

SCOPE OF APPLICATION All Project/Engineering	HYUNDAI AutoEver	SHT/SHTS 1 / 181
Responsibility: Classic AUTOSAR Team	AUTOSAR COM User Manual	DOC. NO
AUTOSAR COM User Manual		

Document Change History				
Date (YYYY-MM-DD)	Ver.	Editor	Chap	Content
2020-12-04	1.0.0.0	SM Kwon	ALL	Initial Version
2021-08-27	1.0.1.0	SM Kwon	4.2 4.3	- Scope Of Release - Add Change Log
2022-02-21	1.1.0.0	PhucNHM	4.3 5 6.3 7.1.1 7.2.2 8	- Add Change Log - Add Configuration Guide - Add Function Definition - Add Error Message - Add Information Message - Add Appendix
2022-04-13	1.1.1.0	PhucNHM	4.2 4.3 7.2 8	- Scope of Release - Add Change Log - Update Generator Error Message - Add Appendix
2022-04-28	1.1.2.0	PhucNHM	4.2 4.3 5.9	- Scope of Release - Add Change Log - Update configuration parameter name from "ComDirectMainFunctionBaseTransmit" to "ComDirectTransmit" and its description.
2022-05-17	1.1.3.0	SMKwon	4.2 4.3 7.1.1	- Scope of Release - Add Change Log - Update the description of ERR050003
2022-06-21	1.1.3.1	SMKwon	4.3	- Add Change Log
2022-08-16	1.1.4.0	PhucNHM	4.2 4.3	- Scope of Release - Add Change Log
2022-10-27	1.1.5.0	PhucNHM	4.2 4.3 7.1.1	- Scope of Release - Add Change Log - Add validation rule ERR050163
2022-11-14	1.1.5.1	PhucNHM	4.3	- Add Change Log
2022-12-05	1.1.6.0	PhucNHM	4.3	- Add Change Log
2022-12-21	1.1.7.0	PhucNHM	4.3	- Add Change Log
2023-03-05	1.1.8.0	PhucNHM	4.2 4.3 7.2.2	- Scope of Release - Add Change Log - Add validation rule WARN050001

Edition Date: 2023-11-17	File Name Com_UM.pdf	Creation PhucNHM 2023-11-17	Check JH Cho 2023-11-17	Approval JH Jeong 2023-11-17
Document Management System				

2023-04-11	1.1.9.0	PhucNHM	4.2 4.3 7.2.2	- Scope of Release - Add Change Log - Add validation rule ERR050164
2023-05-31	1.1.10.0	PhucNHM	4.2 4.3	- Scope of Release - Add Change Log
2023-09-21	1.1.10.1	PhucNHM	4.2 4.3	- Scope of Release - Add Change Log
2023-11-17	1.1.11.0	PhucNHM	4.2 4.3 7.1.1	- Scope of Release - Add Change Log - Add validation rule ERR050165

Table of Contents

1. Overview	8
2. Reference	8
3. AUTOSAR System	9
3.1 Com Module	9
4. Product Release Notes.....	10
4.1 Overview	10
4.2 Scope of the Release.....	10
4.3 Change Log.....	10
4.3.1 Version 1.0.0.0	10
4.3.2 Version 1.0.1.0	10
4.3.3 Version 1.1.0.0	10
4.3.4 Version 1.1.1.0	11
4.3.5 Version 1.1.2.0	12
4.3.6 Version 1.1.3.1	15
4.3.7 Version 1.1.3.0	15
4.3.8 Version 1.1.4.0	15
4.3.9 Version 1.1.5.0	16
4.3.10 Version 1.1.5.1	17
4.3.11 Version 1.1.6.0	17
4.3.12 Version 1.1.7.0	18
4.3.13 Version 1.1.8.0	18
4.3.14 Version 1.1.9.0	19
4.3.15 Version 1.1.10.0	20
4.3.16 Version 1.1.10.1	21
4.3.17 Version 1.1.11.0	21
4.4 Limitations.....	23
4.5 Deviations.....	23
5. Configuration Guide	24

5.1 ComGeneral	25
5.2 ComConfig	25
5.3 Comtimebase.....	26
5.4 ComFilter	27
5.5 ComConfig-ComIpdu	27
5.6 ComConfig-ComIPdu-ComIPduCounter	29
5.7 ComConfig-ComIPdu-ComIPduReplication.....	29
5.8 ComConfig-ComIPdu-ComTxIPdu	30
5.9 ComConfig-ComIPdu-ComTxIPdu-ComTxMode(True/False)-ComTxMode.....	31
5.10 ComConfig-ComIPduGroup.....	32
5.11 ComConfig-ComSignal	32
5.12 ComConfig-ComSingalGroup	35
5.13 ComConfig-ComSingalGroup-ComGroupSignal	37
5.13 ComConfig-ComGwMapping	38
5.14 ComGwMapping-ComGwSource	39
5.15 ComGwMapping-ComGwSource-GwSignal.....	39
5.16 ComGwMapping-ComGwSource-ComGwSourceDescription.....	39
5.17 ComGwMapping-ComGwDestiantion	39
5.18 ComGwMapping-ComGwDestiantion- ComGwDestinationDescription	39
5.19 ComGwMapping-ComGwDestiantion-GwSignal.....	40
6 Application Programming Interface (API).....	40
6.1 Type Definitions	40
6.2 Macro Constants	40
6.3 Functions Definition.....	40
6.3.1 Initialization.....	40
6.3.2 Communication Services	43
6.3.3 Callback function	49
6.3.4 Schedule function	54
7. Generator.....	55
7.1 Generator Error Message	56
7.1.1 Error message	56

7.1.2 Warning Messages	78
7.1.3 Information Messages	78
8. Appendix	78
8.1 Time out.....	78
8.2 Tx Mainfunction	78
8.3 GW Mainfunction.....	78
8.4 Signal group Configuration	79
8.5 Transmit Mode Selection.....	79
8.6 Direct Transmission Without Mainfunction	79
8.7 Multiplexed Message Guide	79
8.8 Signal Gateway Configuration Guide.....	80
8.9 Signal Gateway Callout Function Guide	80
8.9.1 Function overview	80
8.9.2 Configuration	81
8.9.3 Auto setup related.....	85
8.10 Com Signal Callback Function Naming rule and setting	86
8.11 Signal Port Connection and Reception Signal Notification Settings	88
8.11.1 Reception Notification Setting.....	89
8.11.2 Reception Timeout Notification Setting.....	106
8.11.3 Reception Invalid Notification Setting	112
8.12 Signal Port Connection and Transmission Signal Notification Settings.....	113
8.12.1 Transmission Notification Setting	113
8.12.2 Transmission Timeout Notification Settings	127
8.13 Alternative settings for 8.10, 8.11 (Rx) using Create Ports From Network.....	129
8.13.1 Com Harmonize.....	129
8.13.2 Port and Comspec Creation	130
8.13.3 Data Mapping.....	135
8.13.4 Runnable Setting.....	138
8.13.5 DataReceivePointByArgument/DataReceivePointByValue Setting	139
8.13.6 Check your Runnable settings	139
8.13.7 DataReceivedEvent settings.....	140

8.13.8	Checking the DataReceivedEvent	141
8.13.9	Setting DataReceiveErrorEvent.....	142
8.13.10	Check the DataReceiveErrorEvent setting	143
8.13.11	Timeout setting and confirmation.....	145
8.13.12	TaskMapping	148
8.14	Alternative settings for 8.10, 8.12 (Tx) using 8.14 Create Ports From Network	149
8.14.1	Com Harmonize.....	149
8.14.2	Creating Ports and Comspecs.....	149
8.14.3	Data Mapping	150
8.14.4	Runnable settings	152
8.14.5	DataSendPoint settings.....	152
8.14.6	Runnable settings	153
8.14.7	Setting DataSendCompletedEvent	153
8.14.8	Timeout setting.....	154
8.14.9	DataSendCompletedEvent, Timeout setting.....	154
8.14.10	TaskMapping	156
8.15	UINT8_N, UINT8_DYN Signal setting.....	156
8.15.1	Configuration	156
8.16	Guide to check when Com communication is possible.....	170
8.16.1	Background	170
8.16.2	Setup Guide.....	170
8.16.3	Mode Switch Interface Port Creation (Option)	174
8.17	Group Signal IdsM Signal Value Check Setting Guide.....	175
8.17.1	Background	175
8.17.2	Setup Guide.....	175
8.18	Setting Guide for adding application condition to Pdu Group Switch condition.....	177
8.18.1	Mode creation and interface definition.....	177
8.18.2	Mode Request related Application SW-C setting	177
8.18.3	Service Needs Setting.....	177
8.18.4	P-Port, R-Port Connection	179
8.18.5	Create ModeRequestPort, ModeCondition	179

DOCUMENT NUMBER (DOC NO)	SHT/SHTS 7 / 181
-----------------------------	---------------------

8.18.6	Creating LEs	180
8.18.7	Add LE to RuleLE	180
8.18.8	BswM Harmonize	180
8.19	Guide PDU data initialization function guide according to I-PDU Group Control	181
8.19.1	Function overview	181
8.19.2	Configuration	181

1. Overview

This document is noted when user set parameter or design systems in them COM module of the Hyundai Autosar platform or provide a reference. This manual was written base on AUTOSAR standard SRS/SWS It is written based on AUTOSAR standard SRS / SWS. If more detailed functional explanation is needed when using the module, see the Reference Manual. The interpretation of setting related category is as follows:

- Changeable (**C**): Items that can be set by the user.
- Fixed (**F**): Items that cannot be changed by the user.
- Not Supported (**N**): Deprecated item.

2. Reference

Sl. No.	Title	Version
1	AUTOSAR_SWS_COM.pdf	4.4.0
2	AUTOSAR_SRS_COM.pdf	4.4.0

3. AUTOSAR System

3.1 Com Module

Com module is a module that manages overall transmission and reception of messages in ECU
Provision of signal oriented data interface for the RTE.

- Packing of AUTOSAR signals to I-PDUs to be transmitted.
- Un-packing of received I-PDUs and provision of received signals to RTE.
- Routing of signals from received I-PDUs into I-PDUs to become transmitted.
- Routing of signal groups from received I-PDUs into I-PDUs to become transmitted.
- Communication transmission control (start/ stop of I-PDU groups).
- Replications of send requests.
- Guarantee of minimum distances between transmit I-PDUs.
- Monitor of receive signals (signals timeout).
- Filter mechanisms for incoming signals.
- Different notification mechanisms.
- Provision of initial values and update indications.
- Byte order conversion.
- Sign extension.
- Support of two different transmission modes per I-PDU.
- Signal based gateway.
- Support of large and dynamic length data type.

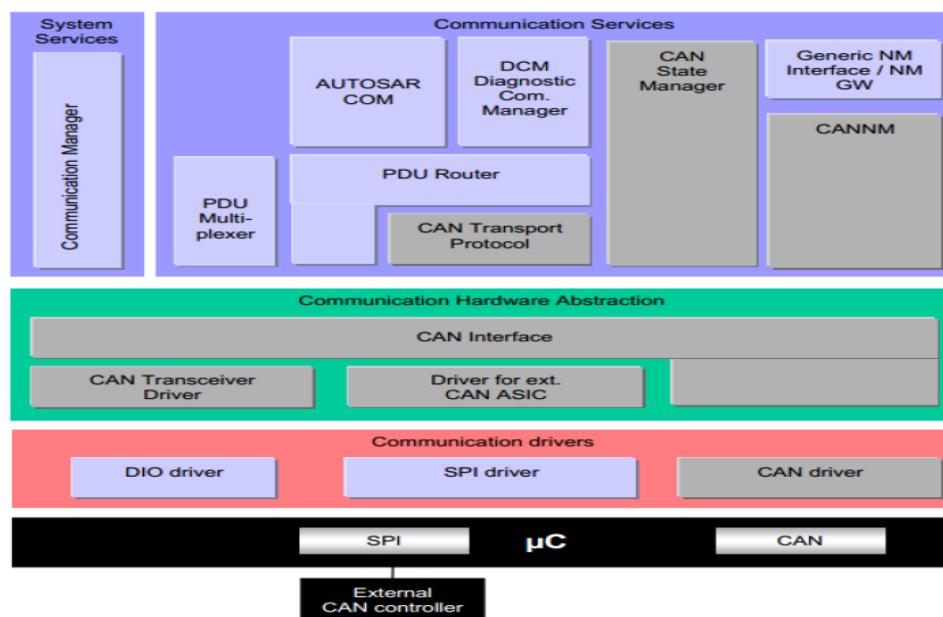


Figure 1

4. Product Release Notes

4.1 Overview

This chapter aims to provide the release information for the Hyundai Autoever Com module.
Describes the limitations and specifics about the software product release version

4.2 Scope of the Release

All contents in this document are limited to the following HUYNDAI AUTOEVER Com modules.

Module Name	AUTOSAR Version	Module Version
Com	4.4.0	1.1.11

※ Module version means the Sw version of each module's BswModule Description (Bswmd) file.

4.3 Change Log

4.3.1 Version 1.0.0.0

- Feature
 - New Com module development

Cause	New development
Operation effect	None
Setting effect	None
ASW Action	None

4.3.2 Version 1.0.1.0

- Defect
 - Fix the problem with using Signal Group only

Cause	If there is no configuration for signal and there're only configuration for Signal Group in Com configuration, it doesn't work.
Operation effect	None
Setting effect	None
ASW Action	None

4.3.3 Version 1.1.0.0

- Change Request
 - R44 Migration: Com Post-build and multicore features.

Cause	Merge R44 Com post-build and multicore functionalities
Operation effect	None

Setting effect	None
ASW Action	None

➤ Defect

- Fix invalid memory section defined for Com data.

Cause	Memory mapping section defined for Com RAM array is invalid
Operation effect	None
Setting effect	None
ASW Action	None

➤ Change Request

- Fix Com compilation warning for RTU.

Cause	When building Com source code, the S32G RTU raise some compile warning.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- Fix invalid initial value generate for FLOAT64 signal.

Cause	Reception timeout action replaces invalid generated initial data in the case signal is FLOAT64.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- Unable to handle reception timeout action SUBSTITUTE

Cause	ComTimeoutSubstitutionValue is not generated when ComRxDataTimeoutAction is SUBSTITUTE
Operation effect	None
Setting effect	None
ASW Action	None

4.3.4 Version 1.1.1.0

➤ Change Request

- Implement missing SUD for Com.

Cause	Missing SUD part
Operation effect	None

Setting effect	None
ASW Action	None

➤ Change Request

- Add missing requirements for Post-build and Multicore features.

Cause	Missing ID of Post-build and Multicore in module
Operation effect	None
Setting effect	None
ASW Action	None

➤ Change Request

- Update Com documents for its missing information (EA-UM).

Cause	EA missing Id and relationship, UM missing Appendix and Error message is obsoleted
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- Replace all _COUNT_MAX used for Det check by using global variable get from Com_PBConfig[].

Cause	Use COUNT_MAX to check for handle id of different variants can cause error when handle id not reach max (of all variants), but out scope (of that variant)
Operation effect	None
Setting effect	None
ASW Action	None

4.3.5 Version 1.1.2.0

➤ Defect

- Fix invalid pre-compile condition in Com_MainFunctionTx.

Cause	Some pre-compile conditions were defined but not used in Com_MainFunctionTx.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- All variable declarations should be put at Com_Ram.h instead of Com_PCTypes.h.

Cause	Move some variable declarations to correct place in the code.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- Fix invalid memmap section defined for Com arrays.

Cause	Some memmap sections were mapped to Com arrays, which should be mapped to UNSPECIFIED.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- Com_CopyRxData should allow NULL_PTR SduDataPtr when SduDataLength is 0.

Cause	Com_CopyRxData should allow SduDataPtr 'NULL_PTR' for the parameter 'info' when SduDataLength of 'info' is set to 0.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- I-PDU should be started if at least one I-PDU group which it belongs to is started.

Cause	I-PDU should be started if at least one I-PDU group which it belongs to is started.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- Overwrite of Receive I-PDU List can be happened when using LIN communication.

Cause	Overwrite of Receive I-PDU List can be happened when using LIN communication (R40 port-fix).
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- I-PDU group can be stopped twice or can be started when Bus-off has happened.

Cause	I-PDU group can be stopped twice or can be started when Bus-off has happened (R40 port-fix).
Operation effect	None
Setting effect	/AUTRON/Rte/BswInstance_Com/Bsw Exclusive Area Impl Should add new Bsw Exclusive Area Impl in Rte Configuration by harmonizing Rte again
ASW Action	None

➤ Change Request

- Fix UNECE security coding rule violations.

Cause	Fix UNECE security coding rule violations.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- ComM_COMCbk_<EIRA> compile error occurs when using PNC and Rte R44.

Cause	“ComM_COMCbk_ComISignal_Rx_EIRA_CanNm” compile error occurs when using PNC and Rte R44 is applied.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Defect

- Increment for RxDmStatus should be performed outside the RxIPduStatus check.

Cause	Corrected the problem that the Rx Deadline Monitoring function of the signal included in the I-PDU does not work when two or more of the I-PDU groups to which the I-PDU belongs are started and only one is stopped.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Change Request

- Change the parameter ComDirectMainFunctionBaseTransmit to ComDirectTransmit.

Cause	Change the parameter name “ComDirectMainFunctionBaseTransmit” to “ComDirectTransmit”.
Operation effect	None
Setting effect	None
ASW Action	None

4.3.6 Version 1.1.3.1

➤ Task

- Editorial Change for Work Products

Cause	Add copyright comment in the source codes, apply the latest template to DeliveryBoxHistory document
Operation effect	None
Setting effect	None
ASW Action	None

4.3.7 Version 1.1.3.0

➤ Defect

- Fix compilation error when Partial Network is not supported

Cause	“Com.h” always includes “ComM_Com.h”, which is generated only when Partial Network is supported. So it should be fixed including “ComM_Com.h” only when Partial Network is supported.
Operation effect	None
Setting effect	None
ASW Action	None

4.3.8 Version 1.1.4.0

➤ Defect

- Com_IpduGroupStart should initialize buffers as defined in AUTOSAR specification

Cause	Com_IpduGroupStart does not initialize IPDU buffer again as defined in SAD specification
Operation effect	None
Setting effect	None
ASW Action	None

➤ Change Request

- Support multicore for CancelTransmit call flow

Cause	Support multicore for calling PduR_CancelTransmit
Operation effect	None
Setting effect	None
ASW Action	None

➤ Change Request

- Fix polyspace cyber security UNECE violations

Cause	Fix polyspace cyber security UNECE violations
Operation effect	None
Setting effect	None
ASW Action	None

4.3.9 Version 1.1.5.0

➤ Bug

- Fix "MASK_NEW_DIFFERS_MASKED_OLD" doesn't work with 16, 32 bit Rx signals after Rx timeout

Cause	Fix "MASK_NEW_DIFFERS_MASKED_OLD" doesn't work with 16, 32 bit Rx signals after Rx timeout
Operation effect	None
Setting effect	None
ASW Action	None

➤ Bug

- Fix packing logic of uint16 and uint32 when violate TC33x memory structure

Cause	Fix packing logic of uint16 and uint32 when violate TC33x memory structure
Operation effect	None
Setting effect	None
ASW Action	None

➤ Improvement

- ES spec coverage improvement - Add necessary work products for Signal Gateway Callout

Cause	ES spec coverage improvement - Add necessary work products for Signal Gateway Callout
Operation effect	None
Setting effect	None
ASW Action	None

➤ Improvement

- Update post-build concept

Cause	Update post-build concept
Operation effect	None

Setting effect	None
ASW Action	None

4.3.10 Version 1.1.5.1

➤ Improvement

- Improve QT tests for testing indirect gateway features in ES91821_06E.

Cause	Add SQT tests for testing below features of indirect gateway - manipulation of received data - modification of transmission properties - rx message timeout handling
Operation effect	None
Setting effect	None
ASW Action	None

4.3.11 Version 1.1.6.0

➤ Bug

- Added Com_PBcfg.h to Com.h to fix compile error related to PB config in EcuM

Cause	Added Com_PBcfg.h to Com.h to fix compile error related to PB config in EcuM
Operation effect	None
Setting effect	None
ASW Action	None

➤ Bug

- When accessing the Com_GaaTxSignal array when setting up Post Build, the problem of accessing an unintended memory area out of range has been fixed.

Cause	When accessing the Com_GaaTxSignal array when setting up Post Build, the problem of accessing an unintended memory area out of range has been fixed.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Bug

- Fixed the problem that the intended signal was not transmitted because the Handle Id of Com_GaaTxSignal was created incorrectly when Post Build was set for Com Signal.

Cause	Fixed the problem that the intended signal was not transmitted because the Handle Id of Com_GaaTxSignal was created incorrectly when Post Build was set for Com Signal.
Operation effect	None

Setting effect	None
ASW Action	None

➤ Bug

- Fixed the problem that the Com_GaaTxIpduCounter PB pointer was not initialized in Com_Init so that the Null Pointer was referenced when using the Tx IPDU Counter.

Cause	Fixed the problem that the Com_GaaTxIpduCounter PB pointer was not initialized in Com_Init, so that the Null Pointer was referenced when using the Tx IPDU Counter.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Bug

- Fixed an error where an incorrectly generated value was assigned to Com_GINoOfIpduGrp

Cause	Fixed an error where an incorrectly generated value was assigned to Com_GINoOfIpduGrp
Operation effect	None
Setting effect	None
ASW Action	None

➤ Improvement

- Fixed UNECE and ES95411-00 violation based on tc39x RTU

Cause	Fixed UNECE and ES95411-00 violation based on tc39x RTU
Operation effect	None
Setting effect	None
ASW Action	None

4.3.12 Version 1.1.7.0

➤ Bug

- PB multiplicity should be supported for ComSignalGroup

Cause	PB multiplicity should be supported for ComSignalGroup
Operation effect	None
Setting effect	None
ASW Action	None

4.3.13 Version 1.1.8.0

➤ Bug

- Com transmits TRIGGERED/TRIGGERED_ON_CHANGE signal only ComTxModeNumberOfRepetitions time when the message is MainFunctionBased-

Transmit

Cause	Com transmits TRIGGERED/TRIGGERED_ON_CHANGE signal only ComTxModeNumberOfRepetitions time when the message is MainFunctionBasedTransmit
Operation effect	None
Setting effect	None
ASW Action	None

➤ Bug

- Tx IPdus are transmitted in Direct Transmit Mode even though ComMainFunction-BasedTransmit is set to 'true'

Cause	Com generator doesn't care about ComMainFunctionBasedTransmit parameter and only cares for ComDirectTransmit parameter when generating ucDirectMainTransmit for both direct and mainfunction based values
Operation effect	None
Setting effect	None
ASW Action	None

4.3.14 Version 1.1.9.0

➤ Bug

- Com generates ddRxSigIndex as start index of ComSignals list. Add validation to ensure the IPDU's ComSignal list HandleId are sequential

Cause	Com generates invalid ddRxSigIndex when orders of ComIPdu and ComSignals are not match
Operation effect	None
Setting effect	None
ASW Action	None

➤ Bug

- Com transmit TRIGGERED/TRIGGER_ON_CHANGE signal only ComTxModeNumberOfRepetitions time when signal is triggered after MDT and the message is Direct Transmit Mode

Cause	Com transmit TRIGGERED/TRIGGER_ON_CHANGE signal time different with ComTxModeNumberOfRepetitions when signal is triggered after MDT and the message is Direct Transmit Mode
Operation effect	None
Setting effect	None

ASW Action	None
------------	------

➤ Bug

- Com Ipdu PERIODIC and TRIGGER* signal is not allowed in configuration

Cause	Configure Com Ipdu PERIODIC refer to TRIGGER*
Operation effect	None
Setting effect	None
ASW Action	None

➤ Improvement

- Update SUD diagrams in EA to match with latest code design

Cause	SUD in EA is not match with latest code design
Operation effect	None
Setting effect	None
ASW Action	None

➤ Improvement

- Fix A-SPICE final inspection findings

Cause	A-SPICE final inspection findings
Operation effect	None
Setting effect	None
ASW Action	None

4.3.15 Version 1.1.10.0

➤ Development

- Fix compile warning - Com R44

Cause	Compile warning on RTU
Operation effect	None
Setting effect	None
ASW Action	None

➤ Task

- [Orin_FSI] Request to review problems that occur during Reference Platform (Alpha-RTU) integration

Cause	Modifying COM_RX_IPDU_GWSTATUS_COUNT precompile options to match reference logic and declaration logic
Operation effect	None

Setting effect	None
ASW Action	None

➤ Bug

- 'ComUserCbkHeaderFile' doesn't work

Cause	'ComUserCbkHeaderFile' have to develop in R44 Com Generator
Operation effect	None
Setting effect	"ComGeneral" > "ComUserCbkHeaderFile" in Ecud_Com (If use)
ASW Action	None

➤ Task

- Update UM template

Cause	Update UM template
Operation effect	None
Setting effect	None
ASW Action	None

4.3.16 Version 1.1.10.1

➤ Task

- Fix Gentoool Version Dependency

Cause	Gentoool compatible version confirmation request
Operation effect	None
Setting effect	None
ASW Action	None

4.3.17 Version 1.1.11.0

➤ Bug

- Fix repetition transmission counter for MainFunction based TMS

Cause	Com transmits direct repetition message with repetition + 2 when MainFunction based transmit is applied
Operation effect	None
Setting effect	None
ASW Action	None

➤ Bug

- Improve data ordering for signal buffer in case RxDataTimeoutAction is SUBSTITUTION

Cause	Signal data is corrupted while packing BIG_ENDIAN signal for its SUBSTITUTION value in case RxDataTimeout
Operation effect	None
Setting effect	None
ASW Action	None

➤ Bug

- Remove SUBSTITUTE dependency when generating Signal Buffer

Cause	Signal data is replaced incorrectly with Substitution data instead of InvalidData during signal invalidation.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Bug

- Remove unnecessary parameter “boolean Initialize” in function Com_IpduGroupStop

Cause	Dummy Com_IpduGroupStop used for Post-build was provided with incorrect prototype.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Bug

- Add exclusive area protection for Com I-PDU reception history

Cause	Com I-PDU reception history was impacted by race condition.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Improvement

- Fix to singleton structure for Gentoool compatible

Cause	For gentool-fw-1.1.6.1 compatible, Com generator was violated in coding style of using singleton. This blocking new framework from improvement its performance.
Operation effect	None
Setting effect	None
ASW Action	None

➤ Improvement

- Add validation rule for naming convention of ComUserCbkHeader

Cause	Validation rule for naming convention of ComUserCbkHeader
Operation effect	None
Setting effect	"ComGeneral" > "ComUserCbkHeaderFile" in Ecud_Com (If use)
ASW Action	None

➤ Improvement

- Fix A-SPICE compliance violations

Cause	Code metric, traceability and code coverage report failed with ASPICE inspection.
Operation effect	None
Setting effect	None
ASW Action	None

4.4 Limitations

➤ I-PDU Counter (Safety function)

In the sender side, the counter value is added immediately before transmission at the configured location of the transmission PDU. The counter value increase to a specific value, and increases again from 0.

