanTrcv module at Rte 〉 Bsw Module Instance



29	BswInstance_OsekNm	BswImplementation_OsekNm [/ArPackage_OsekNm/BswImplementation_...	OsekNm [/ArPackage_OsekNm/O...
30	BswInstance_Det	BswImplementation_Det [/ARPackage_Det/EcuConfiuration/BswImpleme...	Det [/ARPackage_Det/EcuConfiur...
31	BswInstance_CanTrcv_255_Autron	BswImplementation_CanTrcv_255_Autron [/Bsw_CanTrcv_255_Autron/Bs...	CanTrcv_255_Autron [/Bsw_CanTr...
32	BswInstance_Ocu	BswImplementation_Ocu [/ARPackage_Ocu/BswImplementation_Ocu] (/e...	Ocu [/ARPackage_Ocu/Ocu] (/e_...

13. Add a Bsw Exclusive Area at Rte 〉 Bsw Module Instance



Navigator

- BswInstance_Dem
- BswInstance_Det
- BswInstance_Ea
- BswInstance_EcuM
- BswInstance_Eep
- BswInstance_Fee
- BswInstance_FiM
- BswInstance_Fls
- BswInstance_FlsLoader
- BswInstance_Icu
- BswInstance_IoHwAb
- BswInstance_LinIf
- BswInstance_LinSM
- BswInstance_NvM
- BswInstance_PduR
- BswInstance_Spi
- BswInstance_Wdg
- BswInstance_WdgM
- BswInstance_Xcp
- BswInstance_CanCM
- BswInstance_CanTrcv
- BswInstance_Nm
- BswInstance_OsekNm
- BswInstance_RamTst
- BswInstance_Csm
- BswInstance_CanTrcv_255_Autron
  - Bsw Exclusive Area Impl [1]
    - INTERRUPT_PROTECTION_AREA

Container Details - RteBswExclusiveAreaImpl

Short Name*: INTERRUPT_PROTECTION_AREA [Edit]

Exclusive Area Impl Mechanism*: ALL_INTERRUPT_BLOCKING

Bsw Exclusive Area Ref*: INTERRUPT_PROTECTION_AREA [/Bsw_CanTrcv_255_Autron/CanTrcv_255_Autron/BswInternalBehavior/INTERR [Browse...]

To Be Configured:

SchMEnableMacroName:

SchMDisableMacroName:

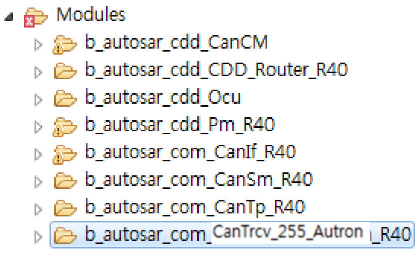
Bsw Exclusive Area Os Resource Ref: [Browse...]

14. Edit Generate.py: Add the new Ecud_CanTrcv_ 255_VendorspecificName and Bswmd_CanTrcv_ 255_VendorspecificName (if necessary) to GenerateCanIf, GenerateCanSm, GenerateEcuM, and GenerateRte.

9.1.3.1.2 Using only the new CANTRCV module

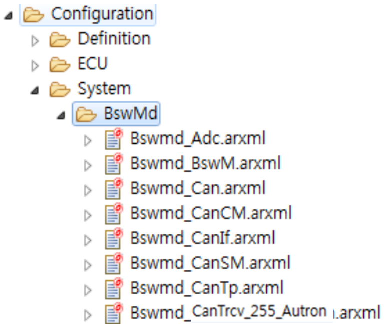
1. Copy the new CANTRCV module to the folder containing the Bsw module.

   (The location of the module can be changed as necessary.)



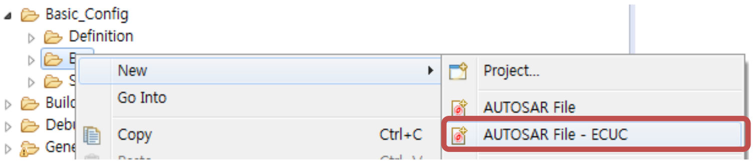
Modules
b_autosar_cdd_CanCM
b_autosar_cdd_CDD_Router_R40
b_autosar_cdd_Ocu
b_autosar_cdd_Pm_R40
b_autosar_com_CanIf_R40
b_autosar_com_CanSm_R40
b_autosar_com_CanTp_R40
b_autosar_com_CanTrcv_255_Autron _R40

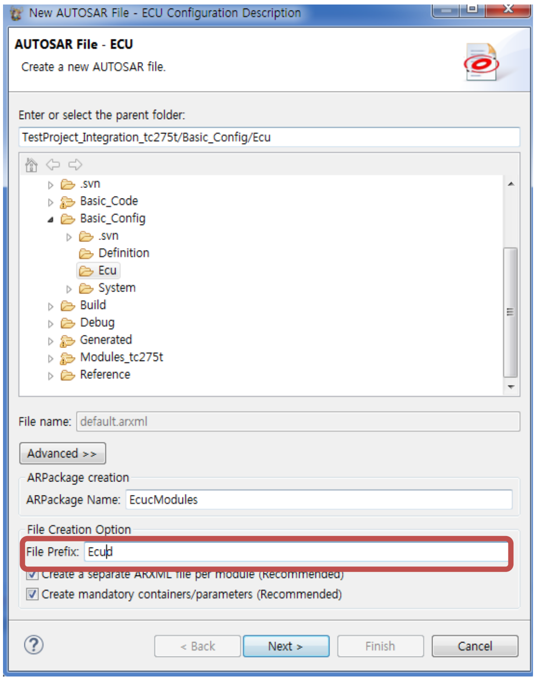
2. Copy the BswMd files of the new CANTRCV module: ECU > System > BswMd



Configuration
Definition
ECU
System
BswMd
Bswmd_Adc.arxml
Bswmd_BswM.arxml
Bswmd_Can.arxml
Bswmd_CanCM.arxml
Bswmd_CanIf.arxml
Bswmd_CanSM.arxml
Bswmd_CanTp.arxml
Bswmd_CanTrcv_255_Autron .arxml

3. Project > New AUTOSAR File - EcuC > select CanTrcv under Module Selection > select ECUConfigurationParameters.arxml





New AUTOSAR File - ECU Configuration Description

AUTOSAR File - ECU
Create a new AUTOSAR file.

Enter or select the parent folder:
TestProject_Integration_tc275t/Basic_Config/Ecu

.svn
Basic_Code
Basic_Config
.svn
Definition
Ecu
System
Build
Debug
Generated
Modules_tc275t
Reference

File name: default.arxml

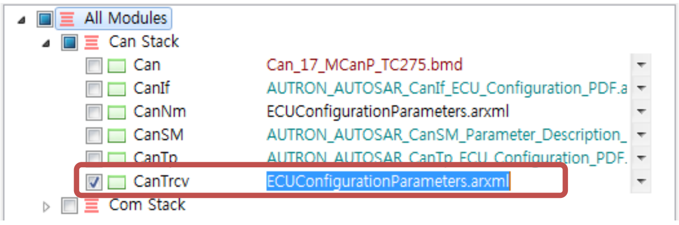
Advanced >>

ARPackage creation
ARPackage Name: EcucModules

File Creation Option
File Prefix: Ecud
☑ Create a separate ARXML file per module (Recommended)
☑ Create mandatory containers/parameters (Recommended)

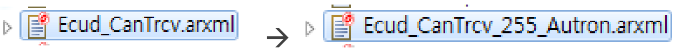
< Back | Next > | Finish | Cancel

Change the File Prefix: Ecucd → Ecud



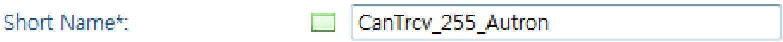
- All Modules
  - Can Stack
    - □ Can — Can_17_MCanP_TC275.bmd
    - □ CanIf — AUTRON_AUTOSAR_CanIf_ECU_Configuration_PDF.a
    - □ CanNm — ECUConfigurationParameters.arxml
    - □ CanSM — AUTRON_AUTOSAR_CanSM_Parameter_Description_
    - □ CanTp — AUTRON_AUTOSAR_CanTp_ECU_Configuration_PDF.
    - ☑ CanTrcv — ECUConfigurationParameters.arxml
  - Com Stack

4. Rename > Ecud_CanTrcv_ VenderID_VendorspecificName



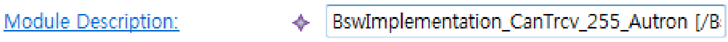
Ecud_CanTrcv.arxml → Ecud_CanTrcv_255_Autron.arxml