Configured in receiver side, check whether the position value is Expected Counter Value and check whether the messages came in order. Whether to discard the message according to the configured Threshold value or to process it by judging it as a normal message. Decide. To use this function, the special location of the message in the DB file is left blank and the Counter value able to go, both Sender and Receiver must support the corresponding function. Features not currently supported.

➤ I-PDU Un-Group:

In both sender and receiver side, An I-PDU is active if it belong to a started I-PDU group or it does not belong to any I-PDU group and it was started implicitly. I-PDU which is not assigned to any I-PDU group will be active when call Com_Init. Therefore Handle ID of IPDU un-group must be in the last in IPDU list.

4.5 Deviations

- Operation according to General->Mainfunction based transmit setting Mainfunction based transmit method is not transmitted immediately upon request of upper layer, but the actual transmit is handled in MainFunction. Because it is processed in the main function, after transmit Functions such as timeout, repetition, and MDT are processed without error. The setting should be used as the default value,

If you want to use the event based transmit method, consult with the person in charge of distribution.

Mainfunction based transmit method receives NOT_OK due to sub-module error from Com_SendSignal.

cannot be heard. (Similar concept to Pending signal processing method).

- Signal Gateway function operation It is based on AUTOSAR 4.0.3 specification, but some 4.2.X specifications were shipped → Perform Signal Gateway at timeout (optional). As the specifications of AUTOSAR 4.0.3, there are parts that are difficult to satisfy the requirements, and some functions have been added. → Signal Gateway callout function
- In case of I-PDU Stop status, Reception Deadline Monitoring process. After the I-PDU is changed to the Stop state, an Enable request for the Reception Deadline Monitoring function is received. In this case, the Com Module determines the I-PDU status as a higher condition and performs Reception Deadline Monitoring.
- ComRxInterruptBased, ComTxInterruptBased
Added as unnecessary Interrupt related handling options
- ComTxSignalValidationCheck
Items added as HMC specifications
- Minimum Delay Time (MDT)
The configured MDT value is calculated in ticks according to the cycle of the Mainfunction. When applying MDT, If a previously transmitted message is affected by another message and is sent late, the message applying MDT is It can be seen on the BUS as being treated as a value smaller than the actual MDT time. to complement this MDT Add 1 to the actual tick value to secure the MDT time.
- Reception Filtering for Group Signal of Signal Group
Reception Filtering for Group Signal is not applied, but Group Signal
- TMS function operation It is based on AUTOSAR 4.0.3 specification, but some 4.2.X specifications have been shipped
- When changing to Periodic mode, even if transmission is delayed by MDT, the cycle starts from the point of change.
- I-PDU Start/Stop Control
It is based on AUTOSAR 4.0.3 specification, but some 4.2.X specifications have been shipped
- In case of I-PDU without I-PDU Group Reference, the PDU is developed to start when the module is initialized.

5. Configuration Guide

The COM setting of the AUTOSAR platform distributed by Hyundai Autoever is set based on the DB file, and if additional functions such as Signal Gateway are required, you must consult with Hyundai Autoever.

5.1 ComGeneral

Parameter Name	Value	Category
ComConfigurationUseDet	True	C
ComEnableMDTForCyclicTransmission	Automated	F
ComRetryFailedTransmitRequests	False	C
ComSupportedIPduGroups	Automated	F
ComVersionInfoApi	False	F
ComEnableSignalGroupArrayApi	False	C
ComCancellationSupport	False	C
ComMetaDataSupport	False	C
ComUserCbkHeaderFile	False	C

- 1) ComConfigurationUseDet:
 - The error hook shall contain code to call the Det.
 - If this parameter is configured COM_DEV_ERROR_DETECT shall be set to ON as output of the configuration tool. (as input for the source code).
- 2) ComEnableMDTForCyclicTransmission:
 - Enables globally for the whole Com module the minimum delay time monitoring for cyclic and repeated transmissions (ComTxModeMode=PERIODIC or ComTxModeMode=MIXED for the cyclic transmissions, ComTxModeNumberOfRepetitions > 0 for repeated transmissions)
- 3) ComRetryFailedTransmitRequests:
 - If this Parameter is set to true, retry of failed transmission requests is enabled.
 - If this Parameter is not present, the default value is assumed.
- 4) ComSupportedIPduGroups:
 - Defines the maximum number of supported I-PDU groups
- 5) ComVersionInfoApi
 - Activate/Deactivate the version information API (Com_GetVersionInfo).
 - True: version information API activated
 - False: version information API deactivated
- 6) ComEnableSignalGroupArrayApi
 - Activate/Deactivate the signal group array access APIs (Com_SendSignalGroupArray, Com_ReceiveSignalGroupArray).
 - true: signal group array access APIs activated false: signal group array access APIs deactives
- 7) ComCancellationSupport
 - This parameter enables/disables the cancellation feature: true: enabled false: disabled
- 8) ComMetaDataSupport
 - This parameter enables/disables the support of meta-data feature including the API Com_TriggerIPDUSendWithMetaData.
- 9) ComUserCbkHeaderFile
 - Defines the header files for callback functions which shall be included by the COM module.

5.2 ComConfig

Parameter Name	Value	Category
ComDataMemSize	User Defined	C
ComMaxIPduCnt	-	F

1) ComDataMemSize

- Size of internal Com data in units of bytes (static memory allocation) - memory required by post-build configuration must be smaller than this constant. This parameter is needed only in case of post-build loadable implementation using static memory allocation.

2) ComMaxIPduCnt

- Maximum number of IPdus. This parameter is needed only in case of post-build loadable implementation using static memory allocation.

5.3 Comtimebase

Parameter Name	Value	Category
ComGwTimeBase	Automated	C
ComRxTimeBase	Automated	C
ComTxTimeBase	Automated	C

1) ComGwTimeBase

- The period between successive calls to Com_MainFunctionRouteSignals in seconds.
- This parameter may be used by the COM generator to transform the values of the signal gateway related timing configuration parameters of the COM module to internal implementation specific counter or tick values.
- The COM module's internal timing handling is implementation specific.

2) ComRxTimeBase

- The period between successive calls to Com_MainFunctionRx in seconds.
- This parameter may be used by the COM generator to transform the values of the reception related timing configuration parameters of the COM module to internal implementation specific counter or tick values.
- The COM module's internal timing handling is implementation specific.

3) ComTxTimeBase

- The period between successive calls to Com_MainFunctionTx in seconds.
- This parameter may be used by the COM generator to transform the values of the transmission related timing configuration parameters of the COM module to internal implementation specific counter or tick values.
- The COM module's internal timing handling is implementation specific.

DOCUMENT NUMBER (DOC NO)	SHT/SHTS 27 / 181
-----------------------------	----------------------

5.4 ComFilter

Parameter Name	Value	Category
ComFilterAlgorithm	User Defined	C
ComFilterMask	-	F
ComFilterMax		
ComFilterMin		
ComFilterOffset		
ComFilterPeriod		
ComFilterX		

- 1) ComFilterAlgorithm
 - The range of values is specified in the [17] specification, chapter 2.2.2, Reception Filtering.
- 2) ComFilterMask
 - The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.
- 3) ComFilterMax
 - The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.
- 4) ComFilterMin
 - The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.
- 5) ComFilterOffset
 - The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering. Range = 0..(ComFilterPeriod-1)
- 6) ComFilterPeriod
 - This parameter defines the period of the ComFilterAlgorithm ONE_EVERY_N.
- 7) ComFilterX
 - The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.

5.5 ComConfig-Comlpdu

Parameter Name	Value	Category
ComIPduCallout	User Defined	C
ComIPduCancellationSupport	-	F
ComIPduDirection	Automated	F
ComIPduHandleId	Automated	F
ComIPduSignalProcessing	DEFERRED	C
ComIPduTriggerTransmitCallout	User Defined	C
ComIPduType	Automated	F
ComIPduGroupRef	Automated	F
ComIPduSignalGroupRef	Automated	F
ComIPduSignalRef	Automated	F
ComPduldRef	Automated	F

1) ComIPduCallout:

- This parameter defines the existence and the name of a callout function for the corresponding I-PDU. If this parameter is omitted no I-PDU callout shall take place for the corresponding I-PDU.
- 2) ComIPduCancellationSupport
- Defines for I-PDUs with ComIPduType NORMAL: If the underlying IF-modul supports cancellation of transmit requests.
 - Defines for I-PDUs with ComIPduType TP: If the underlying TP-module supports RX and TX cancellation of ongoing requests.

3) ComIPduDirection:

- The direction defines if this I-PDU, and therefore the contributing signals and signal groups, shall be sent or received.

4) ComIPduHandleId:

- The numerical value used as the ID of this I-PDU. The ComIPduHandleId is required by the API calls Com_RxIndication, Com_TpRxIndication, Com_StartOfReception and Com_CopyRxData to receive I-PDUs from the PduR (ComIPduDirection: Receive), as well as the Pduld passed to an Rx-I-PDU-callout. For Tx-I-PDUs (ComIPduDirection: Send), this handle Id is used for the APIs calls Com_TxConfirmation, Com_TriggerTransmit, Com_TriggerIPDUSend or Com_TriggerIPDUSendWithMetaData, Com_CopyTxData and Com_TpTxConfirmation to transmit respectively confirm transmissions of I-PDUs, as well as the Pduld passed to the Tx-I-PDU-callout configured with ComIPduCallout and/or ComIPduTriggerTransmitCallout.

5) ComIPduSignalProcessing

- For the definition of the two modes Immediate and Deferred.

6) ComIPduTriggerTransmitCallout

- If there is a trigger transmit callout defined for this I-PDU this parameter contains the name of the callout function.
- 7) ComIPduType
- Defines if this I-PDU is a normal I-PDU that can be sent unfragmented or if this is a large I-PDU that shall be sent via the Transport Protocol of the underlying bus.
- 8) ComIPduGroupRef
- Reference to the I-PDU groups this I-PDU belongs to.
- 9) ComIPduSignalGroupRef
- References to all signal groups contained in this I-Pdu
- 10) ComIPduSignalRef
- References to all signals contained in this I-PDU.
- 11) ComPdulRef
- Reference to the "global" Pdu structure to allow harmonization of handle IDs in the COM-Stack.

5.6 ComConfig-ComIPdu-ComIPduCounter

Parameter Name	Value	Category
ComIPduCounterErrorNotification	-	N
ComIPduCounterSize	-	N
ComIPduCounterStartPosition	-	N
ComIPduCounterThreshold	-	N

- 1) ComIPduCounterErrorNotification
 - Name of Com_CbkCounterErr callback function to be called. If this parameter is omitted no I-PDU counter mismatch notification shall take place.
- 2) ComIPduCounterSize
 - Size of PDU Counter expressed in bits
- 3) ComIPduCounterStartPosition
 - Position of PDU counter expressed in bits from start position of data content of I-PDU (SDU). Note that PDU counter is not allowed to cross a byte border.
 - The parameter ComIPduCounterStartPosition shall define the bit0 of the first byte like in little endian byte order.
- 4) ComIPduCounterThreshold
 - Threshold value of I-PDU counter algorithm,

5.7 ComConfig-ComIPdu-ComIPduReplication

Parameter Name	Value	Category
ComIPduReplicationQuorum	-	N
ComIPduReplicaRef	-	N

- 1) ComIPduReplicationQuorum

- The number of identical I-PDUs needed for successful voting.
- 2) ComIPduReplicaRef
- Reference to replicas PduR PDUs of this IPDU.

5.8 ComConfig-ComIPdu-ComTxIPdu

Parameter Name	Value	Category
ComMetaDataDefault		
ComMinimumDelayTime	Automated	F
ComTxIPduClearUpdateBit	-	N
ComTxIPduUnusedAreasDefault	Automated	C

- 1) ComMetaDataDefault
- In case an I-PDU refers to a globally configured MetaDataType and no explicit meta data is given for a send request (e.g. by using Com_TriggerIPDUSendWithMetaData), the AUTOSAR COM module uses this configured default meta data for sending.
 - The configured string shall be interpreted as a decimal representation of the characters separated by blanks, e.g. "97 98 100" means a string "abd", where the char "a" is in byte 0(lowest address), "b" is in byte 1, and "d" is in byte 2 and (highest address).
- 2) ComMinimumDelayTime
- Defines the Minimum Delay Time (MDT) between successive transmissions of this I-PDU in seconds. The MDT is independent of the possible different transmission modes.
 - There is only one minimum delay time parameter for one I-PDU. The minimum delay timer is not reset by changing the transmission mode.
 - Hence, it is not allowed to violate the minimum delay time by transmission mode changes. It is not possible to monitor the minimum delay time for I-PDUs that are requested using the Com_TriggerTransmit API.
- 3) ComTxIPduClearUpdateBit
- Defines when the update-bits of signals or signal groups, contained in this I-PDU, will be cleared.
- 4) ComTxIPduUnusedAreasDefault
- The AUTOSAR COM module fills not used areas of an I-PDU with this byte pattern.
 - This attribute is mandatory to avoid undefined behaviour.
 - This byte-pattern will be repeated throughout the I-PDU before any init-values or update-bits were set.

5.9 ComConfig-ComIPdu-ComTxIPdu-ComTxMode(True/False)-ComTxMode

Parameter Name	Value	Category
ComTxModeMode	Automated	F
ComTxModeNumberOfRepetitions	Automated	F
ComTxModeRepetitionPeriod	Automated	F
ComTxModeTimeOffset	Automated	F
ComTxModeTimePeriod	Automated	F
ComDirectTransmit	False	C

1) ComTxModeMode

- The available transmission modes described in [18] shall be extended by the additional mode None. The transmission mode None shall not have any further sub-attributes in the ComTxMode object.

2) ComTxModeNumberOfRepetitions

- Defines the number of repetitions for the transmission mode DIRECT and the event driven part of transmission mode MIXED.

3) ComTxModeRepetitionPeriod

- Defines the repetition period in seconds of the multiple transmissions in case ComTxModeNumberOfRepetitions is configured greater than or equal to 1 and ComTxModeMode is configured to DIRECT or MIXED. In case of the mixed transmission mode only the event driven part is affected.

4) ComTxModeTimeOffset

- Defines the period in seconds between the start of the I-PDU by Com_IpduGroupStart and the first transmission request in case ComTxModeMode is configured to PERIODIC or MIXED.
- In case of the mixed transmission mode only the periodic part is affected.
- In case ComTxModeTimeOffset is omitted or configured to 0, the first periodic transmission shall be transmitted within the next invocation of Com_MainFunctionTx.

5) ComTxModeTimePeriod

- Defines the repetition period in seconds of the periodic transmission requests in case ComTxModeMode is configured to PERIODIC or MIXED. In case of the mixed transmission mode only the periodic part is affected.

6) ComDirectTransmit

- If true Com module shall support immediate transmission of IPDU after a Com_SendSignal* is called, without waiting for a Com_MainFunctionTx.
- If false Com Module shall support IPDU transmission follows Com_MainFunctionTx. Com_SendSignal* only updates data to IPDU buffer, and Com_MainFunctionTx called by OS task will invoke PduR_ComTransmit to sent this IPDU to lower layers.

5.10 ComConfig-ComIPduGroup

Parameter Name	Value	Category
ComIPduGroupHandleId	Automated	F
ComIPduGroupGroupRef	-	N

1) ComIPduGroupHandleId

- The numerical value used as the ID of this I-PDU Group .
- The ComIPduGroupHandleId is required by the API calls to start and stop I-PDU Groups. Range: 0 .. (ComSupportedIPduGroups-1)

2) ComIPduGroupGroupRef

- References to all I-PDU groups that includes this I-PDU group.
- If this reference is omitted this I-PDU group does not belong to another I-PDU group.

5.11 ComConfig-ComSignal

Parameter Name	Value	Category
ComBitPosition	Automated	F
ComBitSize	Automated	F
ComDataInvalidAction	User Defined	C
ComErrorNotification	User Defined	C
ComFirstTimeout	User Defined	C
ComHandleId	Automated	F
ComSignalDataInvalidValue	User Defined	C
ComInvalidNotification	User Defined	C
ComNotification	User Defined	C
ComRxDataTimeoutAction	User Defined	C
ComSignalDataInvalidValue		
ComSignalEndianness	Automated	F
ComSignallInitValue	Automated	C
ComSignalLength	Automated	F

Parameter Name	Value	Category
ComSignalType	Automated	F
ComTimeout	Automated	C
ComTimeoutSubstitutionValue		
ComTimeoutNotification	User Defined	C
ComTransferProperty	Automated	F
ComUpdateBitPosition	-	N
ComSigReportIdsMEnable	Automated	C
ComSystemTemplateSystemSignalRef	Automated	F

1) ComBitPosition

- Starting position within the I-PDU. This parameter refers to the position in the I-PDU and not in the shadow buffer. If the endianness conversion is configured to Opaque the parameter ComBitPosition shall define the bit0 of the first byte like in little endian byte order

2) ComBitSize

- Size in bits, for integer signal types.
- For ComSignalType UINT8_N and UINT8_DYN the size shall be configured by ComSignalLength.
- For ComSignalTypes FLOAT32 and FLOAT64 the size is already defined by the signal type and therefore may be omitted.

3) ComDataInvalidAction

- This parameter defines the action performed upon reception of an invalid signal.
- Relating to signal groups the action in case if one of the included signals is an invalid signal.
- If Replace is used the ComSignallInitValue will be used for the replacement.

4) ComErrorNotification

- This parameter defines the action performed upon reception of an invalid signal.
- Relating to signal groups the action in case if one of the included signals is an invalid signal.
- If Replace is used the ComSignallInitValue will be used for the replacement.

5) ComFirstTimeout

- Defines the length of the first deadline monitoring timeout period in seconds.
- This timeout is used immediately after start (or restart) of the deadline monitoring service.
- The timeout period of the successive periods is configured by ECUC_Com_00263.

6) ComHandleId

- The numerical value used as the ID.
- This ID identifies signals and signal groups in the COM APIs using Com_SignalIdType or Com_SignalGroupIdType parameter respectively.

7) ComSignalDataInvalidValue

- Defines the data invalid value of the signal.
 - In case the ComSignalType is UINT8, UINT16, UINT32, UINT64, SINT8, SINT16, SINT32, SINT64 the string shall be interpreted as defined in the chapter Integer Type in the AUTOSAR EcuC specification.
 - In case the ComSignalType is FLOAT32, FLOAT64 the string shall be interpreted as defined in the chapter Float Type in the AUTOSAR EcuC specification. In case the ComSignalType is BOOLEAN the string shall be interpreted as defined in the chapter Boolean Type in the AUTOSAR EcuC specification.
 - In case the ComSignal is a UINT8_N, UINT8_DYN the string shall be interpreted as a decimal representation of the characters separated by blanks, e.g. "97 98 100" means a string "abd", where the char "a" is in byte 0(lowest address), "b" is in byte 1, and "d" is in byte 2 and (highest address). For the ComSignalType UINT8_DYN the dynamic length shall be set to the number of configured characters. An empty string "" shall be interpreted as 0-sized dynamic signal.
- 8) ComSignalEndianness
- Defines the endianness of the signal's network representation.
- 9) ComSignalInitValue
- Initial value for this signal. In case of UINT8_N the default value is a string of length ComSignalLength with all bytes set to 0x00. In case of UINT8_DYN the initial size shall be 0.
 - In case the ComSignalType is UINT8, UINT16, UINT32, UINT64, SINT8, SINT16, SINT32, SINT64 the string shall be interpreted as defined in the chapter Integer Type in the AUTOSAR EcuC specification.
 - In case the ComSignalType is FLOAT32, FLOAT64 the string shall be interpreted as defined in the chapter Float Type in the AUTOSAR EcuC specification.
 - In case the ComSignalType is BOOLEAN the string shall be interpreted as defined in the chapter Boolean Type in the AUTOSAR EcuC specification.
 - In case the ComSignal is a UINT8_N, UINT8_DYN the string shall be interpreted as a decimal representation of the characters separated by blanks, e.g. "97 98 100" means a string "abd", where the char "a" is in byte 0(lowest address), "b" is in byte 1, and "d" is in byte 2 and (highest address).
 - For the ComSignalType UINT8_DYN the dynamic length shall be set to the number of configured characters. An empty string "" shall be interpreted as 0-sized dynamic signal.
- 10) ComSignalLength
- For ComSignalType UINT8_N this parameter specifies the length n in bytes. For ComSignalType UINT8_DYN it specifies the maximum length in bytes.
 - For all other types this parameter shall be ignored. The supported maximum length is restricted by the used transportation system. For non TP-PDUs the maximum size of a PDU, and therefore also of any included signal, is limited by the concrete bus characteristic.
- 11) ComSignalType
- The AUTOSAR type of the signal. Whether or not the signal is signed or unsigned can be found by examining the value of this attribute.
 - This type could also be used to reserved appropriate storage in AUTOSAR COM.
- 12) ComTimeout
- Defines the length of the deadline monitoring timeout period in seconds.
 - The period for the first timeout period can be configured separately by ECUC_Com_00183.
- 13) ComTimeoutSubstitutionValue
- The signal substitution value will be used in case of a timeout and ComRxDataTimeoutAction is set to SUBSTITUTE.