5. Change the module's Short Name > CanTrcv_ VenderID_VendorspecificName



Short Name*: CanTrcv_255_Autron

6. Select Module Description

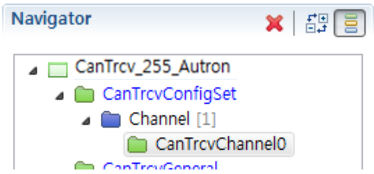


Module Description: BswImplementation_CanTrcv_255_Autron [/B

7. Create a new channel at CanTrcv > Channel

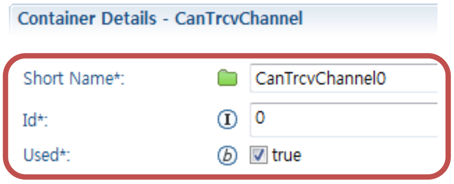
The following three items must be present in Ecud_CanTrcv_255_Autoever.arxml

- Short Name: appropriate name
- Id: set incrementally, starting from 0
- Used: True



Navigator

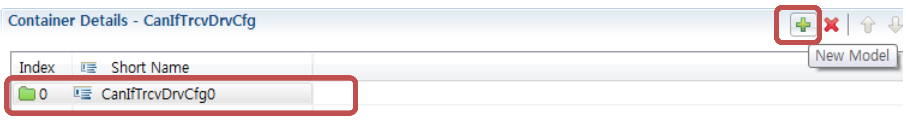
- CanTrcv_255_Autron
  - CanTrcvConfigSet
    - Channel [1]
      - CanTrcvChannel0
  - CanTrcvGeneral



Container Details - CanTrcvChannel

Short Name*: CanTrcvChannel0
Id*: 0
Used*: ☑ true

8. Click the plus button at CanIf > Trcv Drv Cfg to create a new CanIfTrcvDrvCfg

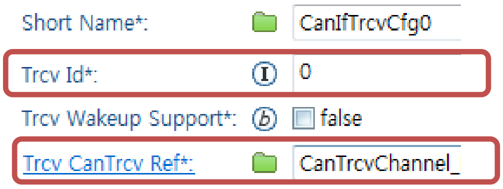


Container Details - CanIfTrcvDrvCfg

Index	Short Name
0	CanIfTrcvDrvCfg0

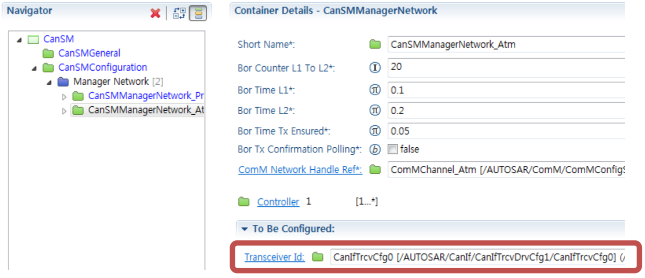
9. Trcv ID should be set incrementally, starting from 0

(Other settings except Id are not used by other BSW modules.)



Short Name*: CanIfTrcvCfg0
Trcv Id*: 0
Trcv Wakeup Support*: □ false
Trcv CanTrcv Ref*: CanTrcvChannel_

10. CanSM > CanSMConfiguration > set up a new transceiver



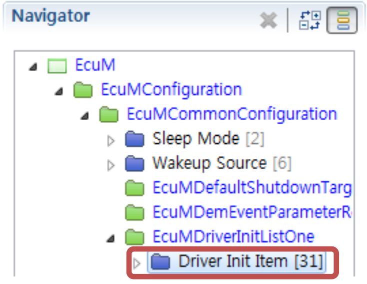
Container Details - CanSMManagerNetwork

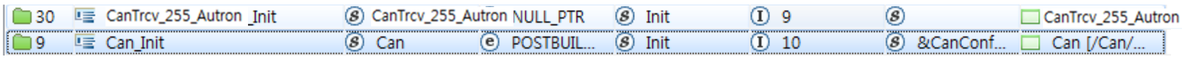
Short Name*:	CanSMManagerNetwork_Atm
Bor Counter L1 To L2*:	20
Bor Time L1*:	0.1
Bor Time L2*:	0.2
Bor Time Tx Ensured*:	0.05
Bor Tx Confirmation Polling*:	□ false
ComM Network Handle Ref*:	ComMChannel_Atm [/AUTOSAR/ComM/ComMConfig
Controller	1 [1...*]

To Be Configured:

Transceiver Id:	CanIfTrcvCfg0 [/AUTOSAR/CanIf/CanIfTrcvDrvCfg1/CanIfTrcvCfg0] (/

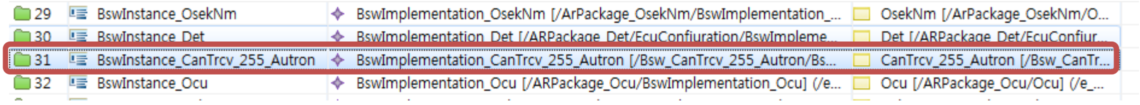
11. Add the Init function of the new CanTrcv module at EcuM 〉 EcuMConfiguration 〉 EcuMCommonConfiguration 〉 EcuMDriverInitListOne (should be one right after the existing CanTrcv)





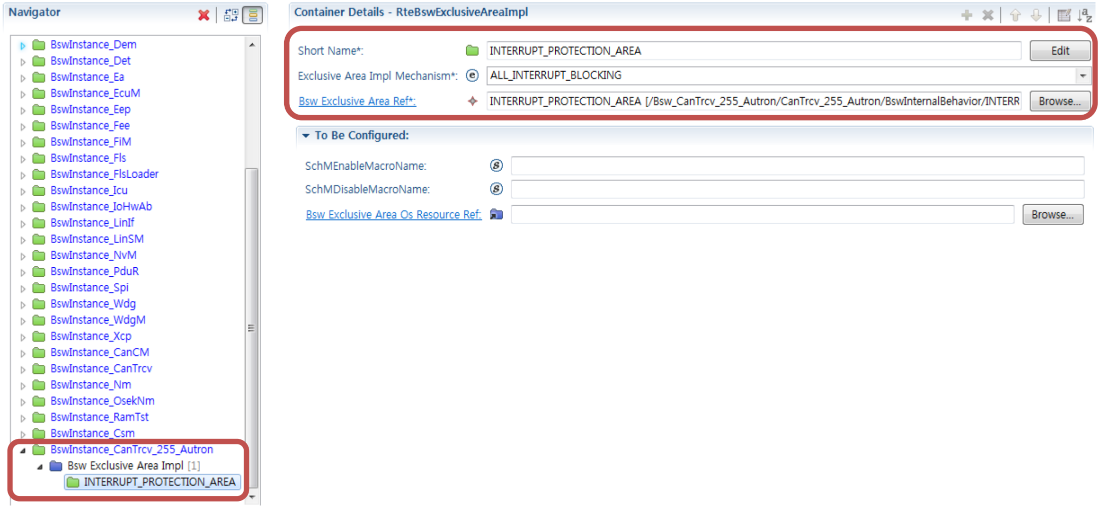
30	CanTrcv_255_Autron _Init	CanTrcv_255_Autron	NULL_PTR	Init	9		CanTrcv_255_Autron
9	Can_Init	Can	POSTBUIL...	Init	10	&CanConf...	Can [/Can/...

12. Add the CanTrcv module at Rte 〉 Bsw Module Instance



29	BswInstance_OsekNm	BswImplementation_OsekNm [/ArPackage_OsekNm/BswImplementation_...	OsekNm [/ArPackage_OsekNm/O...
30	BswInstance_Det	BswImplementation_Det [/ARPackage_Det/EcuConfiuration/BswImpleme...	Det [/ARPackage_Det/EcuConfiur...
31	BswInstance_CanTrcv_255_Autron	BswImplementation_CanTrcv_255_Autron [/Bsw_CanTrcv_255_Autron/Bs...	CanTrcv_255_Autron [/Bsw_CanTr...
32	BswInstance_Ocu	BswImplementation_Ocu [/ARPackage_Ocu/BswImplementation_Ocu] (/e...	Ocu [/ARPackage_Ocu/Ocu] (/e...