- In case of **UINT8_N** the default value is a string of length **ComSignalLength** with all bytes set to **0x00**.
In case of **UINT8_DYN** the initial size shall be 0. In case the **ComSignalType** is **UINT8**, **UINT16**, **UINT32**, **UINT64**, **SINT8**, **SINT16**, **SINT32**, **SINT64** the string shall be interpreted as defined in the chapter **Integer Type** in the AUTOSAR EcuC specification.
- In case the **ComSignalType** is **FLOAT32**, **FLOAT64** the string shall be interpreted as defined in the chapter **Float Type** in the AUTOSAR EcuC specification.
- In case the **ComSignalType** is **BOOLEAN** the string shall be interpreted as defined in the chapter **Boolean Type** in the AUTOSAR EcuC specification.
- In case the **ComSignal** is a **UINT8_N**, **UINT8_DYN** the string shall be interpreted as a decimal representation of the characters separated by blanks, e.g. "97 98 100" means a string "abd", where the char "a" is in byte 0(lowest address), "b" is in byte 1, and "d" is in byte 2 and (highest address).
- For the **ComSignalType** **UINT8_DYN** the dynamic length shall be set to the number of configured characters. An empty string "" shall be interpreted as 0-sized dynamic signal.

14) **ComTimeoutNotification**

- On sender side: Name of **Com_CbkTxTOut** callback function to be called.
- On receiver side: Name of **Com_CbkRxTOut** callback function to be called.

15) **ComTransferProperty**

- Defines if a write access to this signal can trigger the transmission of the corresponding I-PDU.
- If the I-PDU is triggered, depends also on the transmission mode of the corresponding I-PDU.

16) **ComUpdateBitPosition**

- Bit position of update-bit inside I-PDU.
- If this attribute is omitted then there is no update-bit.
- This setting must be consistently on sender and on receiver side. Range: 0..63 for CAN and LIN, 0..511 for CAN FD, 0..2031 for FlexRay, 0..4294967295 for TP.

17) **ComSigReportIdsMEnable**

- Not supoort

18) **ComSystemTemplateSystemSignalRef**

- Reference to the **ISignalToIPduMapping** that contains a reference to the **ISignal** (System Template) which this **ComSignal** (or **ComGroupSignal**) represents.

5.12 ComConfig-ComSingalGroup

Parameter Name	Value	Category
ComDataInvalidAction	User Defined	C
ComErrorNotification	User Defined	C
ComFirstTimeout	User Defined	C
ComHandleId	Automated	F
ComInitialValueOnly		
ComInvalidNotification	User Defined	C
ComNotification	User Defined	C

Parameter Name	Value	Category
ComRxDataTimeoutAction	User Defined	C
ComTimeout	User Defined	C
ComTimeoutNotification	User Defined	C
ComTransferProperty	Automated	F
ComUpdateBitPosition	-	N
ComSystemTemplateSystemSignalGroupRef	Automated	F
ComSignalGroupArrayAccess	User Defined	C

1) ComDataInvalidAction

- This parameter defines the action performed upon reception of an invalid signal.
- Relating to signal groups the action in case if one of the included signals is an invalid signal.
- If Replace is used the ComSignallInitValue will be used for the replacement.

2) ComErrorNotification

- Only valid on sender side: Name of Com_CbkTxErr callback function to be called.
- If this parameter is omitted no error notification shall take place.

3) ComFirstTimeout

- Defines the length of the first deadline monitoring timeout period in seconds. This timeout is used immediately after start (or restart) of the deadline monitoring service.
- The timeout period of the successive periods is configured by ECUC_Com_00263.

4) ComHandleId

- The numerical value used as the ID.
- This ID identifies signals and signal groups in the COM APIs using Com_SignalIdType or Com_SignalGroupIdType parameter respectively.

5) ComInitialValueOnly

- This parameter defines that the respective signal's initial value shall be put into the respective PDU but there will not be any update of the value through the RTE.
- Thus the Com implementation does not need to expect any API calls for this signal (group).

6) ComInvalidNotification

- Only valid on receiver side: Name of Com_CbkInv callback function to be called.
- Name of the function which notifies the RTE about the reception of

7) ComTransferProperty

- On sender side: Name of Com_CbkTxAck callback function to be called.
- On receiver side: Name of Com_CbkRxAck callback function to be called.

8) ComUpdateBitPosition

- Bit position of update-bit inside I-PDU.
- If this attribute is omitted then there is no update-bit.

- This setting must be consistently on sender and on receiver side. Range: 0..63 for CAN and LIN, 0..511 for CAN FD, 0..2031 for FlexRay, 0..4294967295 for TP.
- 9) ComSystemTemplateSystemSignalGroupRef

 - Reference to the ISignalToIPduMapping that contains a reference to the ISignalGroup (SystemTemplate) which this ComSignalGroup represents.

- 10) ComSignalGroupArrayAccess

 - Defines whether the uint8-array based access shall be used for this ComSignalGroup.

5.13 ComConfig-ComSingalGroup-ComGroupSignal

Parameter Name	Value	Category
ComBitPosition	Automated	F
ComBitSize	Automated	F
ComHandleId	Automated	F
ComSignalDataInvalidValue	User Defined	C
ComSignalEndianness	Automated	F
ComSignalInitValue	Automated	C
ComSignalLength	Automated	F
ComSignalType	Automated	F
ComTransferProperty	Automated	F
ComSystemTemplateSystemSignalRef	Automated	F

- 1) ComBitPosition
 - Starting position within the I-PDU. This parameter refers to the position in the I-PDU and not in the shadow buffer. If the endianness conversion is configured to Opaque the parameter ComBitPosition shall define the bit0 of the first byte like in little endian byte order
- 2) ComBitSize
 - Size in bits, for integer signal types.
 - For ComSignalType UINT8_N and UINT8_DYN the size shall be configured by ComSignalLength.
 - For ComSignalTypes FLOAT32 and FLOAT64 the size is already defined by the signal type and therefore may be omitted.
- 3) ComHandleId
 - The numerical value used as the ID.
 - This ID identifies signals and signal groups in the COM APIs using Com_SignalIdType or Com_SignalGroupIdType parameter respectively.
- 4) ComSignalDataInvalidValue

- Defines the data invalid value of the signal.
 - In case the ComSignalType is UINT8, UINT16, UINT32, UINT64, SINT8, SINT16, SINT32, SINT64 the string shall be interpreted as defined in the chapter Integer Type in the AUTOSAR EcuC specification.
 - In case the ComSignalType is FLOAT32, FLOAT64 the string shall be interpreted as defined in the chapter Float Type in the AUTOSAR EcuC specification. In case the ComSignalType is BOOLEAN the string shall be interpreted as defined in the chapter Boolean Type in the AUTOSAR EcuC specification.
 - In case the ComSignal is a UINT8_N, UINT8_DYN the string shall be interpreted as a decimal representation of the characters separated by blanks, e.g. "97 98 100" means a string "abd", where the char "a" is in byte 0(lowest address), "b" is in byte 1, and "d" is in byte 2 and (highest address). For the ComSignalType UINT8_DYN the dynamic length shall be set to the number of configured characters. An empty string "" shall be interpreted as 0-sized dynamic signal.
- 5) ComSignalEndianness
- Defines the endianness of the signal's network representation.
- 6) ComSignalInitValue
- Initial value for this signal. In case of UINT8_N the default value is a string of length ComSignalLength with all bytes set to 0x00. In case of UINT8_DYN the initial size shall be 0.
 - In case the ComSignalType is UINT8, UINT16, UINT32, UINT64, SINT8, SINT16, SINT32, SINT64 the string shall be interpreted as defined in the chapter Integer Type in the AUTOSAR EcuC specification.
 - In case the ComSignalType is FLOAT32, FLOAT64 the string shall be interpreted as defined in the chapter Float Type in the AUTOSAR EcuC specification.
 - In case the ComSignalType is BOOLEAN the string shall be interpreted as defined in the chapter Boolean Type in the AUTOSAR EcuC specification.
 - In case the ComSignal is a UINT8_N, UINT8_DYN the string shall be interpreted as a decimal representation of the characters separated by blanks, e.g. "97 98 100" means a string "abd", where the char "a" is in byte 0(lowest address), "b" is in byte 1, and "d" is in byte 2 and (highest address).
 - For the ComSignalType UINT8_DYN the dynamic length shall be set to the number of configured characters. An empty string "" shall be interpreted as 0-sized dynamic signal.
- 7) ComSignalLength
- For ComSignalType UINT8_N this parameter specifies the length n in bytes. For ComSignalType UINT8_DYN it specifies the maximum length in bytes.
 - For all other types this parameter shall be ignored. The supported maximum length is restricted by the used transportation system. For non TP-PDUs the maximum size of a PDU, and therefore also of any included signal, is limited by the concrete bus characteristic.
- 8) ComSignalType
- The AUTOSAR type of the signal. Whether or not the signal is signed or unsigned can be found by examining the value of this attribute.
 - This type could also be used to reserved appropriate storage in AUTOSAR COM.
- 9) ComSystemTemplateSystemSignalRef
- Reference to the ISignalToIPduMapping that contains a reference to the ISignal (System Template) which this ComSignal (or ComGroupSignal) represents.

5.13 ComConfig-ComGwMapping

Signal Gateway

5.14 ComGwMapping-ComGwSource

Parameter Name	Value	Category
ComGwSignal	Automated	F
ComGwSourceDescription	Automated	F

5.15 ComGwMapping-ComGwSource-GwSignal

Parameter Name	Value	Category
ComGwSignalRef	Automated	F

5.16 ComGwMapping-ComGwSource-ComGwSourceDescription

Parameter Name	Value	Category
ComBitPosition	Automated	F
ComBitSize	Automated	F
ComSignalEndianness		
ComSignalLength		
ComSignalType		
ComUpdateBitPosition		
ComGwIPduRef		

5.17 ComGwMapping-ComGwDestiantion

Parameter Name	Value	Category
ComGwSignal	Automated	F
ComGwSourceDescription	Automated	F

5.18 ComGwMapping-ComGwDestiantion- ComGwDestinationDescription

Parameter Name	Value	Category
ComBitPosition	Automated	F
ComBitSize	Automated	F
ComSignalEndianness		
ComSignalLength		

Parameter Name	Value	Category
ComSignalType		
ComUpdateBitPosition		
ComGwIPduRef		

5.19 ComGwMapping-ComGwDestiantion-GwSignal

Parameter Name	Value	Category
ComGwSignalRef	Automated	F

6 Application Programming Interface (API)

6.1 Type Definitions

None

6.2 Macro Constants

None

6.3 Functions Definition

6.3.1 Initialization

Note: The behavior of the AUTOSAR COM module is unspecified until a correct call to Com_Init is made.

The AUTOSAR COM module's function Com_DelInit shall stop all started I-PDU groups.

Service name:	Com_Init
Syntax:	void Com_Init(const Com_ConfigType* config)
Service ID[hex]:	0x01
Sync/Async:	Synchronous
Reentrancy:	Non Reentrant
Parameters (in):	config Pointer to the AUTOSAR COM module's configuration data.
Parameters (in-out):	None
Parameters (out):	None
Return value:	None
Description:	This service initializes internal and external interfaces and variables of the AUTOSAR COM module layer for the further processing. After calling this function the inter-ECU communication is still disabled.
Available via:	Com.h

Service name:	Com_Delnit	
Syntax:	void Com_Delnit(void)	
Service ID[hex]:	0x02	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	None	
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	None	
Description:	This service stops the inter-ECU communication. All started I-PDU groups are stopped and have to be started again, if needed, after Com_Init is called. By a call to Com_Delnit the AUTOSAR COM module is put into an not initialized state.	
Available via:	Com.h	

Service name:	Com_IpduGroupStart	
Syntax:	void Com_IpduGroupStart(Com_IpduGroupIdType IpduGroupId, boolean initialize)	
Service ID[hex]:	0x03	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant for different I-PDU groups. Non reentrant for the same I-PDU group.	
Parameters (in):	IpduGroupId	Id of I-PDU group to be started
initialize	flag to request initialization of the data in the I-PDUs of this I-PDU group	
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	None	
Description:	Starts a preconfigured I-PDU group. For example, cyclic I-PDUs will be sent out cyclically after the call of Com_IpduGroupStart(). If Initialize is true all I-PDUs of the I-PDU group shall be (re-)initialized before the I-PDU group is started. That is they shall behave like after a start-up of COM, for example the old_value of the filter objects and shadow buffers of signal groups have to be (re-)initialized.	
Available via:	Com.h	

Service name:	Com_IpduGroupStop	
Syntax:	void Com_IpduGroupStop(Com_IpduGroupIdType IpduGroupId)	
Service ID[hex]:	0x04	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant for different I-PDU groups. Non reentrant for the same I-PDU group.	
Parameters (in):	IpduGroupId	Id of I-PDU group to be stopped

Parameters (in-out):	None	
Parameters (out):	None	
Return value:	None	
Description:	Stops a preconfigured I-PDU group. For example, cyclic I-PDUs will be stopped after the call of Com_IpduGroupStop().	
Available via:	Com.h	

Service name:	Com_EnableReceptionDM	
Syntax:	void Com_EnableReceptionDM(Com_IpduGroupIdType IpduGroupId)	
Service ID[hex]:	0x06	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant for different I-PDU groups. Non reentrant for the same I-PDU group.	
Parameters (in):	IpduGroupId	Id of I-PDU group where reception DM shall be enabled.
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	None	
Description:	Enables the reception deadline monitoring for the I-PDUs within the given I-PDU group.	
Available via:	Com.h	

Service name:	Com_DisableReceptionDM	
Syntax:	void Com_DisableReceptionDM(Com_IpduGroupIdType IpduGroupId)	
Service ID[hex]:	0x05	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant for different I-PDU groups. Non reentrant for the same I-PDU group.	
Parameters (in):	IpduGroupId	Id of I-PDU group where reception DM shall be disabled.
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	None	
Description:	Disables the reception deadline monitoring for the I-PDUs within the given I-PDU group.	
Available via:	Com.h	

Service name:	Com_GetStatus	
Syntax:	Com_StatusType Com_GetStatus(void)	
Service ID[hex]:	0x07	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	None	
Parameters (in-out):	None	

Parameters (out):	None	
Return value:	Com_StatusType	COM_UNINIT: the AUTOSAR COM module is not initialized and not usable COM_INIT: the AUTOSAR COM module is initialized and usable
Description:	Returns the status	

Service name:	Com_GetVersionInfo	
Syntax:	void Com_GetVersionInfo(Std_VersionInfoType* versioninfo)	
Service ID[hex]:	0x09	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	None	
Parameters (in-out):	None	
Parameters (out):	versioninfo	Pointer to where to store the version information of this module.
Return value:	None	
Description:	Returns the version information of this module.	
Available via:	Com.h	

6.3.2 Communication Services

Service name:	Com_SendSignal	
Syntax:	uint8 Com_SendSignal(Com_SignalIdType SignalId, const void* SignalDataPtr)	
Service ID[hex]:	0x0a	
Sync/Async:	Asynchronous	
Reentrancy:	Non Reentrant for the same signal. Reentrant for different signals.	
Parameters (in):	SignalId	Id of signal to be sent.
	SignalDataPtr	Reference to the signal data to be transmitted.
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	uint8	E_OK: service has been accepted COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group was stopped (or service failed due to development error) COM_BUSY: in case the TP-Buffer is locked for large data types handling
Description:	The service Com_SendSignal updates the signal object identified by SignalId with the signal referenced by the SignalDataPtr parameter.	
Available via:	Com.h	

Service name:	Com_SendDynSignal
----------------------	-------------------

Syntax:	uint8 Com_SendDynSignal(Com_SignalIdType SignalId, const void* SignalDataPtr, uint16 Length)	
Service ID[hex]:	0x21	
Sync/Async:	Asynchronous	
Reentrancy:	Non Reentrant for the same signal. Reentrant for different signals.	
Parameters (in):	SignalId Id of signal to be sent. SignalDataPtr Reference to the signal data to be transmitted. Length Length of the dynamic length signal	
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	uint8	E_OK: service has been accepted E_NOT_OK: in case the Length is greater than the configured ComSignalLength of this sent signal COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group was stopped (or service failed due to development error) COM_BUSY: in case the TP-Buffer is locked
Description:	The service Com_SendDynSignal updates the signal object identified by SignalId with the signal	

Service name:	Com_ReceiveSignal	
Syntax:	uint8 Com_ReceiveSignal(Com_SignalIdType SignalId, void* SignalDataPtr)	
Service ID[hex]:	0x0b	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant for the same signal. Reentrant for different signals.	
Parameters (in):	SignalId Id of signal to be received.	
Parameters (in-out):	None	
Parameters (out):	SignalDataPtr	Reference to the location where the received signal data shall be stored
Return value:	uint8	E_OK: service has been accepted COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group was stopped (or service failed due to development error) COM_BUSY: in case the TP-Buffer is locked for large data types handling
Description:	Com_ReceiveSignal copies the data of the signal identified by SignalId to the location specified	

Service name:	Com_ReceiveDynSignal	
Syntax:	uint8 Com_ReceiveDynSignal(Com_SignalIdType SignalId, void* SignalDataPtr, uint16* Length)	
Service ID[hex]:	0x22	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant for the same signal. Reentrant for different signals.	
Parameters (in):	SignalId Id of signal to be received.	

Parameters (in-out):	Length	in: maximum length that could be received out: length of the dynamic length signal
Parameters (out):	SignalDataPtr	reference to the location where the received signal data shall be stored
Return value:	uint8	E_OK: service has been accepted E_NOT_OK: in case the Length (as in-parameter) is smaller than the received length of the dynamic length signal COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group was stopped (or service failed due to development error) COM_BUSY: in case the TP-Buffer is locked
Description:	Com_ReceiveDynSignal copies the data of the signal identified by SignalId to the location specified by SignalDataPtr and stores the length of the dynamical length signal at the position given by the Length parameter.	
Available via:	Com.h	

Service name:	Com_SendSignalGroup	
Syntax:	uint8 Com_SendSignalGroup(Com_SignalGroupIdType SignalGroupId)	
Service ID[hex]:	0x0d	
Sync/Async:	Asynchronous	
Reentrancy:	Non Reentrant for the same group signal. Reentrant for different group signals.	
Parameters (in):	SignalGroupId	Id of signal group to be sent.
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	uint8	E_OK: service has been accepted COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group was stopped (or service failed due to development error) COM_BUSY: in case the TP-Buffer is locked for large data types handling
Description:	The service Com_SendSignalGroup copies the content of the associated shadow buffer to the associated I-PDU.	
Available via:	Com.h	

Service name:	Com_ReceiveSignalGroup	
Syntax	uint8 Com_ReceiveSignalGroup(Com_SignalGroupIdType SignalGroupId)	
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	uint8	E_OK: service has been accepted COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group was stopped (or service failed due to development error) COM_BUSY: in case the TP-Buffer is locked for large data types handling
Description:	The service Com_ReceiveSignalGroup copies the received signal group from the I-PDU to the shadow buffer.	

Available via:	Com.h	
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	uint8	E_OK: service has been accepted COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group was stopped (or service failed due to development error) COM_BUSY: in case the TP-Buffer is locked for large data types handling
Description:	The service Com_ReceiveSignalGroup copies the received signal group from the I-PDU to the shadow buffer.	
Available via:	Com.h	

Service name:	Com_SendSignalGroupArray	
Syntax:	uint8 Com_SendSignalGroupArray(Com_SignalGroupIdType SignalGroupId, const uint8* SignalGroupArrayPtr)	
Service ID[hex]:	0x23	
Sync/Async:	Asynchronous	
Reentrancy:	Non Reentrant for the same signal group. Reentrant for different signal groups.	
Parameters (in):	SignalGroupId	Id of signal group to be sent.
SignalGroupArrayPtr	Reference to the signal group array.	
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	uint8	E_OK: service has been accepted COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group was stopped (or service failed due to development error) COM_BUSY: in case the TP-Buffer is locked for large data types handling
Description:	The service Com_SendSignalGroupArray copies the content of the provided SignalGroupArrayPtr to the associated I-PDU. The provided data shall correspond to the array representation of the signal group.	
Available via:	Com.h	

Service name:	Com_ReceiveSignalGroupArray	
Syntax:	uint8 Com_ReceiveSignalGroupArray(Com_SignalGroupIdType SignalGroupId, uint8* SignalGroupArrayPtr)	
Service ID[hex]:	0x24	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant for the same signal group. Reentrant for different signal groups.	
Parameters (in):	SignalGroupId	Id of signal group to be received.
Parameters (in-out):	None	
Parameters (out):	SignalGroupArrayPtr	reference to the location where the received signal group array shall be stored

Return value:	uint8	E_OK: service has been accepted COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group was stopped (or service failed due to development error) COM_BUSY: in case the TP-Buffer is locked for large data types handling
Description:	The service Com_ReceiveSignalGroupArray copies the received signal group array representation from the PDU to the location designated by SignalGroupAr-rayPtr.	
Available via:	Com.h	

Service name:	Com_InvalidateSignal	
Syntax:	uint8 Com_InvalidateSignal(Com_SignalIdType SignalId)	
Service ID[hex]:	0x10	
Sync/Async:	Asynchronous	
Reentrancy:	Non Reentrant for the same signal. Reentrant for different signals.	
Parameters (in):	SignalId	Id of signal to be invalidated.
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	uint8	E_OK: service has been accepted COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group is stopped, no ComSignalDataInvalidValue is configured for the given signalId or service fails due to development error COM_BUSY: in case the TP-Buffer is locked
Description:	The service Com_InvalidateSignal invalidates the signal with the given SignalId by setting its value to its configured ComSignalDataInvalidValue.	
Available via:	Com.h	

Service name:	Com_InvalidateSignalGroup	
Syntax:	uint8 Com_InvalidateSignalGroup(Com_SignalGroupIdType SignalGroupId)	
Service ID[hex]:	0x1b	
Sync/Async:	Asynchronous	
Reentrancy:	Non Reentrant for the same signal group. Reentrant for different signal groups.	
Parameters (in):	SignalGroupId	Id of signal group to be invalidated.
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	uint8	E_OK: service has been accepted COM_SERVICE_NOT_AVAILABLE: corresponding I-PDU group is stopped, no ComSignalDataInvalidValue is configured for any of the group signals or service fails due to

Description:	The service Com_InvalidateSignalGroup invalidates all group signals of the signal group with the given SignalGroupId by setting their values to their configured ComSignalDataInvalidValues.	
Available via:	Com.h	

Service name:	Com_TriggerIPDUSend	
Syntax:	Std_ReturnType Com_TriggerIPDUSend(PdulIdType PdulId)	
Service ID[hex]:	0x17	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	PdulId	The I-PDU-ID of the I-PDU that shall be triggered for sending
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	Std_ReturnType	E_OK: I-PDU was triggered for transmission E_NOT_OK: I-PDU is stopped, the transmission could not be triggered
Description:	By a call to Com_TriggerIPDUSend the I-PDU with the given ID is triggered for transmission.	
Available via:	Com.h	

Service name:	Com_TriggerIPDUSendWithMetaData	
Syntax:	Std_ReturnType Com_TriggerIPDUSendWithMetaData(PdulIdType PdulId, const uint8* MetaData)	
Service ID[hex]:	0x28	
Sync/Async:	Synchronous	
Reentrancy:	Non Reentrant	
Parameters (in):	PdulId	The I-PDU-ID of the I-PDU that shall be triggered for sending
	MetaData	A pointer to the metadata for the triggered send-request
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	Std_ReturnType	E_OK: I-PDU was triggered for transmission E_NOT_OK: I-PDU is stopped, the transmission could not be triggered

Description:	By a call to Com_TriggerIPDUSendWithMetaData the AUTOSAR COM module updates its internal metadata for the I-PDU with the given ID by copying the metadata from the given position and with respect to length of the globally configured MetaDataType of this I-PDU. Then the I-PDU is triggered for transmission.
Available via:	Com.h

Service name:	Com_SwitchIpduTxMode
Syntax:	void Com_SwitchIpduTxMode(PduldType Pduld, boolean Mode)
Service ID[hex]:	0x27
Sync/Async:	Synchronous
Reentrancy:	Reentrant for different Pdulds. Non reentrant for the same Pduld.
Parameters (in):	Pduld Id of the I-PDU of which the transmission mode shall be changed.
Mode	the transmission mode that shall be set.
Parameters (in-out):	None
Parameters (out):	None
Return value:	None
Description:	The service Com_SwitchIpduTxMode sets the transmission mode of the I-PDU referenced by Pduld to Mode. In case the transmission mode changes, the new mode shall immediately be effective (see SWS_Com_00239). In case the requested transmission mode was already active for this I-PDU, the call will have no effect.
Available via:	Com.h

6.3.3 Callback function

Service name:	Com_TriggerTransmit	
Syntax:	Std_ReturnType Com_TriggerTransmit(PduldType TxPduld, PduInfoType* PduInfoPtr)	
Service ID[hex]:	0x41	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant for different Pdulds. Non reentrant for the same Pduld.	
Parameters (in):	TxPduld	ID of the SDU that is requested to be transmitted.
Parameters (in-out):	PduInfoPtr	Contains a pointer to a buffer (SduDataPtr) to where the SDU data shall be copied, and the available buffer size in SduLength. On return, the service will indicate the length of the copied SDU data in SduLength.
Parameters (out):	None	

Return value:	Std_ReturnType	E_OK: SDU has been copied and SduLength indicates the number of copied bytes. E_NOT_OK: No SDU data has been copied. PduInfoPtr must not be used since it may contain a NULL pointer or point to invalid data.
Description:	Within this API, the upper layer module (called module) shall check whether the available data fits into the buffer size reported by PduInfoPtr->SduLength. If it fits, it shall copy its data into the buffer provided by PduInfoPtr->SduDataPtr and update the length of the actual copied data in PduInfoPtr->SduLength. If not, it returns E_NOT_OK without changing PduInfoPtr.	
Available via:	Com.h	

Service name:	Com_RxIndication	
Syntax:	void Com_RxIndication(PduldType RxPduld, const PduInfoType* PduInfoPtr)	
Service ID[hex]:	0x42	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant for different Pdulds. Non reentrant for the same Pduld.	
Parameters (in):	RxPduld	ID of the received PDU.
PduInfoPtr	Contains the length (SduLength) of the received PDU, a pointer to a buffer (SduDataPtr) containing the PDU, and the MetaData related to this PDU.	
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	None	
Description:	Indication of a received PDU from a lower layer communication interface module.	
Available via:	Com.h	

Service name:	Com_TpRxIndication	
Syntax:	void Com_TpRxIndication(PduldType id, Std_ReturnType result)	
Service ID[hex]:	0x45	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	id	Identification of the received I-PDU.
result	Result of the reception.	
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	None	
Description:	Called after an I-PDU has been received via the TP API, the result indicates whether the transmission was successful or not.	
Available via:	Com.h	

Service name:	Com_TxConfirmation	
Syntax:	void Com_TxConfirmation(PduldType TxPduld, Std_ReturnType result)	
Service ID[hex]:	0x40	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant for different Pdulds. Non reentrant for the same Pduld.	
Parameters (in):	TxPduld	ID of the PDU that has been transmitted.
result	E_OK: The PDU was transmitted. E_NOT_OK: Transmission of the PDU failed.	
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	None	
Description:	The lower layer communication interface module confirms the transmission of a PDU, or the failure to transmit a PDU.	
Available via:	Com.h	

Service name:	Com_TpTxConfirmation	
Syntax:	void Com_TpTxConfirmation(PduldType id, Std_ReturnType result)	
Service ID[hex]:	0x48	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	id	Identification of the transmitted I-PDU.
result	Result of the transmission of the I-PDU.	
Parameters (in-out):	None	
Parameters (out):	None	
Return value:	None	
Description:	This function is called after the I-PDU has been transmitted on its network, the result indicates whether the transmission was successful or not.	
Available via:	Com.h	

Service name:	Com_StartOfReception	
Syntax:	BufReq_ReturnType Com_StartOfReception(PduldType id, const PdulInfoType* info, PduLengthType TpSduLength, PduLengthType* bufferSizePtr)	
Service ID[hex]:	0x46	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	id	Identification of the I-PDU.

info	Pointer to a PduInfoType structure containing the payload data (without protocol information) and payload length of the first frame or single frame of a transport protocol I-PDU reception, and the MetaData related to this PDU. If neither first/single frame data nor MetaData are available, this parameter is set to NULL_PTR.	
TpSduLength	Total length of the N-SDU to be received.	
Parameters (in-out):	None	
Parameters (out):	bufferSizePtr	Available receive buffer in the receiving module. This parameter will be used to compute the Block Size (BS) in the transport protocol module.
Return value:	BufReq_ReturnType BUFREQ_OK: Connection has been accepted. bufferSizePtr indicates the available receive buffer; reception is continued. If no buffer of the requested size is available, a receive buffer size of 0 shall be indicated by bufferSizePtr. BUFREQ_E_NOT_OK: Connection has been rejected; reception is aborted. bufferSizePtr remains unchanged. BUFREQ_E_OVFL: No buffer of the required length can be provided; reception is aborted. bufferSizePtr remains unchanged.	
Description:	This function is called at the start of receiving an N-SDU. The N-SDU might be fragmented into multiple N-PDUs (FF with one or more following CFs) or might consist of a single N-PDU (SF). The service shall provide the currently available maximum buffer size when invoked with TpSduLength equal to 0.	
Available via:	Com.h	

Service name:	Com_CopyRxData	
Syntax:	BufReq_ReturnType Com_CopyRxData(PdulIdType id, const PduInfoType* info, PduLengthType* bufferSizePtr)	
Service ID[hex]:	0x44	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	id Identification of the received I-PDU.	
info	Provides the source buffer (SduDataPtr) and the number of bytes to be copied (SduLength). An SduLength of 0 can be used to query the current amount of available buffer in the	

	upper layer module. In this case, the SduDataPtr may be a NULL_PTR.	
Parameters (in-out):	None	
Parameters (out):	bufferSizePtr	Available receive buffer after data has been copied.
Return value:	BufReq_ReturnType	BUFREQ_OK: Data copied successfully BUFREQ_E_NOT_OK: Data was not copied because an error occurred.
Description:	This function is called to provide the received data of an I-PDU segment (N-PDU) to the upper layer. Each call to this function provides the next part of the I-PDU data. The size of the remaining buffer is written to the position indicated by bufferSizePtr.	
Available via:	Com.h	

Service name:	Com_CopyTxData	
Syntax:	BufReq_ReturnType Com_CopyTxData(PdulIdType id, const PdulInfoType* info, const RetryInfoType* retry, PduLengthType* availableDataPtr)	
Service ID[hex]:	0x43	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	id	Identification of the transmitted I-PDU.
info	Provides the destination buffer (SduDataPtr) and the number of bytes to be copied (SduLength). If not enough transmit data is available, no data is copied by the upper layer module and BUFREQ_E_BUSY is returned. The lower layer module may retry the call. An SduLength of 0 can be used to indicate state changes in the retry parameter or to query the current amount of available data in the upper layer module. In this case, the SduDataPtr may be a NULL_PTR.	
retry	This parameter is used to acknowledge transmitted data or to retransmit data after transmission problems. If the retry parameter is a NULL_PTR, it indicates that the transmit data can be removed from the buffer immediately after it has been copied. Otherwise, the retry parameter must point to a valid RetryInfoType element. If TpDataState indicates TP_CONFPENDING, the previously copied data must remain in the TP buffer to be available for error recovery. TP_DATACONF indicates that all data that has been copied before this call is confirmed and can be removed from the TP buffer. Data copied by this API call is excluded and will be	

	confirmed later. TP_DATARETRY indicates that this API call shall copy previously copied data in order to recover from an error. In this case TxTpDataCnt specifies the offset in bytes from the current data copy position.	
Parameters (in-out):	None	
Parameters (out):	availableDataPtr	Indicates the remaining number of bytes that are available in the upper layer module's Tx buffer. availableDataPtr can be used by TP modules that support dynamic payload lengths (e.g. FrIsoTp) to determine the size of the following CFs.
Return value:	BufReq_ReturnType	BUFREQ_OK: Data has been copied to the transmit buffer completely as requested. BUFREQ_E_BUSY: Request could not be fulfilled, because the required amount of Tx data is not available. The lower layer module may retry this call later on. No data has been copied. BUFREQ_E_NOT_OK: Data has not been copied. Request failed.
Description:	This function is called to acquire the transmit data of an I-PDU segment (N-PDU). Each call to this function provides the next part of the I-PDU data unless retry->TpDataState is TP_DATARETRY. In this case the function restarts to copy the data beginning at the offset from the current position indicated by retry->TxTpDataCnt. The size of the remaining data is written to the position indicated by availableDataPtr.	
Available via:	Com.h	

6.3.4 Schedule function

Service name:	Com_MainFunctionRx
Syntax:	void Com_MainFunctionRx(void)
Service ID[hex]:	0x18
Description:	This function performs the processing of the AUTOSAR COM module's receive processing that are not directly handled

	within the COM's functions invoked by the PDU-R, for example Com_RxIndication.
Available via:	SchM_Com.h

Service name:	Com_MainFunctionTx
Syntax:	void Com_MainFunctionTx(void)
Service ID[hex]:	0x19
Description:	This function performs the processing of the AUTOSAR COM module's transmission activities that are not directly handled within the COM's function invoked by the RTE, for example Com_SendSignal.
Available via:	SchM_Com.h

Service name:	Com_MainFunctionRouteSignals
Syntax:	void Com_MainFunctionRouteSignals(void)
Service ID[hex]:	0x1a
Description:	Calls the signal gateway part of the AUTOSAR COM module to forward received signals to be routed.
Available via:	SchM_Com.h

7. Generator

Generator option

Options	Description
-G,--Generation	Symbolic parameters to be used for fore generation (skip validation).
-H,--Help	Display this help message.
-I,--Input <I>	ECU description file path of the module for which generation tool need to run.
-L,--Log	Symbolic parameters to be used for generation error log.
-M,--Module <M>	Specify module name and version to be generated code for.
-O,--Output <O>	Project-relative path to location where the generated code is to be placed.
-T,--Top_path <T>	Symbolic parameters to be used for set path of module.
-V,--Validate	Symbolic parameters to be used for invoking validation checks.

7.1 Generator Error Message

7.1.1 Error message

The following section gives the list of error messages displayed by the Generation Tool.

7.1.1.1 ERR050001: The input arxmls are not validated against the schema. Please correct the arxml as per schema or If you need any support contact HYUNDAI AUTOEVER Co.,Ltd.

This is an Unexpected Error. On the occurrence of this error contact Hyundai-Autoever Basic Platform Development Team.

7.1.1.2 ERR050002: Unexpected Error Found. This error may be due to the incorrect configuration of the element(s) ‘Element Name’. Please correct the arxml as per schema or If you need any support contact HYUNDAI AUTOEVER Co.,Ltd.

This error occurs, if the structure fields that are to be generated in the C Source file are empty. Contact Hyundai-Autoever Basic Platform Development Team.

7.1.1.3 ERR050003: ‘Component Name’ Component is not present in the input file(s).

This error occurs, if any one of Com or EcuC or PduR or ComM component is not present in any of the input ECU Configuration Description File(s) and ECU Configuration PDF File(s).

7.1.1.4 ERR050004: The Reference path is empty for the parameter ‘Parameter Name’ in the container ‘Container Name’.

This error occurs, if no reference path is configured for the below parameters of the respective container.

Container Name	Parameter Name
ComGwSignal	ComGwSignalRef
ComIPdu	ComIPduGroupRef
	ComPdulRef
ComIPduReplication	ComIPduReplicaRef
ComIPduGroup	ComIPduGroupGroupRef
ComGwDestinationDescription /ComGwSourceDescription	ComGwlPduRef

7.1.1.5 ERR050005: Parameter ‘Parameter Name’ in the container ‘Container Name’ should be configured.

This error occurs, if value of any of the mandatory parameters mentioned in the below table are not configured.

Container Name	Parameter Name
ComIPdu	ComIPduDirection
	ComIPduSignalProcessing
	ComIPduType
ComTxIPdu	ComTxIPduUnusedAreasDefault
ComTxMode	ComTxModeMode
ComGwDestinationDescription	ComSignalEndianness
	ComBitPosition
ComGwSourceDescription	ComBitPosition
	ComSignalType
	ComSignalEndianness
ComConfig	ComConfigurationId
ComIPduReplication	ComIPduReplicationQuorum
ComIPduCounter	ComIPduCounterSize
	ComIPduCounterStartPosition
ComIPduGroup	ComIPduGroupHandleId
ComSignal	ComBitPosition
	ComHandleId
	ComSignalEndianness
	ComSignalType
ComSignalGroup	ComHandleId
	ComBitPosition
	ComHandleId

ComGroupSignal	ComSignalEndianness
	ComSignalType
	ComTransferProperty
ComGeneral	ComSupportedIPduGroups
	ComVersionInfoApi
ComGWSourceDescription	ComGWIPduRef
ComIPduReplication	ComIPduReplicaRef

7.1.1.6 ERR050013: The reference path <Reference Path> provided for the parameter ‘Parameter Name’ in the container ‘Container Name’, having short name <Short Name> is incorrect.

This error occurs, if the reference path provided for the below parameters of their respective containers are incorrect.

Container Name	Parameter Name
ComGwSignal	ComGwSignalRef
ComIPdu	ComIPduGroupRef
	ComPdulRef
ComIPduReplication	ComIPduReplicaRef
ComIPduGroup	ComIPduGroupGroupRef
ComGwDestinationDescription /ComGWSourceDescription	ComGwIPduRef

7.1.1.7 ERR050022: Value of the parameter ‘Parameter Name’ in ‘Container Name’ should not be configured as <0>.

This error occurs, if the below parameters of respective containers are configured as 0.

Container Name	Parameter Name
ComTimeBase	ComRxTimeBase
	ComTxTimeBase

	ComGwTimeBase
--	---------------

7.1.1.8 ERR050051: I-PDU cannot be included in the I-PDU Group because the direction of other I-PDUs in the I-PDU Group is different from this I-PDU.

This error occurs, if an I-PDU tries to be included in the I-PDU Group that has different direction.

7.1.1.9 ERR050052: Parameter ‘ComRxTimeBase’ in the container ‘ComTimeBase’ should not be configured since Rx I-PDUs is/are configured.

This error occurs, if parameter "ComRxTimeBase" in container "ComTimeBase" is not configured when Rx I-PDUs is/are configured.

7.1.1.10 ERR050053: Parameter ‘ComTxTimeBase’ in the container ‘ComTimeBase’ should not be configured since TX I-PDUs is/are configured.

This error occurs, if parameter "ComTxTimeBase" in container "ComTimeBase" is not configured when Tx I-PDUs is/are configured.

7.1.1.11 ERR050054: Short name of the container ‘Container Name’ should be unique.

This error occurs, if values configured for the below parameters are not unique across the below parameters/container short names:

- ComSignal
- ComSignalGroup
- ComGroupSignal
- ComIPduGroup
- ComIPdu

7.1.1.12 ERR050055: Parameter ‘ComGwTimeBase’ in the container ‘ComTimeBase’ should not be configured since Gateways is/are configured.

This error occurs, if parameter "ComGwTimeBase" in container "ComTimeBase" is not configured when Gateway is/are configured.

7.1.1.13 ERR050058: Value of the parameter ‘ComInvalidNotification’ should not be configured for the Signal, when the ‘ComDataInvalidAction’ is configured as <REPLACE>.

This error occurs, if value of the parameter "ComDataInvalidAction" is configured as "REPLACE" and value of the parameter "ComInvalidNotification" is configured in the container "ComSignal".

7.1.1.14 ERR050059: Value for the parameter ‘ComTransferProperty’ should be configured for TX Signals.

This error occurs, if the parameter "ComTransferProperty" is not configured for Tx Signals.

7.1.1.15 ERR050060: Value of parameter ‘ComFilterAlgorithm’ should not be equal to <NEW_IS_WITHIN/NEW_IS_OUTSIDE>, since the ‘ComSignalType’ in the container ‘ComSignal’ is configured as <BOOLEAN >.

This error occurs, when at least one of the following filter types in "ComFilter" container is not configured as:

- ALWAYS
- NEVER
- MASKED_NEW_EQUALS_X
- MASKED_NEW_DIFFERS_X
- MASKED_NEW_DIFFERS_MASKED_OLD
- ONE_EVERY_N

When parameter "ComSignalType" is configured as "BOOLEAN" in container "ComSignal".

(ComFilterAlgorithm is to be configured as NEW_IS_WITHIN/NEW_IS_OUTSIDE).

7.1.1.16 ERR050062: Parameter ‘ComBitPosition’ should be multiple of 8, since parameter ‘ComSignalEndianess’ is equal to <OPAQUE> in container ‘ComSignal’.

This error occurs, if parameter "ComSignalEndianess" is configured as "OPAQUE" and parameter "ComBitPosition" is not multiple of 8 in container "ComSignal".

7.1.1.17 ERR050063: Parameter ‘ComBitSize’ should be in the range of <0 or \$min - \$max>, since value of the parameter ‘ComSignalType’ in the container ‘ComSignal’ is configured as <signal type>.

This error occurs, if parameter ComBitSize is not consistent with parameter ComSignalType in the container ComSignal.

ComSignalType	ComBitSize
UINT8	0 or (0-8)
UINT16	0 or (9-16)
UINT32	0 or (17-32)

7.1.1.18 ERR050064: For the Signal, parameter ‘ComSignalDataInvalidValue’ should lie within the range of (0 - 2 ** BitSize) since Signal Size is < ComBitSize >.

This error occurs, if value of the parameter "ComSignalDataInvalidValue" of the container "ComSignal" does not lie within the range (0 - 2 ** BitSize), since value of the parameter ComBitSize is configured.

7.1.1.19 ERR050065: I-PDU Group cannot be nested into the I-PDU group which has a different direction.

This error occurs, if an I-PDU Group tries to contain I-PDU group that has a different direction.

7.1.1.20 ERR050066: For the Signal, ‘ComFilter’ should not be configured, since ‘ComSignalType’ is configured to <FLOAT32/FLOAT64>.

This error occurs, if parameter "ComSignalType" in container "ComSignal" is configured as "FLOAT32"/"FLOAT64" and "ComFilter" sub container is configured.

7.1.1.21 ERR050069: For the GroupSignal, ‘ComFilter’ should not be configured, since ‘ComSignalType’ is configured to < FLOAT32/FLOAT64>.

This error occurs, if "ComSignalType" is configured as "FLOAT32"/"FLOAT64" in container "ComGroupSignal" and sub-container "ComFilter" is configured.

7.1.1.22 ERR050070: Parameter ‘ComIPduCounterThreshold’ and ‘ComIPduCounterErrorNotification’ in the container ‘ComIPduCounter’ should not be configured for TX I-PDU.

This error occurs, if parameters "ComIPduCounterThreshold" and "ComIPduCounterErrorNotification" in container "ComIPduCounter" are configured for Tx Ipdu's.

7.1.1.23 ERR050071: Counter Start Position of I-PDU does not fit within an I-PDU. Valid range is <0 to (IpduSize * 8 – 1)>.

This error occurs, if parameter "ComIPduCounterStartPosition" in container "ComIPduCounter" does not lie within I-PDU.

7.1.1.24 ERR050072: For the Signal parameter ‘ComSignalType’ should be equal to <UINT8_N/UINT8_DYN> since ‘ComSignalEndianness’ is equal to <OPAQUE> for TX Signals.

This error occurs, if parameter "ComSignalType" is not equal to "UINT8_N"/"UINT8_DYN" and parameter "ComSignalEndianness" is configured as "OPAQUE" in container "ComSignal" for Tx Signals.

7.1.1.25 ERR050073: Counter Start Position of I-PDU is overlapping with Update Bit or Counter Position or Signal in I-PDU.

This error occurs, if parameter "ComIPduCounterStartPosition" in container "ComIPduCounter" does not overlap with Signal/update bit position of Signal/update bit position of Signal Group/Signal group/group Signal.

7.1.1.26 ERR050074: The Signal belongs to more than one I-PDU. The Signal should belong to only one I-PDU.

This error occurs, if Signal is mapped to more than one I-PDU.

7.1.1.27 ERR050075: For the Rx SignalGroup, ‘ComFirstTimeout/ComTimeout’ should be configured when value of the parameter ‘ComRxDataTimeoutAction’ is configured.

This error occurs, if value of the parameter "ComFirstTimeout" and/or "ComTimeout" is not configured, when value of the parameter "ComRxDataTimeoutAction" in container ComSignalGroup is configured for Rx Signal Group.

7.1.1.28 ERR050076: The Containers ‘ComTxModeTrue’ and ‘ComTxModeFalse’ needs to be configured for I-PDUs, when signal is configured for ‘ComFilter’ with algorithm as other than <ALWAYS> or <NEVER>.

This error occurs, if in an I-PDU, one of the container "ComTxModeTrue" or container "ComTxModeFalse" is not configured and container "ComSignal" is configured with parameter "ComFilterAlgorithm" as other than ALWAYS or NEVER in container "ComFilter".

7.1.1.29 ERR050077: Value of the parameter ‘ComInvalidNotification’ should be configured, when the ‘ComDataInvalidAction’ is configured as <NOTIFY> for Signals.

This error occurs, if value of the parameter "ComDataInvalidAction" is configured as "NOTIFY" and value of the parameter "ComInvalidNotification" is not configured in the container "ComSignal".

7.1.1.30 ERR050078: Value of the parameter ‘ComIPduReplicationQuorum’ should not be greater than (multiplicity of the parameter ‘ComIPduReplicaRef’ + 1) in the container ‘ComIPduReplication’.

This error occurs, if parameter "ComIPduReplicationQuorum" value in container "ComIPduReplication" is not less than or equal to multiplicity of the parameter (ComIPduReplicaRef + 1).

7.1.1.31 ERR050079: Value for the parameter ‘ComInvalidNotification’ should be configured, since the parameter ‘ComDataInvalidAction’ is configured as <NOTIFY> for SignalGroup.

This error occurs, if parameter "ComDataInvalidAction" is configured as "NOTIFY" and parameter "ComInvalidNotification" is not configured in container "ComSignalGroup".

7.1.1.32 ERR050080: Container ‘ComIPduCounter’ should be configured when container ‘ComIPduReplication’ is configured for Rx I-PDU.

This error occurs, if container "ComIPduCounter" is not configured for Rx IPdus when container "ComIPduReplication" is configured.

7.1.1.33 ERR050081: Value of the parameter ‘ComTransferProperty’ should be configured in the container TX ‘ComSignalGroup’ .

This error occurs, if the parameter “ComTransferProperty” is not configured for Tx Signal Groups.

7.1.1.34 ERR050082: Value of parameter ‘ComFilterAlgorithm’ should not be equal to <NEW_IS_WITHIN/NEW_IS_OUTSIDE>, since the ‘ComSignalType’ in the container ‘GroupSignal’ is configured as <BOOLEAN>.

This error occurs, if value of the parameter "ComFilterAlgorithm" in sub-container "ComFilter" is configured as "NEW_IS_WITHIN"/"NEW_IS_OUTSIDE" and parameter "ComSignalType" in container "ComGroupSignal" is configured as BOOLEAN.

7.1.1.35 ERR050083: For the GroupSignal Parameter ‘ComBitSize’ should be greater than one, since Signal type is SINT8.

This error occurs, if the value of the parameter “ComBitSize” is less than 1, when Signal type is configured as “SINT8”.

7.1.1.36 ERR050084: An I-PDU Group must not contain itself.

This error occurs, if an I-PDU Group contains itself.

7.1.1.37 ERR050085: Parameter ‘ComSignalDataInvalidValue’ in the container ‘ComGroupSignal’ should be in the range of <-1.175494351E-38 to -3.402823466E+38>, since value of the ‘ComSignalType’ is configured as < FLOAT32 >.

This error occurs, if parameter "ComSignalType" is "FLOAT32" and parameter "ComSignalDataInvalidValue" is not in between the range of (-1.175494351E-38 to -3.402823466E+38) or parameter "ComSignalDataInvalidValue" is not equal to zero in container "ComGroupSignal".

7.1.1.38 ERR050086: For the GroupSignal, parameter ‘ComBitSize’ should range between <1/1, 1/8, 9/16, 17/32>, since Signal Type is <BOOLEAN, UINT8/SINT8, UINT16/SINT16, UINT32/SINT32>.

This error occurs, if parameters "ComBitSize" and "ComSignalType" (except for `UINT8_N/FLOAT32/FLOAT64`) are not consistent with each other in container "ComGroupSignal".