13. Add a Bsw Exclusive Area at Rte 〉 Bsw Module Instance



Container Details - RteBswExclusiveAreaImpl

Short Name*:	INTERRUPT_PROTECTION_AREA
Exclusive Area Impl Mechanism*:	ALL_INTERRUPT_BLOCKING
Bsw Exclusive Area Ref*:	INTERRUPT_PROTECTION_AREA [/Bsw_CanTrcv_255_Autron/CanTrcv_255_Autron/BswInternalBehavior/INTERR

To Be Configured:

SchMEnableMacroName:	
SchMDisableMacroName:	
Bsw Exclusive Area Os Resource Ref:	

14. Edit Generate.py: Add the new Ecud_CanTrcv_ 255_VendorspecificName and Bswmd_CanTrcv_ 255_VendorspecificName (if necessary) to GenerateCanIf, GenerateCanSm, GenerateEcuM, and GenerateRte.

15. Edit Generate.py: Add the new Ecud_CanTrcv_ 255_VendorspecificName and Bswmd_CanTrcv_ 255_VendorspecificName to GenerateCanIf, GenerateCanSm, and GenerateEcuM.

9.1.4 Precautions for setting the CANTRCV module

① CANTRCV's CanTrcvChannelId and CanIf's CanIfTrcvId must be identical.

(Otherwise, an error will occur in CanIf.exe.)

② When it comes to the CanTrcvChannelId of CANTRCV, the CanTrcvChannelId of AUTOEVER CANTRCV must set first before the external CANTRCV's CanTrcvChannelId is configured.

③ The AUTOEVER CANTRCV module doesn't support the transceiver wakeup feature.

9.1.5 Behavior description for the CANTRCV module

For the basics, refer to the AUTOSAR CANTRCV specifications.

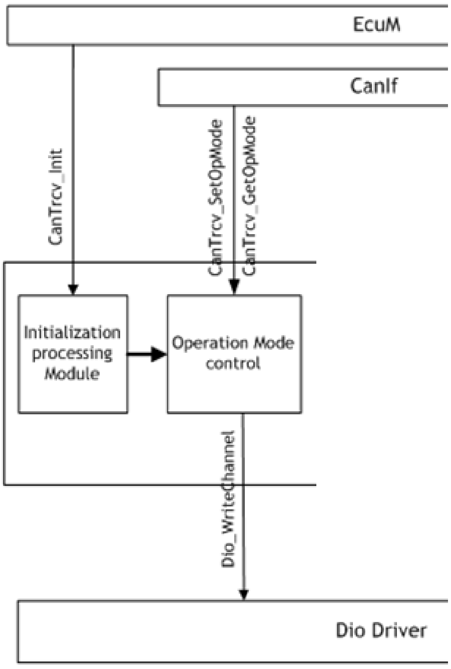
1. When making a request for full communication, request for CANTRCV_TRCVMODE_NORMAL by making an API call of CanTrcv_255_VendorspecificName_SetOpMode.

2. When making a request for no communication, request for CANTRCV_TRCVMODE_STANDBY by making an API call of CanTrcv_255_VendorspecificName_SetOpMode.

   (If the CANTranceiver provides both standby mode and sleep mode by hardware, the user must make their own decision and design the software to function in either mode.

   AUTOSAR requests for STANDBY, sending a no communication command.)

3. After the mode is successfully switched, CanIf_TrcvModeIndication API must be used to make indication call related to the new mode.