7.1.1.39 ERR050087: The parameters ‘ComTxModeTimeOffset’ and ‘ComTxModeTimePeriod’ needs to be configured, when parameter ‘ComTxModeMode’ is configured for <PERIODIC or MIXED>.

This error occurs, if the parameters "ComTxModeTimeOffset" and "ComTxModeTimePeriod" are not configured when the "ComTxModeMode" is configured as either "PERIODIC" or "MIXED" in the "ComTxMode" container.

7.1.1.40 ERR050088: Value of the parameter ‘ComIPduCounterThreshold’ in the container ‘ComIPduCounter’ should be configured as <0>, since container ‘ComIPduReplication’ is configured for Rx I-PDU.

This error occurs, if parameter "ComIPduCounterThreshold" in container "ComIPduCounter" is not configured as 0 for Rx IPdus when container "ComIPduReplication" is configured.

7.1.1.40 ERR050089: Value of the parameter ‘ComIPduCounterThreshold’ should be in consistent with value of the parameter ‘ComIPduCounterSize’ in the container ‘ComIPduCounter’ for Rx I-PDU since I-PDU Replication is not configured.

This error occurs, when parameter "ComIPduCounterThreshold" is not consistent with parameter "ComIPduCounterSize" in container "ComIPduCounter" for Rx I-PDU since container "ComIPduReplication" is configured.

7.1.1.41 ERR050090: The parameter 'ComTxModeRepetitionPeriod' should be configured, when parameter 'ComTxModeNumberOfRepetition' is greater than zero.

This error occurs, if value of the parameter "ComTxModeRepetitionPeriod" is not configured when value of the parameter "ComTxModeNumberOfRepetitions" is greater than 0.

7.1.1.42 ERR050091: For Signal, parameter ‘Filter Parameters’ should be configured, since ‘ComFilterAlgorithm’ is configured to ‘Algorithm’.

This error occurs, if any of the required filter parameters for corresponding filter algorithm is not configured as mentioned below.

Algorithm/Filter Parameters	ComFilter Mask	ComFilter Max	ComFilterMin	ComFilterOffset	ComFilterPeriodFactor	ComFilterX
ALWAYS						
NEVER						
MASKED_NEW_DIFFERS_MASKED_OLD	X					
MASKED_NEW_DIFFERS_X	X					X
MASKED_NEW_EQUALS_X	X					X
NEW_IS_OUTSIDE		X	X			
NEW_IS_WITHIN		X	X			
ONE_EVERY_N				X	X	

Note: In the above table only crossed parameters need to be configured for the filter algorithm.

7.1.1.43 ERR050092: For the Signal parameter ‘ComFilterOffset’ should be less than parameter ‘ComFilterPeriodFactor’ in container ‘ComFilter’.

This error occurs, if value of parameter "ComFilterOffset" is not less than the value of the parameter "ComFilterPeriodFactor" in container "ComFilter".

7.1.1.44 ERR050093: Signal does not fit within an I-PDU.

This error occurs, if Signal does not lie within an I-PDU.

7.1.1.45 ERR050094: Group Signal does not fit within the I-PDU.

This error occurs, if Group signal does not lie within the I-PDU.

7.1.1.46 ERR050095: Signals/GroupSignals should not overlap each other.

This error occurs, if Signals/Group Signals are overlapping each other.

7.1.1.47 ERR050096: Update Bit of Signal does not fit within the IPDU". Valid range is <0 to (IpduSize* 8 - 1)>.

This error occurs, if update bit of the Signal does not lie within an I-PDU.

7.1.1.48 ERR050097: Update Bit of Signal should not overlap with Update Bit or Signal in I-PDU.

This error occurs, if update bit of the Signal is overlapping with update bit or with Signal.

7.1.1.49 ERR050098: The Parameter 'ComTxTimeBase' in container 'ComTimeBase' should be configured, since parameter 'ComTxModeRepetitionPeriod/ ComTxModeTimeOffset/ ComTxModeTimePeriod' in the container 'ComTxMode' is configured.

This error occurs, if parameter "ComTxModeRepetitionPeriod"/"ComTxModeTimeOffset"/"ComTxModeTimePeriod" in the container "ComTxMode" is configured and parameter "ComTxTimeBase" in container "ComTimeBase" is not configured.

7.1.1.50 ERR050099: The Parameter 'ComTxTimeBase' in container 'ComTimeBase' should be configured, since parameter 'ComMinimumDelayTime' in the container 'ComTxlpdu' is configured.

This error occurs, if parameter "ComMinimumDelayTime" in container "ComTxlpdu" is configured and parameter "ComTxTimeBase" in container "ComTimeBase" is not configured.

7.1.1.51 ERR050100: The Parameters ‘ComTxTimeBase/ ComRxTimeBase’ in container ‘ComTimeBase’ should be configured, since parameter ‘ComFirstTimeout/ ComTimeout’ in the container ‘ComSignal’ is configured.

This error occurs, if parameter "ComFirstTimeout"/"ComTimeout" in container "ComSignal" is configured and parameter "ComRxTimeBase" in container "ComTimeBase" is not configured.

7.1.1.52 ERR050101: The RX SignalGroup should contain mapping to TX SignalGroup for the GateWay mapping container.

This error occurs, if receive Signal Group is not pointing to transmit Signal Group.

7.1.1.53 ERR050102: The Parameter ‘ComTxTimeBase’ in container ‘ComTimeBase’ should be configured, since parameter ‘ComFirstTimeout/ ComTimeout’ in the container ‘ComSignalGroup’ are configured.

This error occurs, if parameter "ComFirstTimeout"/"ComTimeout" in the container "ComSignalGroup" is configured and parameter "ComTxTimeBase" in container "ComTimeBase" is not configured.

7.1.1.54 ERR050104: Signal size and Signal type should be same for source and destination Signals in the GateWay mapping container.

This error occurs, if Signal size and Signal type are not same for source and destination Signals.

7.1.1.55 ERR050107: Value of the parameter ‘ComHandleId’ should start from <0> and should be sequential for Tx Group Signals.

This error occurs, if Tx Group Signals are not sequential and does not start from 0 when Tx Signals are not configured.

7.1.1.56 ERR050109: Value of the parameter ‘ComHandleId’ should start from <0> and should be sequential for Rx Group Signals.

This error occurs, if Rx Group Signals are not sequential and does not start from 0 when Rx Signals are not configured.

7.1.1.57 ERR050111: Parameter ‘ComIPduHandleId’ is not configured for the following Rx Ipdu(s) in the container ‘ComIPdu’.

This error occurs, if the parameter "ComIPduHandleId" of the container "ComIPdu" for Rx I-PDU are not configured within the configuration set.

7.1.1.58 ERR050112: The parameter ‘ComIPduHandleId’ in the container ‘ComIPdu’ should be unique for TX I-PDUs.

This error occurs, if Tx I-PDU HandleIds are not unique within the configuration.

7.1.1.59 ERR050113: The parameter ‘ComIPduHandleId’ in the container ‘ComIPdu’ should start from <0> and should be sequential for TX I-PDUs.

This error occurs, if Tx I-PDU HandleIds does not start from 0 and not sequential within the configuration.

7.1.1.60 ERR050114: The Parameter ‘PduLength’ in ECUC Module should have a maximum length of ‘4095’ bytes, since parameter ‘ComIPduType’ in the container ‘ComIPdu’ is configured for ‘TP’.

This error occurs, if "PduLength" in ECUC module is configured greater than 4095 bytes and parameter 'ComIPduType' in container "ComIPdu" is configured for "TP".

7.1.1.61 ERR050115: Parameter ‘ComSignalType’ in the container ‘ComGroupSignal’ should not be configured as ‘UINT8_DYN’.

This error occurs, if parameter "ComSignalType" in container 'ComGroupSignal' is configured as "UINT8_DYN".

7.1.1.62 ERR050116: Dynamic Signals should always be present at the end of the I-PDU.

This error occurs, if Dynamic Signals are not present at the end of the I-PDU.

7.1.1.63 ERR050117: 'ComIpduReplication' should not be configured, since parameter 'ComIpduType' in container 'ComIpdu' is configured as <TP>.

This error occurs, if container "ComIPduReplication" is configured and parameter "ComIPduType" in container "ComIPdu" is configured as <TP> only for Rx I-PDUs.

7.1.1.64 ERR050118: Parameter 'ComSignalLength' should be configured, since parameter 'ComSignalType' in the container 'ComSignal' is configured as UINT8_N/ UINT8_DYN.

This error occurs, if parameter "ComSignalType" is configured as "UINT8_N" or "UINT8_DYN" and parameter "ComSignalLength" is not configured in container "ComSignal".

7.1.1.65 ERR050119: Parameter 'ComSignalLength' should be configured, since parameter 'ComSignalType' in the container 'ComGroupSignal' is configured as UINT8_N.

This error occurs, if parameter "ComSignalType" is configured as "UINT8_N" and "ComSignalLength" is not configured in container "ComGroupSignal".

7.1.1.66 ERR050120: Parameter 'ComBitSize' should be configured, since the 'ComSignalType' in the container 'ComSignal' is configured as UINT8_N/ UINT8_DYN/ FLOAT32/ FLOAT64.

This error occurs, if parameter "ComSignalType" is configured other than "UINT8_N" or "UINT8_DYN" or "FLOAT32" or "FLOAT64" and "ComBitSize" is not configured in container "ComSignal".

7.1.1.67 ERR050121: The parameter 'ComHandleId' in the container 'ComSignal' should be unique for RX Signals.

This error occurs, if Rx Signal HandleIds are not unique within the configuration.

7.1.1.68 ERR050122: The parameter 'ComHandleId' in the container 'ComSignal' should start from <0> and should be sequential for RX Signals.

This error occurs, if Rx Signal HandleIds does not start from 0 and not sequential within the configuration.

7.1.1.69 ERR050123: The parameter ‘ComHandleId’ in the container ‘ComSignal’ should be unique for Tx Signals.

This error occurs, if Tx Signal HandleIds are not unique within the configuration.

7.1.1.70 ERR050124: The parameter ‘ComHandleId’ in the container ‘ComSignal’ should start from <0> and should be sequential for Tx Signals.

This error occurs, if Tx Signal HandleIds does not start from 0 and not sequential within the configuration.

7.1.1.71 ERR050125: The parameter ‘ComHandleId’ in the container ‘ComSignalGroup’ should be unique for Rx SignalGroups.

This error occurs, if Rx SignalGroup HandleIds are not unique within the configuration.

7.1.1.72 ERR050126: The parameter ‘ComHandleId’ in the container ‘ComSignalGroup’ should start from <0> and should be sequential for Rx SignalGroups.

This error occurs, if Rx SignalGroup HandleIds does not start from 0 and not sequential within the configuration.

7.1.1.73 ERR050127: The parameter ‘ComHandleId’ in the container ‘ComSignalGroup’ should be unique for Tx SignalGroups.

This error occurs, if Tx SignalGroup HandleIds are not unique within the configuration.

7.1.1.74 ERR050128: The parameter ‘ComHandleId’ in the container ‘ComSignalGroup’ should start from <0> and should be sequential for Tx SignalGroups.

This error occurs, if Tx SignalGroup HandleIds does not start from 0 and not sequential within the configuration.

7.1.1.75 ERR050129: Value of the parameter ‘ComHandleId’ in the container ‘ComGroupSignal’ should follow parameter ‘ComHandleId’ in the container ‘ComSignal’.

This error occurs, if Rx Group Signals does not follow Rx signals.

7.1.1.76 ERR050130: Value of the parameter ‘ComHandleId’ in the container ‘ComGroupSignal’ should be sequential for Rx Group Signals.

This error occurs, if Rx Group Signals are not sequential when Rx Signals are configured.

7.1.1.77 ERR050131: Value of the parameter ‘ComHandleId’ in the container ‘ComGroupSignal’ should follow parameter ‘ComHandleId’ in the container ‘ComSignal’.

This error occurs, if Tx Group Signals does not follow Tx signals.

7.1.1.78 ERR050132: Value of the parameter ‘ComHandleId’ in the container ‘ComGroupSignal’ should be sequential for Tx Group Signals.

This error occurs, if Tx Group Signals are not sequential when Tx Signals are configured.

7.1.1.79 ERR050133: The parameter ‘ComIPduHandleId’ in the container ‘ComIPdu’ should be unique for Rx I-PDUs.

This error occurs, if Rx I-PDU HandleIds are not unique within the configuration.

7.1.1.80 ERR050134: The parameter ‘ComIPduHandleId’ in the container ‘ComIPdu’ should start from <0> and should be sequential for Rx I-PDUs.

This error occurs, if Rx I-PDU HandleIds does not start from 0 and not sequential within the configuration.

7.1.1.81 ERR050135: The parameter ‘ComIPduGroupHandleId’ in the container ‘ComIPduGroup’ should be unique for IpduGroups.

This error occurs, if IpduGroup HandleIds are not unique within the configuration.

7.1.1.82 ERR050136: The parameter ‘ComIPduGroupHandleId’ in the container ‘ComIPduGroup’ should start from <0> and should be sequential for IpduGroups.

This error occurs, if IpduGroup HandleIds does not start from 0 and not sequential within the configuration.

7.1.1.83 ERR050139: The Container ‘ComTxModeTrue’ needs to be configured for I-PDU, when none of the Signal is configured for ‘ComFilter’.

This error occurs, if container "ComTxModeTrue" is not configured when none of the Signals of an I-PDU is not configured for "ComFilter".

7.1.1.84 ERR050140: The Container ‘ComTxModeTrue’ needs to be configured for I-PDU.

This error occurs, if "ComTxModeTrue" container is not configured for Tx I-PDU.

7.1.1.85 ERR050141: Parameter ‘ComSignallInitValue’ in the container ‘ComSignal’ should be in the range of <1.175494351E-38 to 3.402823466E+38>, since value of the ‘ComSignalType’ is configured as < FLOAT32 >.

This error occurs, if parameter "ComSignalType" is "FLOAT32" and parameter "ComSignallInitValue" is not in between the range of (1.175494351E-38 to 3.402823466E+38) or parameter "ComSignallInitValue" is not equal to zero in container "ComSignal".

7.1.1.86 ERR050142: Parameter ‘ComSignalDataInvalidValue’ in the container ‘ComSignal’ should be in the range of <1.175494351E-38 to 3.402823466E+38>, since value of the ‘ComSignalType’ is configured as < FLOAT32 >.

This error occurs, if the parameter "ComSignalType" is "FLOAT32" and parameter "ComSignalDataInvalidValue" is not in between the range of (1.175494351E-38 to 3.402823466E+38) or parameter "ComSignalDataInvalidValue" is not equal zero in container "ComSignal".

7.1.1.87 ERR050145: The Parameter ‘ComRxTimeBase’ in container ‘ComTimeBase’ should be configured, since parameter ‘ComFirstTimeout/ ComTimeout’ in the container ‘ComSignalGroup’ is configured.

This error occurs, if parameter "ComFirstTimeout"/"ComTimeout" in the container "ComSignalGroup" is configured and parameter "ComRxTimeBase" in container "ComTimeBase" is not configured.

7.1.1.88 ERR050146: Parameter ‘ComSignallInitValue’ in the container ‘ComGroupSignal’ should be in the range of <1.175494351E-38 to 3.402823466E+38>, since value of the ‘ComSignalType’ is configured as < FLOAT32 >.

This error occurs, if parameter "ComSignalType" is "FLOAT32" and parameter "ComSignallInitValue" is not in between the range of (1.175494351E-38 to 3.402823466E+38) or parameter "ComSignallInitValue" is not equal to zero in container "ComGroupSignal".

7.1.1.89 ERR050147: Parameter ‘ComSignalDataInvalidValue’ in the container ‘ComGroupSignal’ should be in the range of <1.175494351E-38 to 3.402823466E+38>, since value of the ‘ComSignalType’ is configured as < FLOAT32 >.

This error occurs, if parameter "ComSignalType" is "FLOAT32" and parameter "ComSignalDataInvalidValue" is not in between the range of (1.175494351E-38 to 3.402823466E+38) or parameter "ComSignalDataInvalidValue" is not equal to zero in container "ComGroupSignal".

7.1.1.90 ERR050148: Parameter ‘ComSignallInitValue’ in the container ‘ComGroupSignal’ should be in the range of <-1.175494351E-38 to -3.402823466E+38>, since value of the ‘ComSignalType’ is configured as < FLOAT32 >.

This error occurs, if parameter "ComSignalType" is "FLOAT32" and parameter "ComSignallInitValue" is not in between the range of (-1.175494351E-38 to -3.402823466E+38) or parameter "ComSignallInitValue" is not equal to zero in container "ComGroupSignal".

7.1.1.91 ERR050149: Value of the parameter ‘ComIPduCallout’ should be unique across the container ‘ComIPdu’.

This error occurs, if value configured for the parameter "ComIPduCallout" is not unique across the container "ComIPdu".

7.1.1.92 ERR050150: Value of the parameter ‘Parameter Name’ should be unique across the container ‘Container Name’

This error occurs, if value configured for the below parameters are not unique across the container.

Container Name	Parameter Name
ComSignal	ComInvalidNotification
	ComErrorNotification
	ComTimeoutNotification
	ComNotification
ComSignalGroup	ComInvalidNotification
	ComErrorNotification
	ComTimeoutNotification
	ComNotification

7.1.1.93 ERR050153: Parameter 'ComTransProperty' should not be configured as TRIGGERED_ON_CHANGE and TRIGGERED_ON_CHANGE_WITHOUT_REPETITION as parameter "ComBitSize" in container "ComSignal" is configured to Zero.

This error occurs, if the parameter "ComTransProperty" is configured as TRIGGERED_ON_CHANGE and TRIGGERED_ON_CHANGE_WITHOUT_REPETITION and parameter "ComBitSize" in container "ComSignal" is configured to Zero.

7.1.1.94 ERR050154: I-PDU Groups are more than ComSupportedIPduGroups. ComSupportedIPduGroups should be the same or greater than the number of configured I-PDU Groups.

This error occurs, if the parameter "ComSupportedIPduGroups" is too small. It should be the same or greater than the number of configured I-PDU Groups.

7.1.1.95 ERR050155: There is a loop in the nesting configuration. Please check and disconnect following loops by configure ComIPduGroupGroupRef correctly.

This error occurs, if some I-PDU Groups are making loops. There should not be loops in the nesting configuration.

7.1.1.96 ERR050006: ComSignalGroup {0} should not contain Dynamic ComGroupSignal since ComSignalGroupArrayAccess is TRUE.

This error occurs, if some I-PDU Groups are making loops. There should not be loops in the nesting configuration.

7.1.1.97 ERR050007: ComSignalGroup {0} should be mapped byte aligned to the I-PDU since ComSignalGroupArrayAccess is TRUE.

This error occurs, if the parameter "ComSignalGroupArrayAccess" is set to TRUE and ComSignalGroup is not mapped byte aligned.

7.1.1.98 ERR050008: ComSignalGroup {0} should not be intermitted by other signals since ComSignalGroupArrayAccess is TRUE.

This error occurs, if the parameter "ComSignalGroupArrayAccess" is set to TRUE and ComSignalGroup is intermitted by other signal(s).

7.1.1.99 ERR050156: ComSignal {0} overlaps other ComSignals in the same ComIPdu. Please edit your configuration.

This error occurs, if a ComIPdu is configured with some signals where one signal overlaps other signal on its bit size.

7.1.1.100 ERR050157: ComGroupSignal {0} overlaps other ComGroupSignals in the ComSignalGroup.

This error occurs, if a ComIPdu is configured with some group signals where one signal overlaps other signal on its bit size.

7.1.1.101 ERR050158: ComIPdu {0} is not assigned to any ComIPduGroups so ComIPduHandleId of it should be started from {1}.

This error occurs, if the parameter "ComIPduGroupRef" is not set and the parameter ComIPduHandleId is configured to value within In-group ComIPdu(s) range.

7.1.1.102 ERR050159: Signal {0} with type DYN, Bit pos must be config in the alignment byte ((bit position)%8 == 0).

This error occurs, if the parameter "ComSignalType" is configured with UINT8_DYN and parameter ComBitPosition is not in the alignment byte.

7.1.1.103 ERR050160: ComSignalEndianess {0} must be OPAQUE when ComSignalType is configured DYN.

This error occurs, if the parameter "ComSignalType" is configured with UINT8_DYN and parameter ComSignalEndianness is not configured with OPAQUE.

7.1.1.104 ERR050161: ComSignal or ComSignalGroup {0} is refered by ComIpdu {1} must be common or same variant

This error occurs, if a ComIPdu is configured with some signals and/or group signals which are defined in different variants.

7.1.1.105 ERR050162: {0} {1} which is configured variant must be.

This error occurs, if a ComIPdu is configured with some signals and/or group signals which are not all defined with variants.

7.1.1.106 ERR050022: Value of the parameter {0} in {1} should not be configured as <0>.

This error occurs, if container ComTimeBase is not provided or provided with insufficient parameters including {ComGwTimeBase, ComRxTimeBase, ComTxTimeBase}.

7.1.1.107 ERR050163: Mismatch between post-build variants collection defined in EcucPostBuildVariants and the list of variation points that was applied in Com module

This error occurs, if the predefined variants collection from ECUC module is not a subset of the list of variants to be applied in Com module in post-build configuration.

7.1.1.108 ERR050164: IPdu {0} have TxModeMode = "PERIODIC" can not refer to signal {1} have ComTransferProperty = {2}

This error occurs, if configure PERIODIC Ipdu refer to TRIGGER signal.

7.1.1.109 ERR050165: ComUserCbkHeaderFile must follow naming convention: {0}

This error occurs, if UserCbkHeader have wrong naming convention: “Not starting with a digit” and “Not using spaces or special characters”

7.1.2 Warning Messages

The following section gives the list of warning messages displayed by the Generation Tool.

7.1.2.1 WARN050001: Mismatch between false value in ComMainFunctionBasedTransmit and true value in /AUTRON/Com/ComConfig/<ComIPdu>/ComTxIPdu/ComTxModeTrue(OrFalse)/ComTxMode / ComDirectTransmit

This error occurs, if value of ComMainFunctionBasedTransmit is mismatched with any value in containers ComIPdu/ComTxIPdu/ComTxModeTrue/ComTxMode/ComDirectTransmit.

7.1.3 Information Messages

The following section gives the list of Information messages displayed by the Generation Tool.

8. Appendix

8.1 Time out

- Regarding timeout, the timeout value used in the existing Vector CAN is not in the COM specification. There is a function to replace with init value and replace function at timeout as a similar function.
- Although it is defined as GenSigTimeoutValue in the DBC definition, in the Xenon platform, there is a value that sets the relevant item and converts it to the corresponding value at timeout. You should.

8.2 Tx Mainfunction

- If there is a message sent with a period of 10ms, the period of the main function should be set to at least 5ms.
- The smaller the cycle value of MainFunction, the smaller the time error in the operation of Cyclic message, Repetition, MDT, etc. processed through MainFunction. But CPU usage goes up.

8.3 GW Mainfunction

- Usually, two of Com_MainfunctionTx and Com_MainFunctionRx are registered and used among the main functions of COM. In case of using Signal Gateway, Com_MainFunctionRouteSignals must be added.
- During signal gateway, it affects the gateway speed according to the main function (Com_MainFunctionRouteSignals) cycle of COM. Also, arranging the main function on the TASK in the order of Rx, Gw, and Tx can improve the gateway performance.

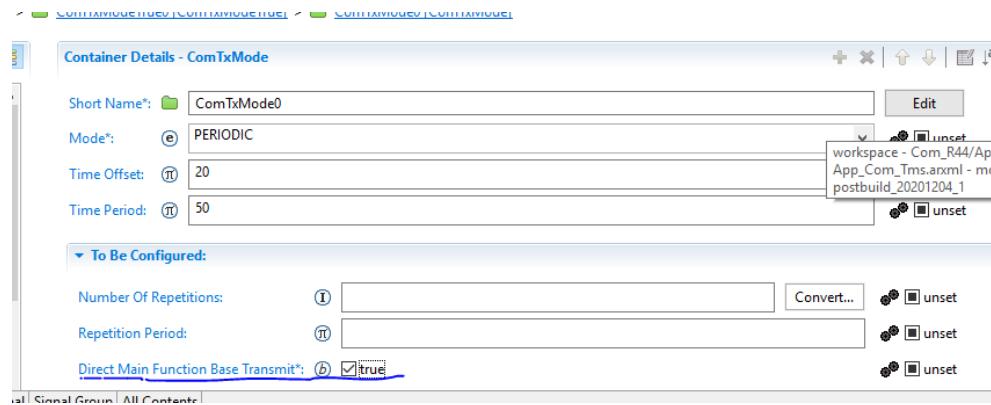
8.4 Signal group Configuration

- If you want to use a signal group, modify the group-related information in the DB file and proceed with import and harmonize again.
- Since there are many related settings to modify directly, it is recommended to use automatic settings through DB change.

8.5 Transmit Mode Selection

- Two transmission modes may be configured in one Com I-PDU (ComTxModeTrue / ComTxModeFalse). Transmission Mode Selection is a function of dynamically selecting a transmission method for a specific Com I-PDU as one of the above two transmission modes. For example, a message that was transmitted as a periodic message may be transmitted as an event message.
- As a method of selecting a transmission method, there are Signal Based TMS and Explicit TMS.
- In Signal Based TMS, a transmission method is selected according to the result of ComFilter set in ComTxSignal belonging to the corresponding Com I-PDU. When Rte_Write is called for the corresponding ComTxSignal, ComTxModeTrue is selected if the filter result of the requested data is True, and ComTxModeFalse is selected if the result is False.
- Explicit TMS is a method in which the transmission method is changed by calling Com_SwitchIpduTxMode function of Com, and the corresponding API must be called as Action of BswM.

8.6 Direct Transmission Without Mainfunction



When parameter TxMode/ComDirectMainFunctionBasedTransmit is configured true. This IPDU will be transmit directly without waiting for next Tx Mainfunction if any signal with transfer property configured different to PENDING belong to this IPDU is send by calling Com_SendSignal.

8.7 Multiplexed Message Guide

- Signals corresponding to the multiplexor of Multiplexed Message must not be written through RTE.
- In the screen below, MUL_CODE Signal is a value corresponding to the Multiplexor, and the Init Value of Com should be maintained. (It should not be updated)

- The corresponding signal is included in each Dynamic PDU, and if the Init Value set in the Com module does not match the Selector Field, the initial value (1 or 3 in the screen below) should be set after checking.

Name	Message	Multiplexing/Group
☒ TQ_STND	EMS12	MUL_CODE = 0x3
☒ CONF_TCU	EMS12	MUL_CODE = 0x1
☒ MUL_CODE	EMS12	Multiplexor
☒ TEMP_ENG	EMS12	-
☒ BRAKE_ACT	EMS12	-
☒ ENG_CHR	EMS12	-
☒ TPS	EMS12	-
☒ PV_AV_CAN	EMS12	-
☒ ENG_VOL	EMS12	-

8.8 Signal Gateway Configuration Guide

- Refer to PduR Module User Manual

8.9 Signal Gateway Callout Function Guide

8.9.1 Function overview

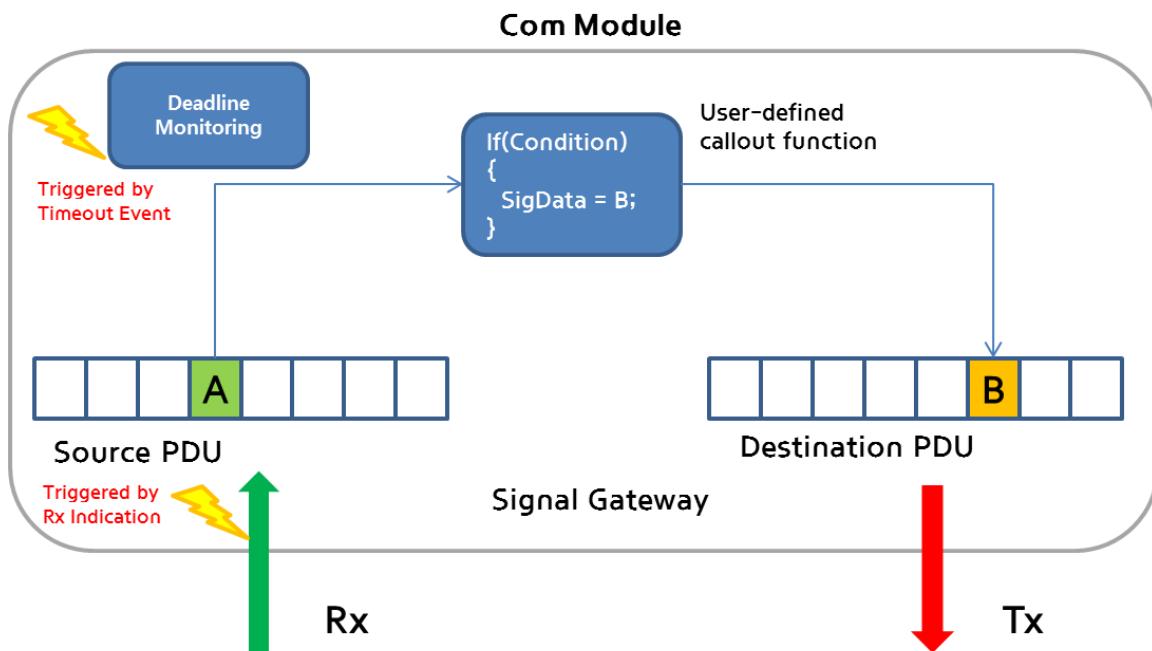
AUTOSAR's basic rule of Signal Gateway (Indirect Gateway) transmits the value to Tx message in signal unit without modification of received signal data, and specifies that there should be no modification of data. In actual project requirements, most Signal Gateway (Indirect Gateway) functions change values, and in many cases, different values must be delivered according to conditions. For this reason, Signal Gateway (Indirect Gateway) could not process in the AUTOAR Com module, and had to structure the application to process it through RTE.

The structure in which signals are transmitted to the application through RTE goes through several layers, which leads to an increase in CPU load, and supports the signal gateway callout function as a way to improve this.

As shown in the figure below, when the signal gateway is established, the callout function registered by the user is called, and the signal value can be checked and changed to the desired signal value. Endian conversion is handled by COM, so you can read the value by casting to the source signal type, and assign the value by casting to the destination signal type according to the condition.

According to the specification, the bit size and signal type (e.g. UINT8, UINT16) of the source signal and the destination signal should be the same. Restrictions are lifted.

Signal Gateway Callout function is an additional function not defined in AUTOSAR for application implementation and performance. Signal data related pointers must be handled directly. The platform is not concerned with pointer arithmetic.



8.9.2 Configuration

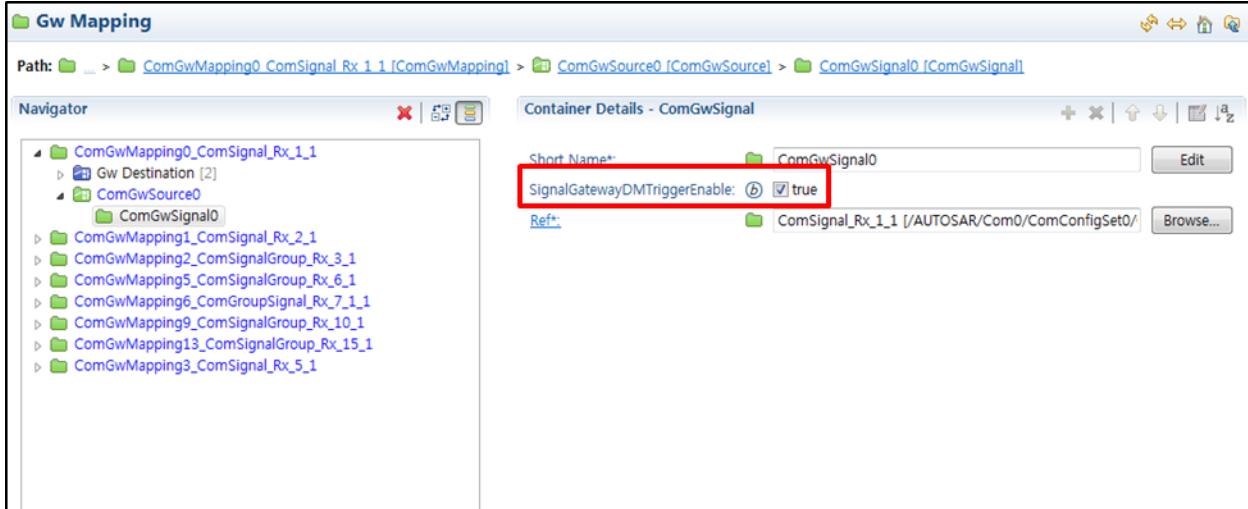
8.9.2.1 Source Signal Setting

Since there are many Indirect Gateway requirements to be processed when timeout occurs, a part of AUTOSAR Com Module 4.2.X specification is applied in advance. If a timeout occurs during Deadline Monitoring in Rx PDU, a Signal Gateway is performed with the value of the current Rx PDU. However, if Replace is set at timeout, the signal value changed to Init Value is transmitted. Timeout Value is not supported. If necessary, it must be implemented through Signal Gateway Callout or implemented in the application through RTE.

In order to be able to select whether to use the signal gateway function at timeout, a configuration item has been added.

8.9.2.1.1 ComSignalGatewayDMTriggerEnable

TRUE	When timeout occurs, gateway processing
FALSE	When timeout occurs, gateway processing is not performed



Note: In order to activate the Rx PDU Timeout function, the following two conditions must be satisfied.

Condition 1: Prerequisite

- Among the signals included in the Rx PDU, a signal with a ComTimeout value of 0 or more exists

Condition 2: Activated when only one of the following three conditions is satisfied

- When ComTimeoutNotification is set in the signal corresponding to condition 1
- When ComRxDataTimeoutAction is set to REPLACE in the signal corresponding to condition 1
- When the signal corresponding to condition 1 has Signal Gateway setting (Source) and ComSignalGatewayDMTriggerEnable is set to TRUE

8.9.2.2 Destination Signal Setting

AUTOSAR Com Module supports 1:N Signal Gateway, so multiple destination signals can be set for one source signal. Signal gateway callout function can be set for each destination signal.

8.9.2.2.1 ComGwSignalCallout

Defines the callout function name to be called when the set signal gateway is processed.

- Callout function is declared in Com_Cbk.h file created by Com generator.
- Application developer must implement the body of the callout function.

Signal Gateway Callout Function Definition:

```
boolean Com_SignalGWTest00 (void* SignalDataPtr, uint8 status);
```

SignalDataPtr	Signal data pointer address Values can be assigned through pointers. The data type of the signal should be checked by the application developer and processed by casting.
status	Information about the signal gateway state when a callout is called 0x00 : Signal gateway executed when a general Rx message is received 0x01 : Signal gateway executed when timeout occurs
Return	Depending on the return value, it is decided whether to process the Signal Gateway. Return to TRUE: Signal gateway processing. Return to FALSE: Signal gateway processing is not performed.

Signal Gateway Callout function code example:

- You can receive the value by casting SignalDataPtr by checking the type of the source signal.
- By checking the type of the destination signal in SignalDataPtr, the value can be assigned by casting.
- A value exceeding the bit size of the destination signal is removed by masking. It needs to be implemented so that the correct value can be assigned by checking the type and bit size of the destination signal.
- Through the status value, it is possible to determine whether it is a Signal Gateway called at timeout.
- If you do not need to do the gateway, you can handle it by adjusting the return value.

The example code below is for reference only.

```
boolean Com_SignalGWTest00 (void* SignalDataPtr, uint8 status)
{
    uint8 ABC = 0;
    uint16 ABC_u16 = 0;
    boolean rtnValue = FALSE;
    if(status == 1)
    {
        *(uint16*)(SignalDataPtr) = 0xFFFF;
    }
    else
    {
        ABC = *(uint8*)(SignalDataPtr);
        if(ABC < XXX)
        {
            ABC_u16 = 0x1234;
        }
        else if(ABC < XXX)
        {
            ABC_u16 = 0x3456;
        }
        else if(ABC < XXX)
        {
            ABC_u16 = 0x5678;
        }
    }
}
```

```
{  
    ABC_u16 = 0x789A;  
}  
else  
{  
    ABC_u16 = 0xBCDE;  
}  
  
*(uint16*)(SignalDataPtr) = ABC_u16;  
}  
  
If(GW 전송조건)  
{  
    rtnValue = TRUE;  
}  
return rtnValue;  
}
```

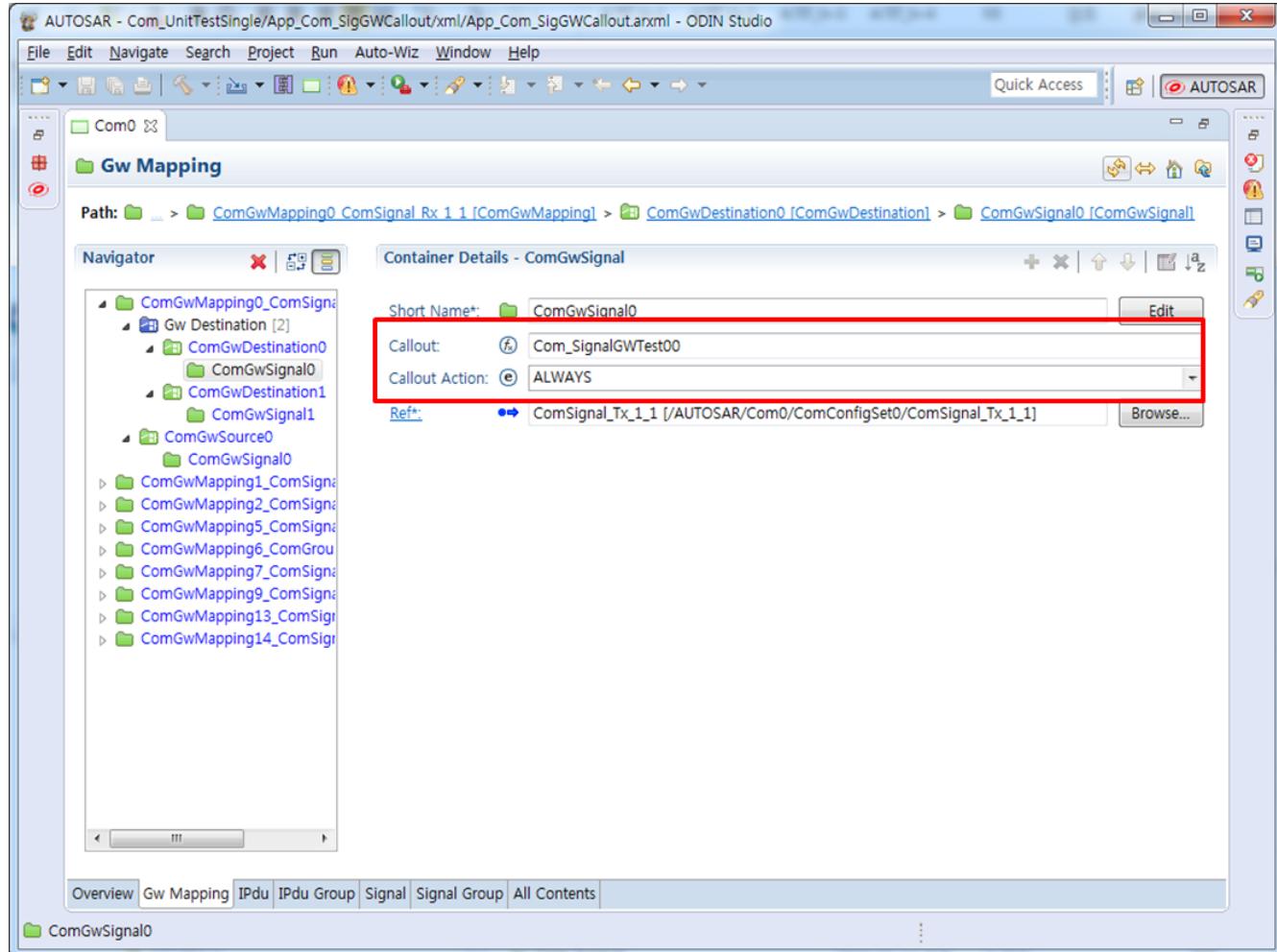
Notes:

In the case of SignalGroup gateway, it is not possible to change the value of the group signal, only processing the return value of the callout function.

8.9.2.2ComGwSignalCalloutAction

Defines the condition under which the callout function is called.

ALWAYS	When Signal Gateway is processed, all callout functions are called.
TIMEOUT_ONLY	Callout function is called only when Signal Gateway is processed due to timeout.



8.9.3 Auto setup related

After creating the gateway system configuration file (e.g. Gateway.arxml) through RDB Importer and manual setting, when Signal Gateway is automatically set using Odin Tool, basic Signal Gateway settings and Signal Gateway callout related settings are also automatically set. The Signal Gateway Callout Function name is set based on the Source and Destination signal names.

The declaration part of Signal Gateway Callout Funciton is automatically created in Com_Cbk.h, and the part corresponding to the function body should be created by the application developer. For the convenience of generating the function body, the Com Module Generator generates the sample code in the form of Com_CalloutSample.c.txt. Generally, it is created in the path of "ProjectRoot\Generated\Bsw_Output\src". It is a sample code for link errors when compiling and for the convenience of application developers, so please refer only to it.

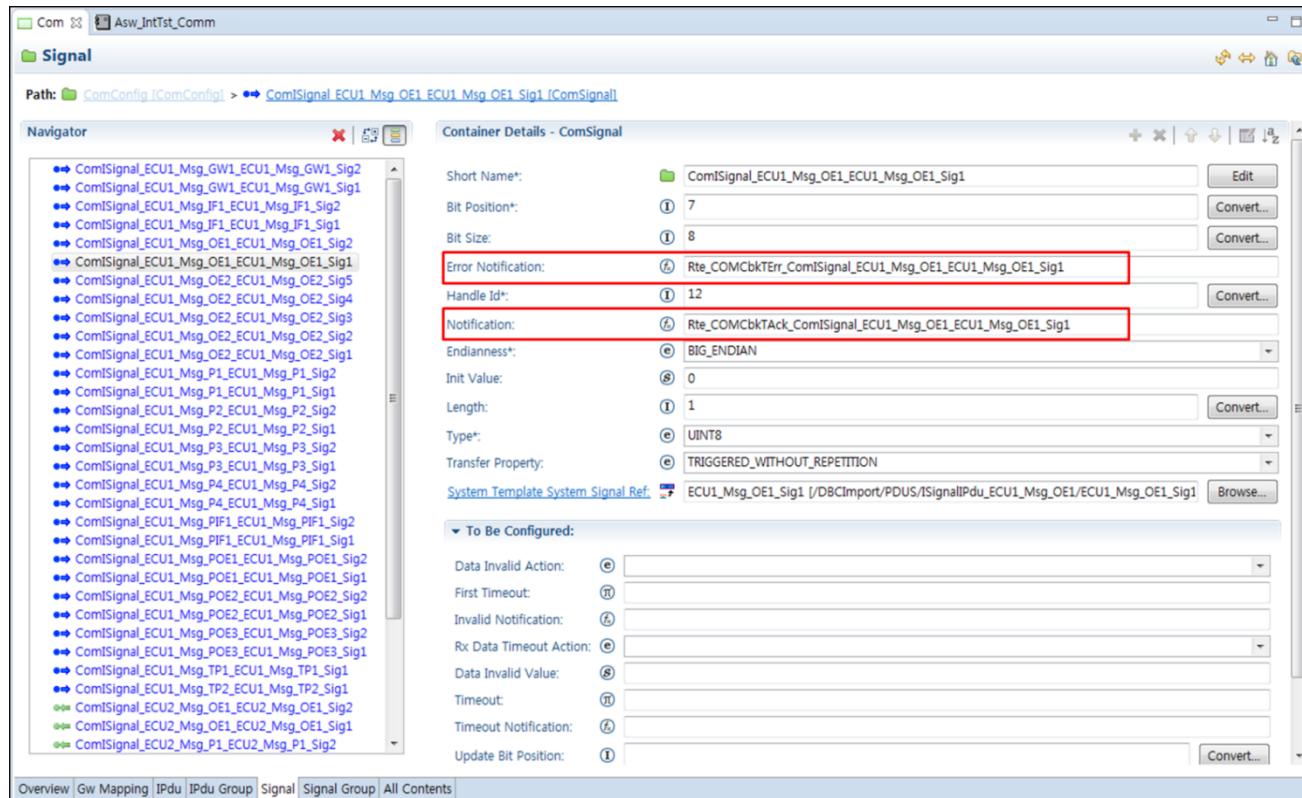
8.10 Com Signal Callback Function Naming rule and setting

The following is the naming convention for the Callback function defined in AUTOSAR.

If the following naming rules are not followed, RTE does not process it.

Rte_COMCbkTAck_<sn> callbacks	0x90	X				
Rte_COMCbkTErr_<sn> callbacks	0x91	X				
Rte_COMCbkInv_<sn> callbacks	0x92	X				
Rte_COMCbkRxTOut_<sn> callbacks	0x93	X				
Rte_COMCbkTxTOut_<sn> callbacks	0x94	X				
Rte_COMCbk_<sg> callbacks	0x95	X				
Rte_COMCbkTAck_<sg> callbacks	0x96	X				
Rte_COMCbkTErr_<sg> callbacks	0x97	X				
Rte_COMCbkInv_<sg> callbacks	0x98	X				
Rte_COMCbkRxTOut_<sg> callbacks	0x99	X				
Rte_COMCbkTxTOut_<sg> callbacks	0x9A	X				

This callback function can be set in each signal.



If necessary, they can be added in bulk as follows.

Reference:

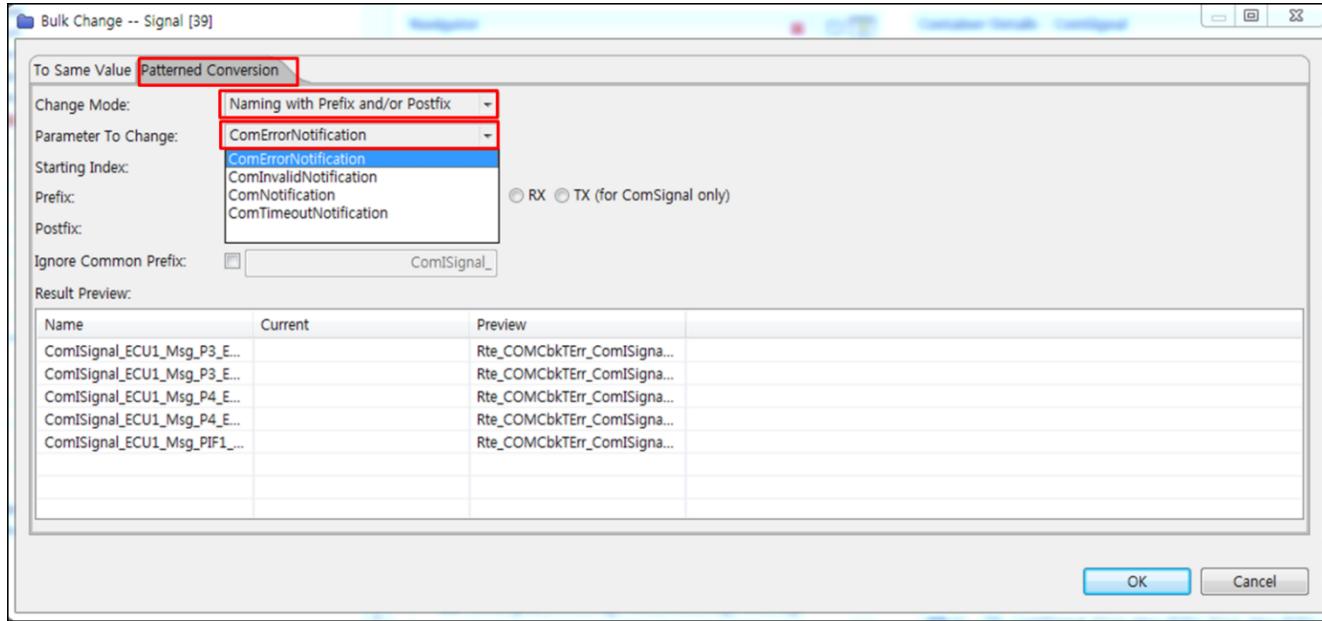
The setting guide below is for Odin 2017a.