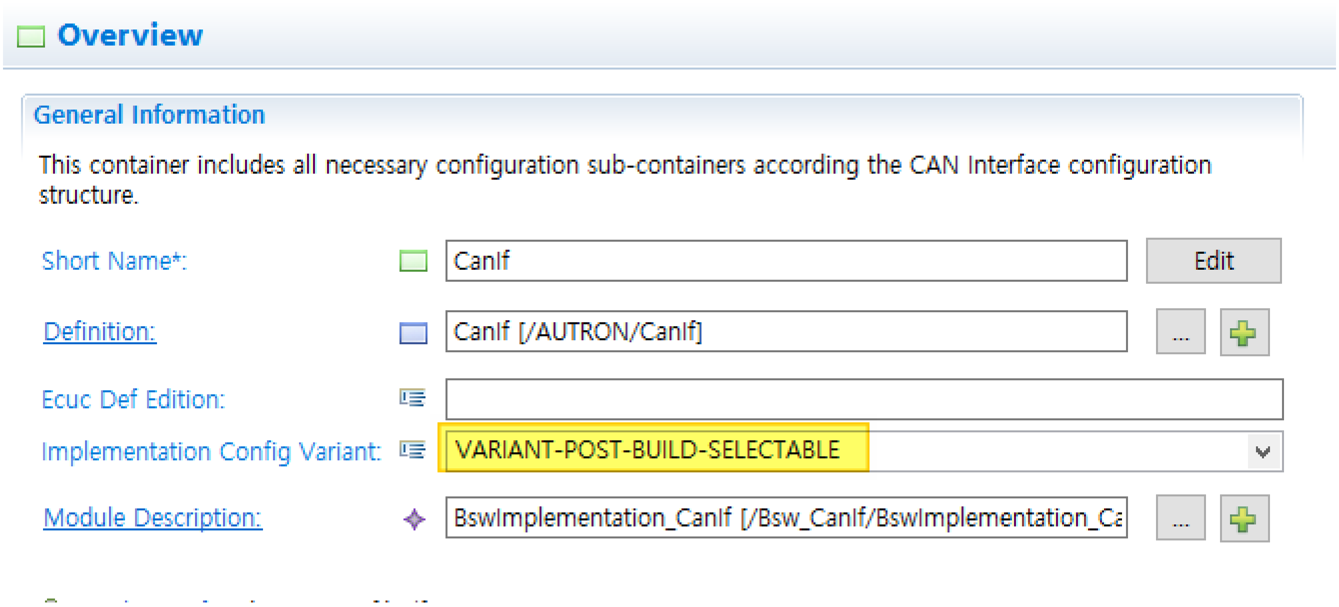


9.1.6 Precautions for when selecting CANTRCV hardware

To decide whether the communication is awake or not in the CANCM module, check the level of CAN RX (low).
In case CANTRCV hardware is selected, if the wakeup level is not maintained as low, the function can't be used.

9.2 PostBuild settings

In order to use the PostBuild feature, the implementation configuration variant value should be set to VARIANT-POST-BUILD-SELECTABLE.



Overview

General Information

This container includes all necessary configuration sub-containers according the CAN Interface configuration structure.

Short Name*: CanIf

Definition: CanIf [/AUTRON/CanIf]

Ecuc Def Edition:

Implementation Config Variant: VARIANT-POST-BUILD-SELECTABLE

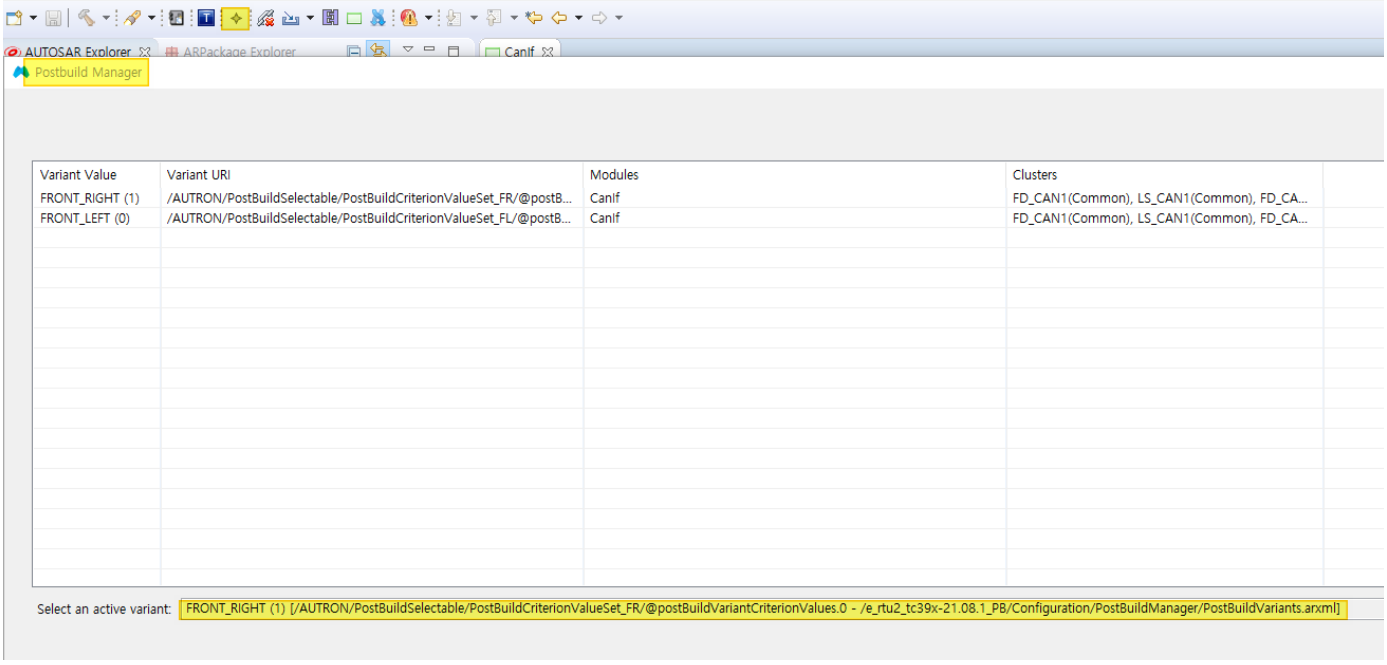
Module Description: BswImplementation_CanIf [/Bsw_CanIf/BswImplementation_Ca