After Odin 2017b (SP1), it is set as a separate dedicated UI. (9.11, 9.12)

Index	Short Name	Bit Position	Bit Size	Data Type	Change Mode
0	ComISignal_ECU1_Msg_GW1_ECU1_Msg_GW1_Sig2	15	1	e	✓
1	ComISignal_ECU1_Msg_GW1_ECU1_Msg_GW1_Sig1	7	1	e	✓
2	ComISignal_ECU1_Msg_IF1_ECU1_Msg_IF1_Sig2	15	1	e	✓
3	ComISignal_ECU1_Msg_IF1_ECU1_Msg_IF1_Sig1	7	1	e	✓
4	ComISignal_ECU1_Msg_OE1_ECU1_Msg_OE1_Sig2	12	1	e	✓
5	ComISignal_ECU1_Msg_OE1_ECU1_Msg_OE1_Sig1	7	1	e	✓
6	ComISignal_ECU1_Msg_OE2_ECU1_Msg_OE2_Sig5	35	1	e	Rte_CO
7	ComISignal_ECU1_Msg_OE2_ECU1_Msg_OE2_Sig4	26	1	e	✓
8	ComISignal_ECU1_Msg_OE2_ECU1_Msg_OE2_Sig3	17	1	e	✓
9	ComISignal_ECU1_Msg_OE2_ECU1_Msg_OE2_Sig2	11	1	e	✓
10	ComISignal_ECU1_Msg_OE2_ECU1_Msg_OE2_Sig1	7	1	e	✓
11	ComISignal_ECU1_Msg_P1_ECU1_Msg_P1_Sig2	15	1	e	✓
12	ComISignal_ECU1_Msg_P1_ECU1_Msg_P1_Sig1	7	1	e	✓
13	ComISignal_ECU1_Msg_P2_ECU1_Msg_P2_Sig2	15	1	e	✓
14	ComISignal_ECU1_Msg_P2_ECU1_Msg_P2_Sig1	7	1	e	✓
15	ComISignal_ECU1_Msg_P3_ECU1_Msg_P3_Sig2	15	1	e	✓
16	ComISignal_ECU1_Msg_P3_ECU1_Msg_P3_Sig1	7	1	e	✓
17	ComISignal_ECU1_Msg_P4_ECU1_Msg_P4_Sig2	15	1	e	✓
18	ComISignal_ECU1_Msg_P4_ECU1_Msg_P4_Sig1	7	1	e	✓
19	ComISignal_ECU1_Msg_PIF1_ECU1_Msg_PIF1_Sig2	15	1	e	✓
20	ComISignal_ECU1_Msg_PIF1_ECU1_Msg_PIF1_Sig1	7	1	e	✓
21	ComISignal_ECU1_Msg_POE1_ECU1_Msg_POE1_Sig2	15	1	e	✓
22	ComISignal_ECU1_Msg_POE1_ECU1_Msg_POE1_Sig1	7	1	e	✓
23	ComISignal_ECU1_Msg_POE2_ECU1_Msg_POE2_Sig2	15	1	e	✓
24	ComISignal_ECU1_Msg_POE2_ECU1_Msg_POE2_Sig1	7	1	e	✓
25	ComISignal_ECU1_Msg_POE3_ECU1_Msg_POE3_Sig2	15	1	e	✓
26	ComISignal_ECU1_Msg_POE3_ECU1_Msg_POE3_Sig1	7	1	e	✓
27	ComISignal_ECU1_Msg_TP1_ECU1_Msg_TP1_Sig1	0	64	e	Rte_CO

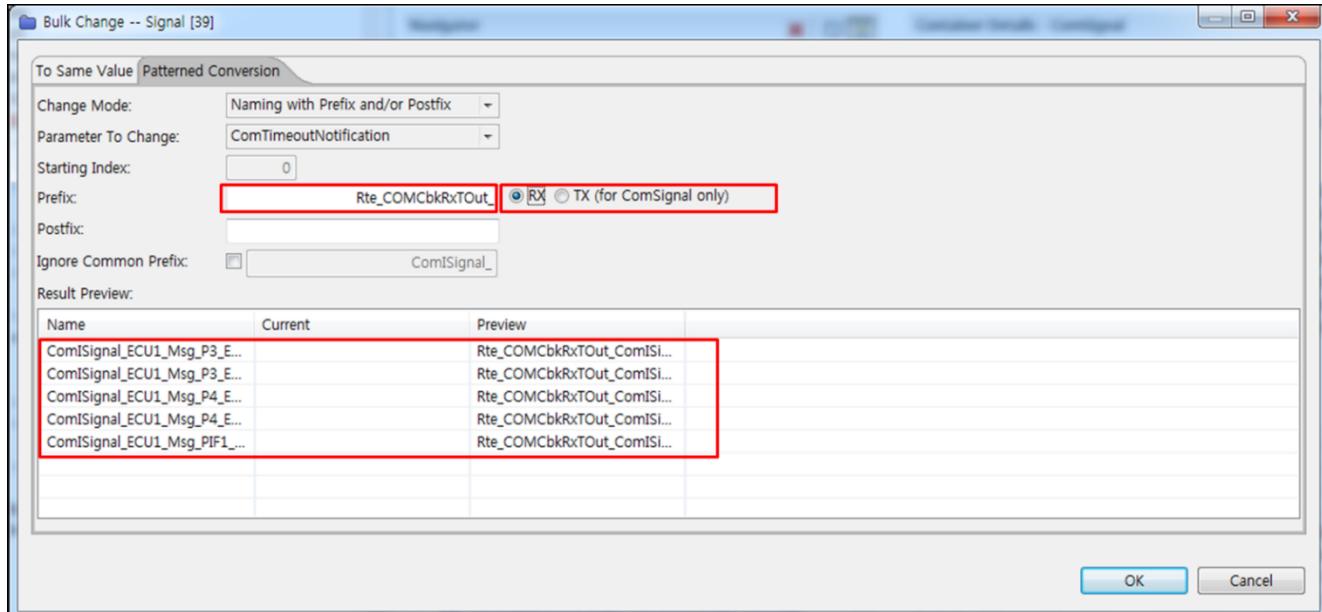
Go to Patterned Conversion and change the Change Mode as follows.

You can select the desired callback function in Parameter To Change.



According to the above option, you can see the characters that are attached to the prefix at once, and select the Tx, Rx Signal properties. (Because the naming convention is different depending on Tx and Rx)

In the preview screen, you can check the name of the callback function to be actually created.



8.11 Signal Port Connection and Reception Signal Notification Settings

Reference:

The setting guide below is for Odin 2017a.

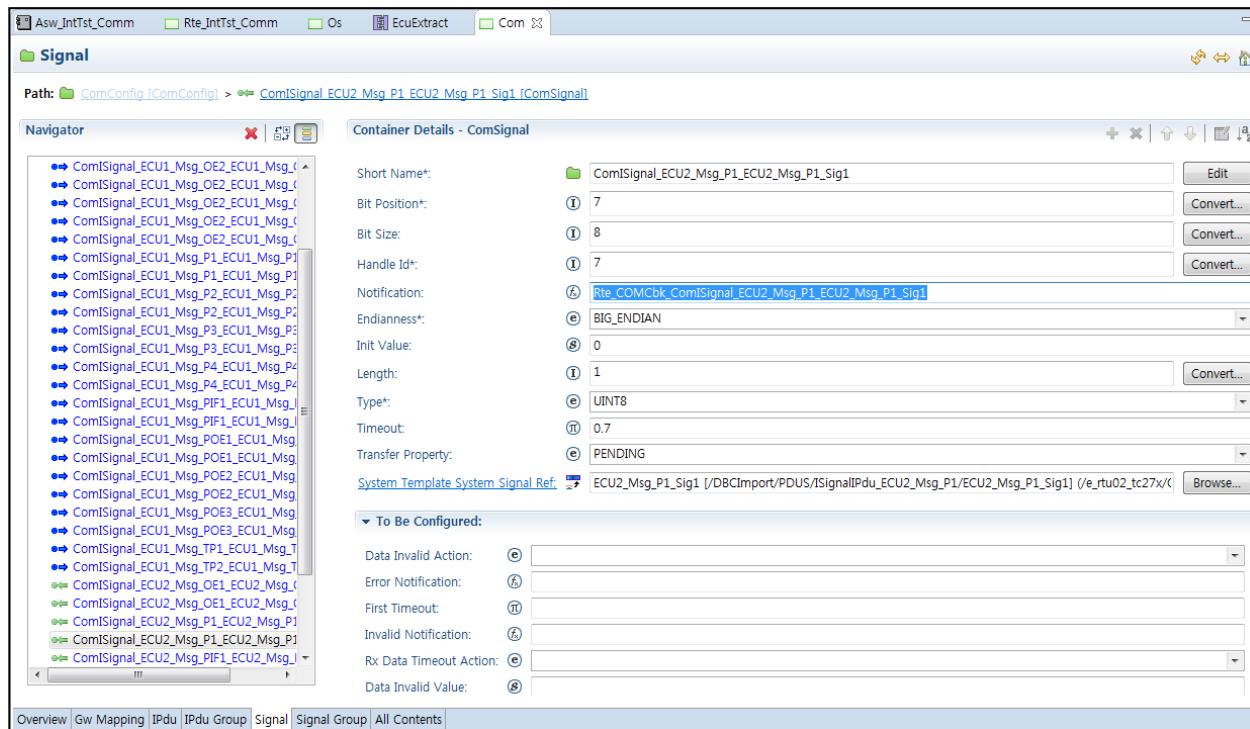
After Odin 2017b (SP1), it is set as a separate dedicated UI. (8.13,8.15)

8.11.1 Reception Notification Setting

8.11.1.1 Configuration

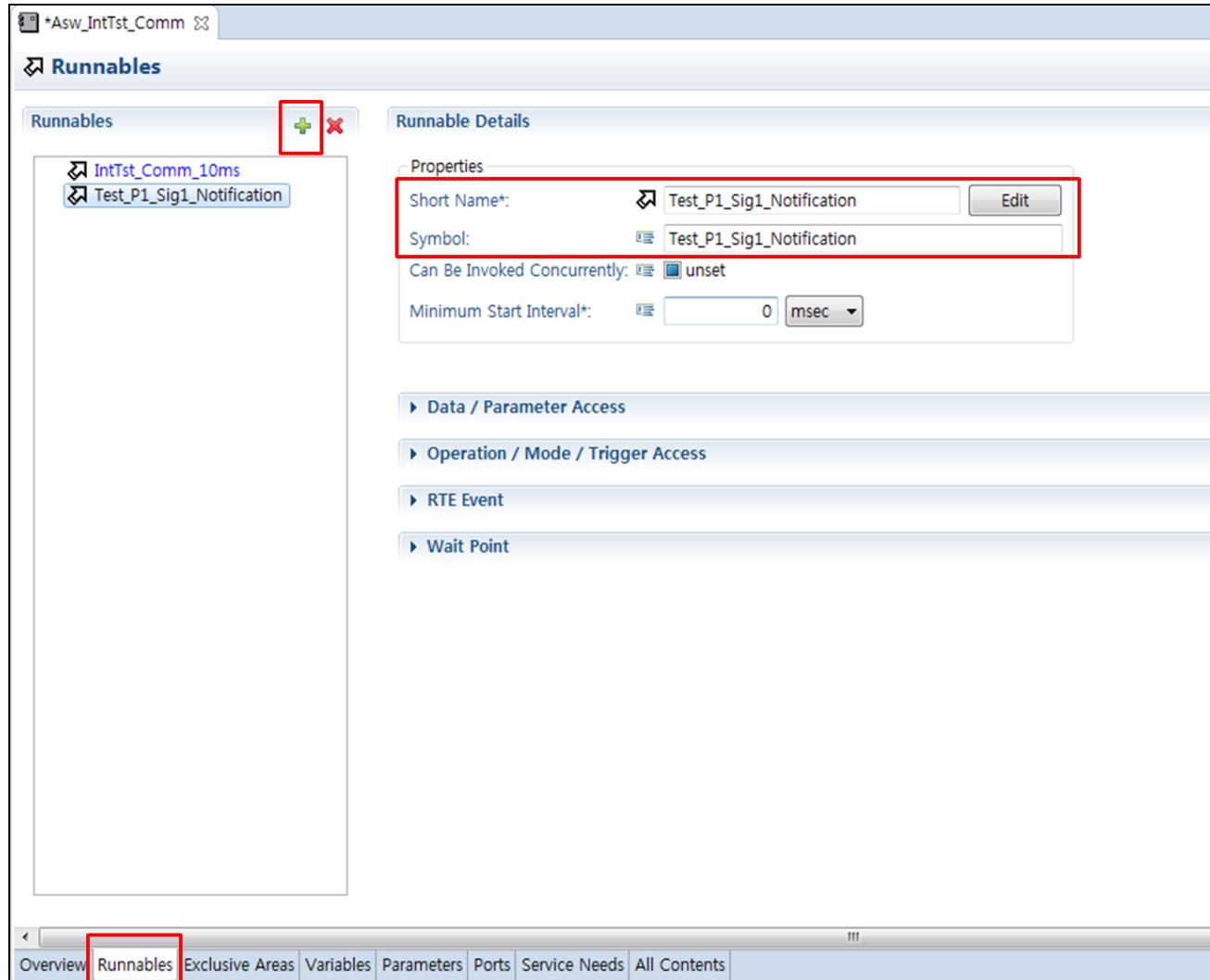
8.11.1.1.1 Com Configuration

1. Select the relevant signal in the Com module setting screen and register the Notification function.

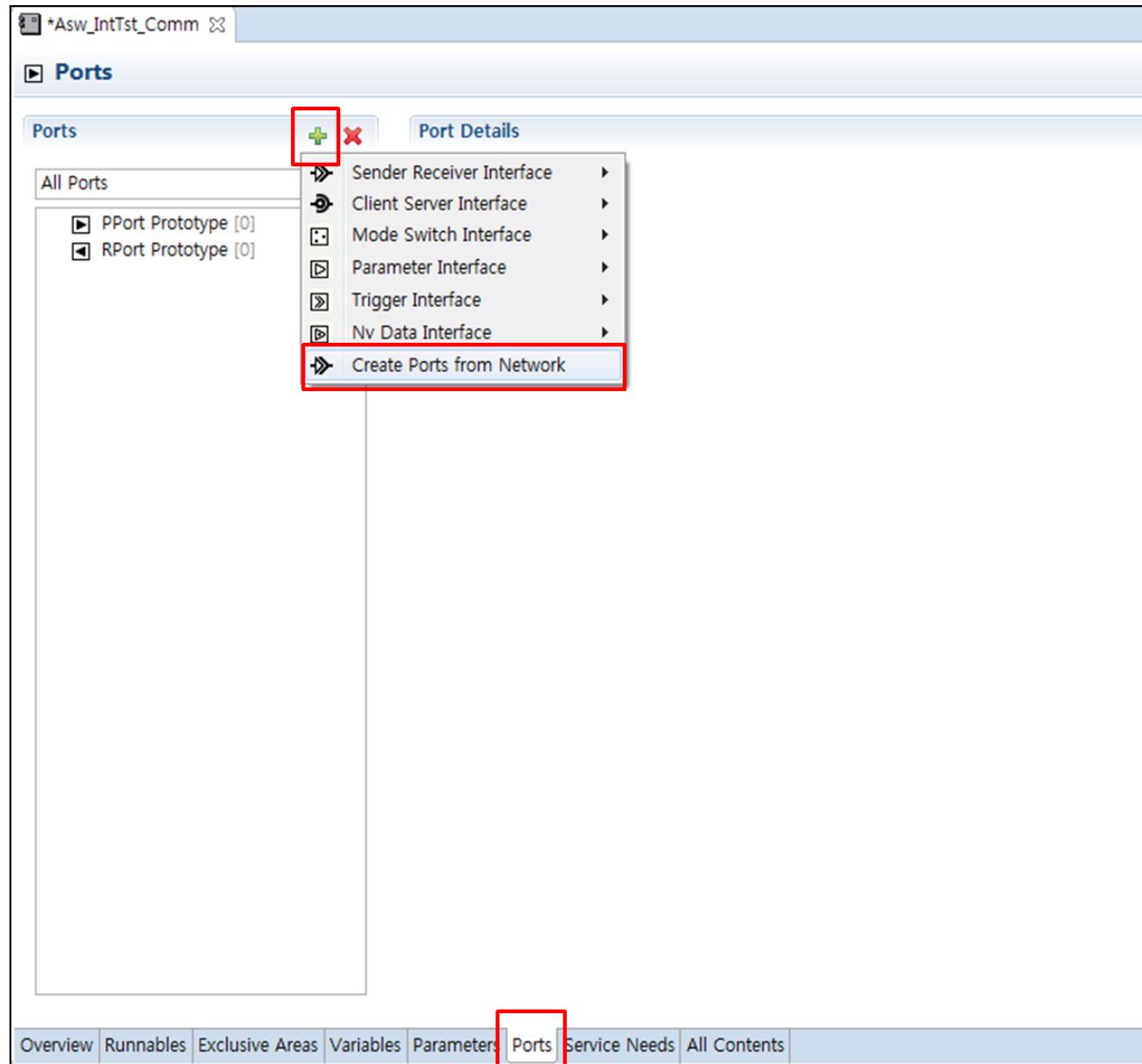


8.11.1.1.2 SW-C Configuration

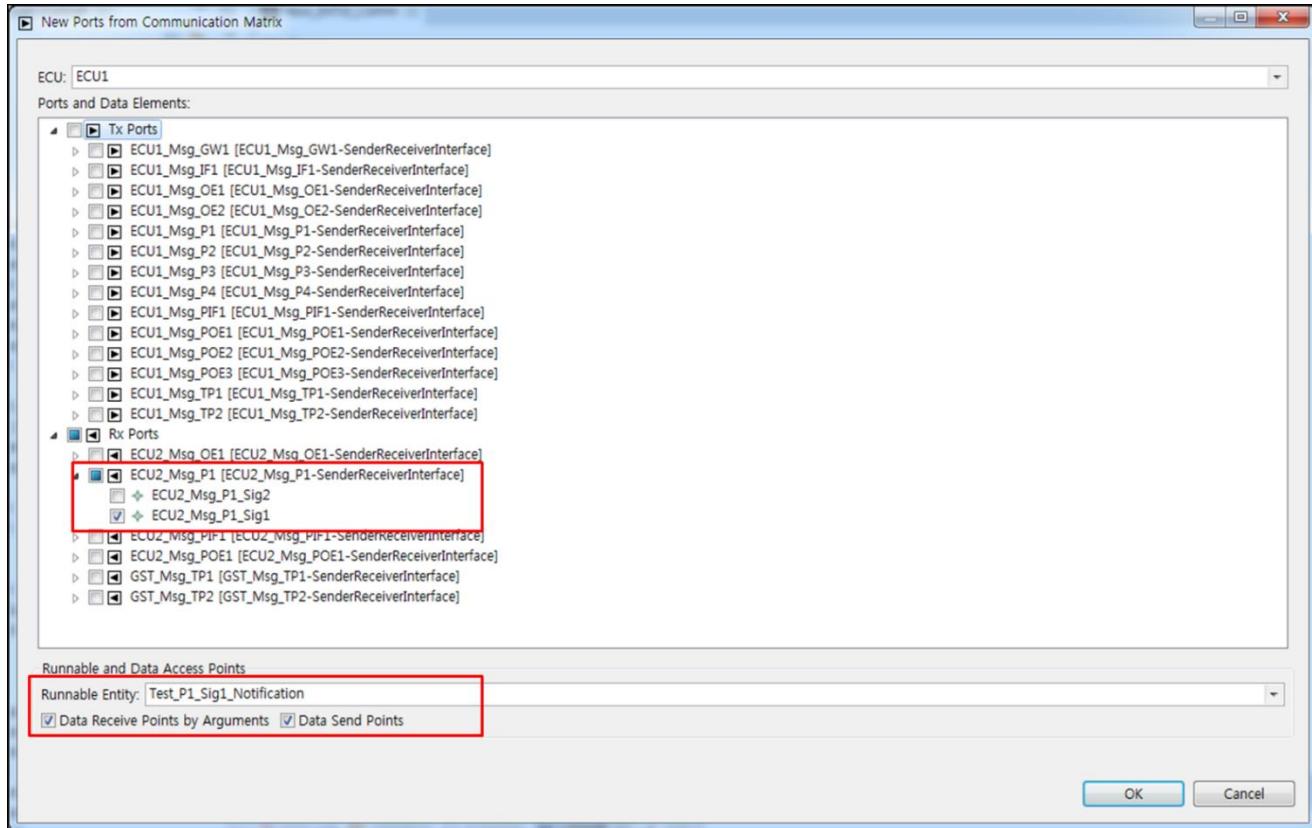
1. Select SW-C to process signal and create Runnable.



2. Go to the Port tab and click the Add button.



3. Select the signal to register the port, and select the runnable to use the port in the select box below.



4. Check if it has been added normally in the All Contents tab.

The screenshot displays the 'All Contents' tab of the AUTOSAR Com User Manual. The Navigator pane on the left shows a hierarchical tree structure of the project. Two specific sections are highlighted with red boxes:

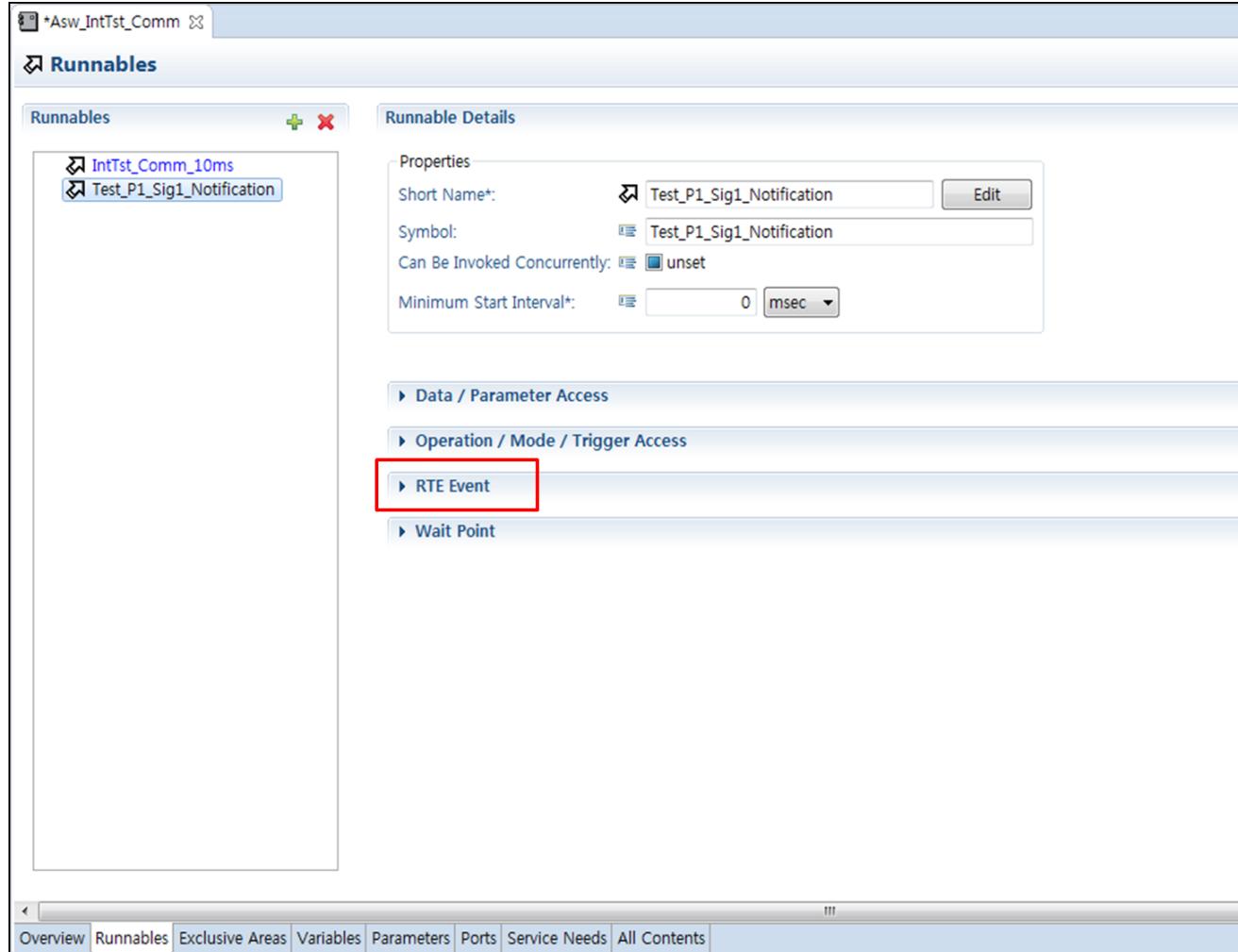
- Test_P1_Sig1_Notification**: Located under the **SwcDB_IntTst_Comm** node. This node is checked in the 'To Be Configured' section of the dialog box.
- Required Com Specs**: Located under the **ECU2_Msg_P1** node. This node is also checked in the 'To Be Configured' section of the dialog box.