As the support scope of PostBuild R40 is limited to Can Id, modifying any other parameter is not supported.

In order to change Can Id, configure the active variant using the Postbuild Manager and modify CanIfTxPduCanId or CanIfRxPduCanId for the selected variant before setting the 'apply variant' to true; unlike Precompile, the generated files in Postbuild also include CanIf_PBcfg.c/CanIf_PBcfg.h.

Detailed steps are explained below. (Mobilgene for PostBuild should be used.)

1. Configure the active variant using the Postbuild Manager.

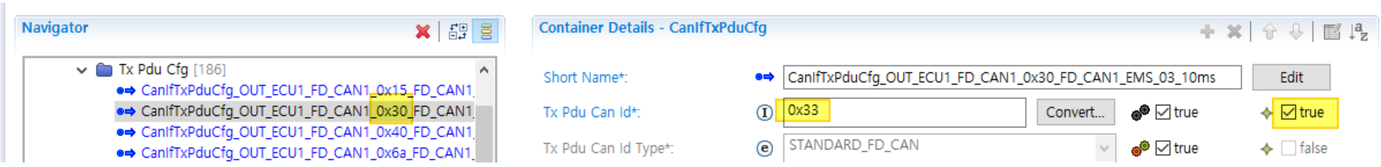


Variant Value	Variant URI	Modules	Clusters
FRONT_RIGHT (1)	/AUTRON/PostBuildSelectable/PostBuildCriterionValueSet_FR/@postB...	CanIf	FD_CAN1(Common), LS_CAN1(Common), FD_CA...
FRONT_LEFT (0)	/AUTRON/PostBuildSelectable/PostBuildCriterionValueSet_FL/@postB...	CanIf	FD_CAN1(Common), LS_CAN1(Common), FD_CA...

Select an active variant: FRONT_RIGHT (1) [/AUTRON/PostBuildSelectable/PostBuildCriterionValueSet_FR/@postBuildVariantCriterionValues.0 - /e_rtu2_tc39x-21.08.1_PB/Configuration/PostBuildManager/PostBuildVariants.arxml]

2. Modify the relevant CanIfTxPduCanId and CanIfRxPduCanId values; switch the 'apply variant' setting to true. Rx only functions with messages bound with Basic Can; check the filter range

value of the relevant Rx hardware object before modifying. (When PostBuild becomes enabled, the range from 0x500 to 0x7FF and J1939 messages are generated as Basic.



Navigator

Tx Pdu Cfg [186]
- CanIfTxPduCfg_OUT_ECU1_FD_CAN1_0x15_FD_CAN1_
- CanIfTxPduCfg_OUT_ECU1_FD_CAN1_0x30_FD_CAN1_
- CanIfTxPduCfg_OUT_ECU1_FD_CAN1_0x40_FD_CAN1_
- CanIfTxPduCfg_OUT_ECU1_FD_CAN1_0x6a_FD_CAN1_

Container Details - CanIfTxPduCfg

Short Name*: CanIfTxPduCfg_OUT_ECU1_FD_CAN1_0x30_FD_CAN1_EMS_03_10ms

Tx Pdu Can Id*: 0x33 ☑ true ☑ true

Tx Pdu Can Id Type*: STANDARD_FD_CAN ☑ true □ false

3. When the apply variant setting is checked as shown above, the value in other variants get erased. Therefore, the Can ID should be manually set in these other variants. (The apply variant setting for the Can ID in those variants should also be set to true.)

4. When the build is complete, CanIf_PBcfg.c and CanIf_PBcfg.h are generated.