The dialog box on the right is titled 'Container Details - Autosar Variable > Variable In Atomic SWC Type Instance Ref'. It contains several configuration fields:

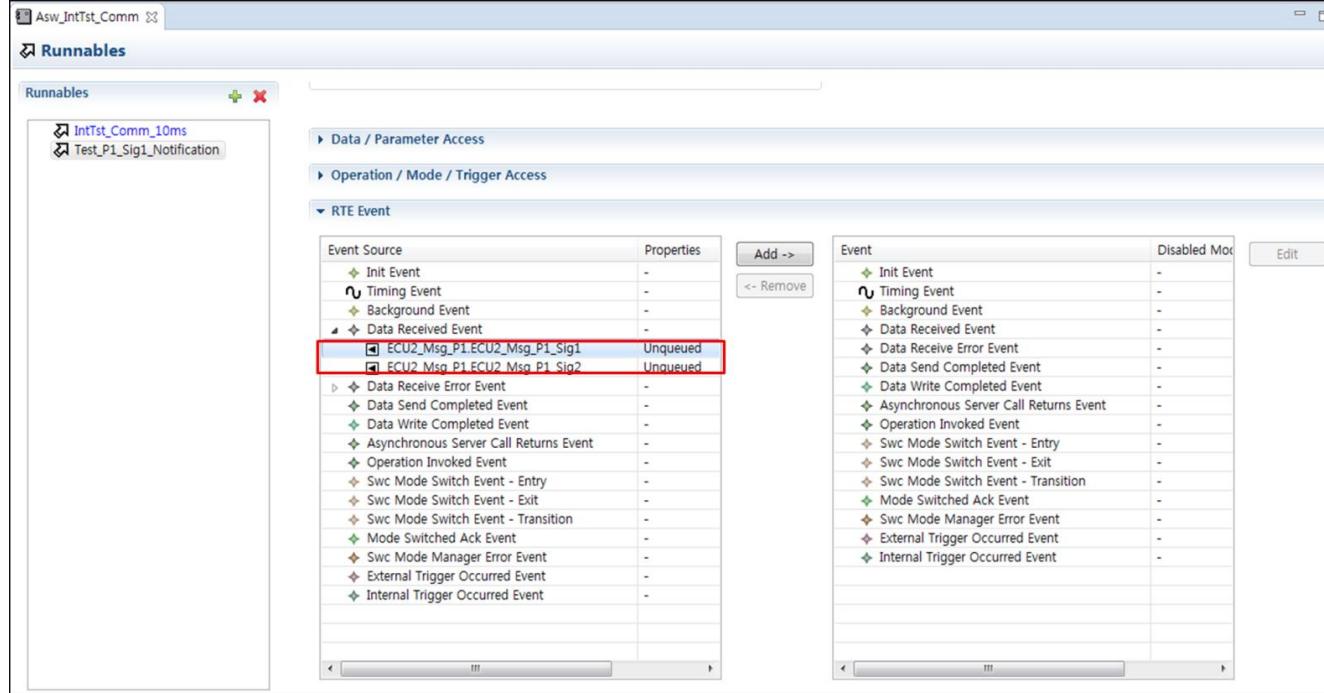
- Base:** Asw_IntTst_Comm [/ARPackage_IntTst_Comm/Asw_IntTst_Comm] (/e_rtu02_tc27x/Static_Code/b_)
- Port Prototype:** ECU2_Msg_P1 [/ARPackage_IntTst_Comm/Asw_IntTst_Comm/ECU2_Msg_P1] (/e_rtu02_tc27x/Stat) [Browse...]
- Target Data Prototype:** ECU2_Msg_P1_Sig1 [/DBCImport/INTERFACES/ECU2_Msg_P1/ECU2_Msg_P1_Sig1] (/e_rtu02_tc27x/Stat) [Browse...]
- To Be Configured:**
 - Root Variable Data Prototype:** [] [Browse...]
 - Context Data Prototypes:** [] [Browse...]
- General:** (This section is collapsed)

At the bottom of the interface, there is a navigation bar with links: Overview, Runnables, Exclusive Areas, Variables, Parameters, Ports, Service Needs, All Contents.

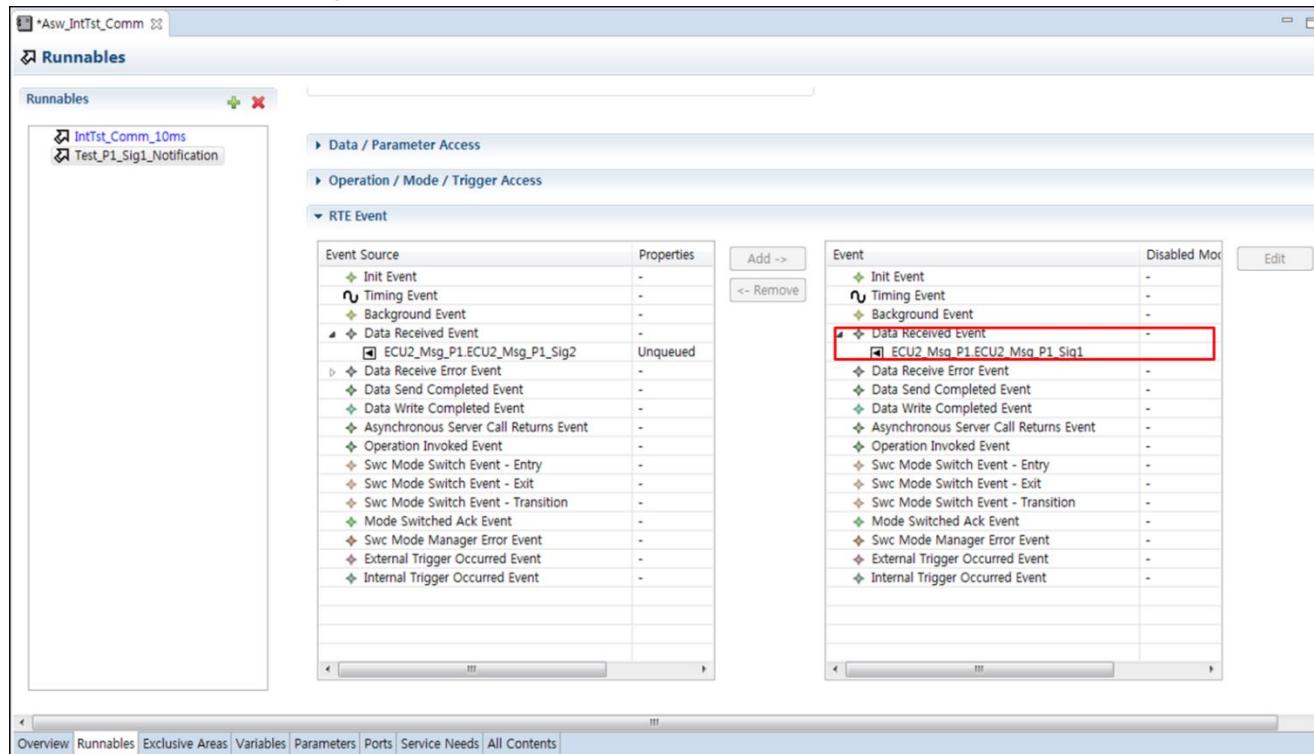
5. Go to the Runnable tab, and press RTE Event.



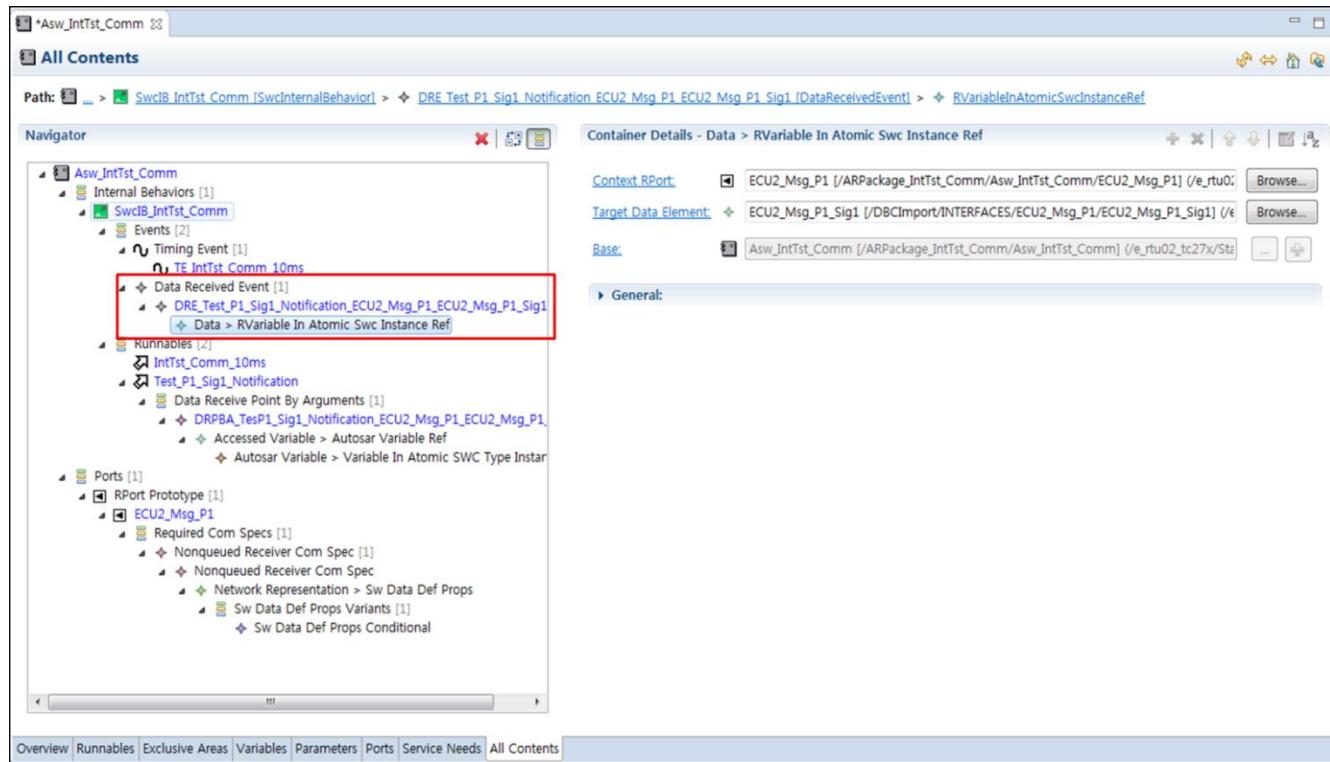
6. Select the corresponding signal of Data Received Event and click the Add button



7. Check if the event is registered.

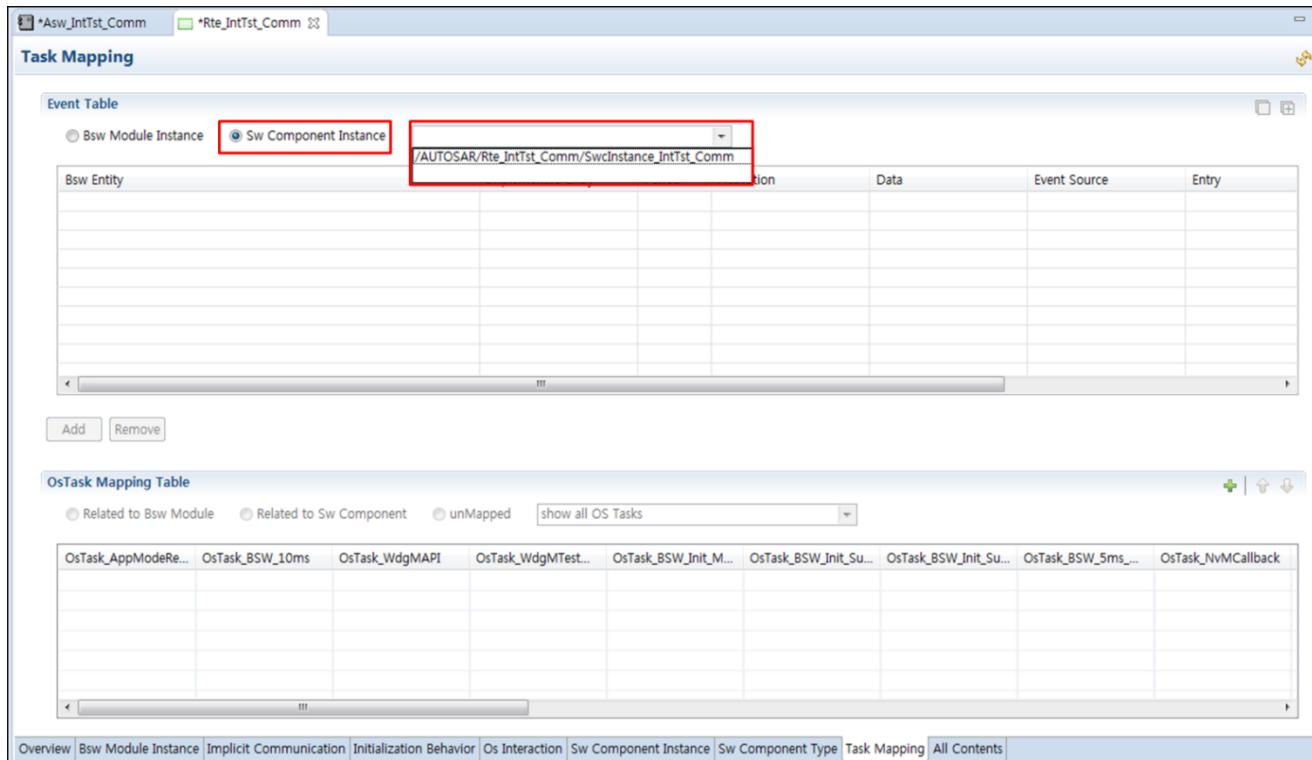


8. Go to the All Contents tab and check whether the event is set normally.



8.11.1.1.3 Rte/OS Configuration

1. Go to Rte settings and go to Task Mapping tab. Select Sw Component Instance and select SW-C set above in the select box.



2. Select the Runnable to be mapped from the Event Table, and if there is a task created in advance in the OsTask Mapping Table, select it through the select box. If not, click the Add button on the right to add the

task.

The screenshot shows the Task Mapping interface in a software tool. At the top, there are tabs for "Asw_IntTst_Comm" and "Rte_IntTst_Comm". The main area is titled "Task Mapping" and contains two tables: "Event Table" and "OsTask Mapping Table".

Event Table: This table lists events mapped to tasks. The columns are: Runnable Entity, Period, Operation, Activation, Data, Event Source, and Trigger. A red box highlights the row for "Test_P1_Sig1_Notification" under "Runnable Entity".

Runnable Entity	Period	Operation	Activation	Data	Event Source	Trigger
Asw_IntTst_Comm						
Asw_IntTst_Comm						
IntTst_Comin_10ms						
TE_IntTst_Comm_10ms	0.01					
Test_P1_Sig1_Notification					ECU2_Msg_P1_Sig1	
DRE_Test_P1_Sig1_Notification_ECU2_Msg_P1_ECU2_Ms						

OsTask Mapping Table: This table maps RTE events to OS tasks. The columns are: Position, RteEvent, Software Compone..., Runnable Entity, Operation, Period, Offset, Activation, and Data. A red box highlights the row for "TE_IntTst_Comm_10ms" under "RteEvent". Another red box highlights the "Related to Sw Component" filter dropdown set to "OsTask_ASW_10ms".

Position	RteEvent	Software Compone...	Runnable Entity	Operation	Period	Offset	Activation	Data
103	TE_IntTst_Comm_10ms	Asw_IntTst_Comm	IntTst_Comin_10ms	-	0.01	-	-	

At the bottom of the interface, there is a navigation bar with links: Overview, Bsw Module Instance, Implicit Communication, Initialization Behavior, Os Interaction, Sw Component Instance, Sw Component Type, Task Mapping, and All Contents.

3. Add necessary tasks in OS setting. (Task detailed setting items are set by the user considering the project.)

The screenshot shows a software interface for managing tasks. At the top, there are tabs for "Asw_IntTst_Comm", "Rte_IntTst_Comm", and "Os". Below the tabs, the title bar says "Task". The path is listed as "Path: Os [Os] > OsTask0 [OsTask]". On the left, a "Navigator" pane displays a tree view of tasks, including "OsTask_AppModeRequest", "OsTask_BSW_10ms", "OsTask_WdgMAPI", "OsTask_WdgMTestMode", "OsTask_BSW_Init_Main", "OsTask_BSW_Init_Sub1", "OsTask_BSW_Init_Sub2", "OsTask_NvMCallback", "OsTask_ASW_10ms", "OsTask_BSW_10ms_Sub1", "OsTask_BSW_10ms_Sub2", "OsTask_BSW_5ms_Comp", "OsTask_BSW_RteStart_Sub1", "OsTask_BSW_RteStop_Sub1", "OsTask_BSW_RteStart_Sub2", "OsTask_BSW_RteStop_Sub2", "OsTask_BSW_Mem_Process", "OsTask_RteComTask", "OsTask_ASW_5ms", "OsTask_ASW_100ms", "OsTask_ASW_LinScheduleRequest", "OsTask_BSW_FG1_IOManager", "OsTask_BSW_FG2_IOManager", and "OsTask_CommCbkTask". The main area is titled "Container Details - OsTask" and contains a table with columns: Index, Short Name, Activation, Priority, Schedule, Stack Size, and Accessing Application. The table lists 24 tasks, with the last task, "OsTask_CommCbkTask" (Index 24), highlighted with a red border. The "Accessing Application" column shows various applications like "OsApplication_Sub1", "OsApplication_Sub2", "OsApplication_Main", and "OsApplication_Sub1, OsApplication_Sub2".

Index	Short Name	Activation	Priority	Schedule	Stack Size	Accessing Application
0	OsTask_AppModeRequest	① 1	① 250	④ FULL	① 1000	
1	OsTask_BSW_10ms	① 1	① 180	④ FULL	① 4000	OsApplication_Sub1, OsApplication_Sub2
2	OsTask_WdgMAPI	① 1	① 210	④ FULL	① 1000	
3	OsTask_WdgMTestMode	① 1	① 205	④ FULL	① 1000	
4	OsTask_BSW_Init_Main	① 1	① 99	④ FULL	① 4000	
5	OsTask_BSW_Init_Sub1	① 1	① 99	④ FULL	① 800	
6	OsTask_BSW_Init_Sub2	① 1	① 99	④ FULL	① 800	
7	OsTask_BSW_5ms_Mem	① 1	① 198	④ FULL	① 2000	OsApplication_Sub1, OsApplication_Sub2
8	OsTask_NvMCallback	① 1	① 187	④ FULL	① 2000	
9	OsTask_ASW_10ms	① 1	① 140	④ FULL	① 800	OsApplication_Sub1, OsApplication_Sub2
10	OsTask_BSW_10ms_Sub1	① 1	① 180	④ FULL	① 800	
11	OsTask_BSW_10ms_Sub2	① 1	① 180	④ FULL	① 800	
12	OsTask_BSW_5ms_Comp	① 1	① 197	④ FULL	① 4000	
13	OsTask_BSW_RteStart_Sub1	① 1	① 10	④ FULL	① 800	OsApplication_Main [/AUTOSAR/Os/OsAp]
14	OsTask_BSW_RteStop_Sub1	① 1	① 10	④ FULL	① 800	OsApplication_Main [/AUTOSAR/Os/OsAp]
15	OsTask_BSW_RteStart_Sub2	① 1	① 10	④ FULL	① 800	OsApplication_Main [/AUTOSAR/Os/OsAp]
16	OsTask_BSW_RteStop_Sub2	① 1	① 10	④ FULL	① 800	OsApplication_Main [/AUTOSAR/Os/OsAp]
17	OsTask_BSW_Mem_Process	① 1	① 98	④ FULL	① 800	OsApplication_Sub1, OsApplication_Sub2
18	OsTask_RteComTask	① 1	① 30	④ FULL	① 800	
19	OsTask_ASW_5ms	① 1	① 194	④ FULL	① 800	OsApplication_Sub1, OsApplication_Sub2
20	OsTask_ASW_100ms	① 1	① 116	④ FULL	① 800	OsApplication_Sub1, OsApplication_Sub2
21	OsTask_ASW_LinScheduleReq...	① 1	① 221	④ FULL	① 800	
22	OsTask_BSW_FG1_IOManager	① 1	① 195	④ FULL	① 1000	
23	OsTask_BSW_FG2_IOManager	① 1	① 220	④ FULL	① 1000	
24	OsTask_CommCbkTask	① 1	① 200	④ FULL	① 800	

4. The registered task must be registered as a used task in the Application tab.

The screenshot shows the AUTOSAR Com User Manual interface with the 'Application' tab selected. The 'Container Details - OsApplication' panel displays the configuration for the 'OsApplication_Main' container. Key fields include:

- Short Name*: OsApplication_Main
- Core Assignment: 0
- Trusted*: true
- App Alarm Ref.: OsAlarm_BSW_10ms, OsAlarm_BSW_5ms_Mem, OsAlarm_ASW_10ms, OsAlarm_BSW_5ms_Com, OsAlarm_ASW_5ms, OsAI
- App Counter Ref.: OsCounter_Main [/AUTOSAR/Os/OsCounter_Main] (/e_rtu02_tc27x/Configuration/Ecu/Ecud_Os.arxml)
- App Ecuc Partition Ref.: EcucPartition_Main [/AUTOSAR/Ecu/EcucPartitionCollection/EcucPartition_Main] (/e_rtu02_tc27x/Configuration/Ecu/Ecuc
- App Isr Ref.: CANSR4_ISR, CANSR0_ISR, CANSR8_ISR, ADC0SR1_ISR, ADC1SR1_ISR, ADC2SR1_ISR, ADC3SR1_ISR, GTMTIMOSR0_ISR, C
- App Task Ref.: OsTask_BSW_10ms, OsTask_BSW_Init_Main, OsTask_WdgMAPI, OsTask_WdgMTestMode, OsTask_BSW_5ms_Mem, OsTask_A (highlighted with a red box)
- App Resource Ref.: OsResource0, XcpResource0

The 'Hooks' section shows 1 item. The 'To Be Configured:' section includes fields for App Schedule Table Ref. and Restart Task, both with browse buttons.

At the bottom, tabs for Overview, Alarm, App Mode, Application (highlighted with a red box), Counter, Event, Isr, Resource, Schedule Table, Spinlock, Task, and All Contents are visible.

DOCUMENT NUMBER (DOC NO)	SHT/SHTS 101 / 181
-----------------------------	-----------------------

5. In the RTE setting screen, after clicking the Refresh button at the top right, click the select box to display the added task. Select the relevant task and click the Add button.



6. Select Runnable and check if it is normally connected to the task.



7. Extract by clicking ECU Software Components Mapping in the Overview tab of EcuExtract.

DOCUMENT NUMBER (DOC NO)	SHT/SHTS 104 / 181
-----------------------------	-----------------------

8. Go to the Data Mappings tab of EcuExtract, select the added port to add, or click the Add All button at the bottom right to add the added ports in bulk.

[In the case of adding with the Add button, the corresponding Signal on the right must be selected accurately.]



If Data Mapping already exists, delete all DataMappings from All Contents and perform Add All. In case of adding after selecting individually, there is no need to delete.



DOCUMENT NUMBER (DOC NO)	SHT/SHTS 106 / 181
-----------------------------	-----------------------

9. Check if Data Mapping is done normally.

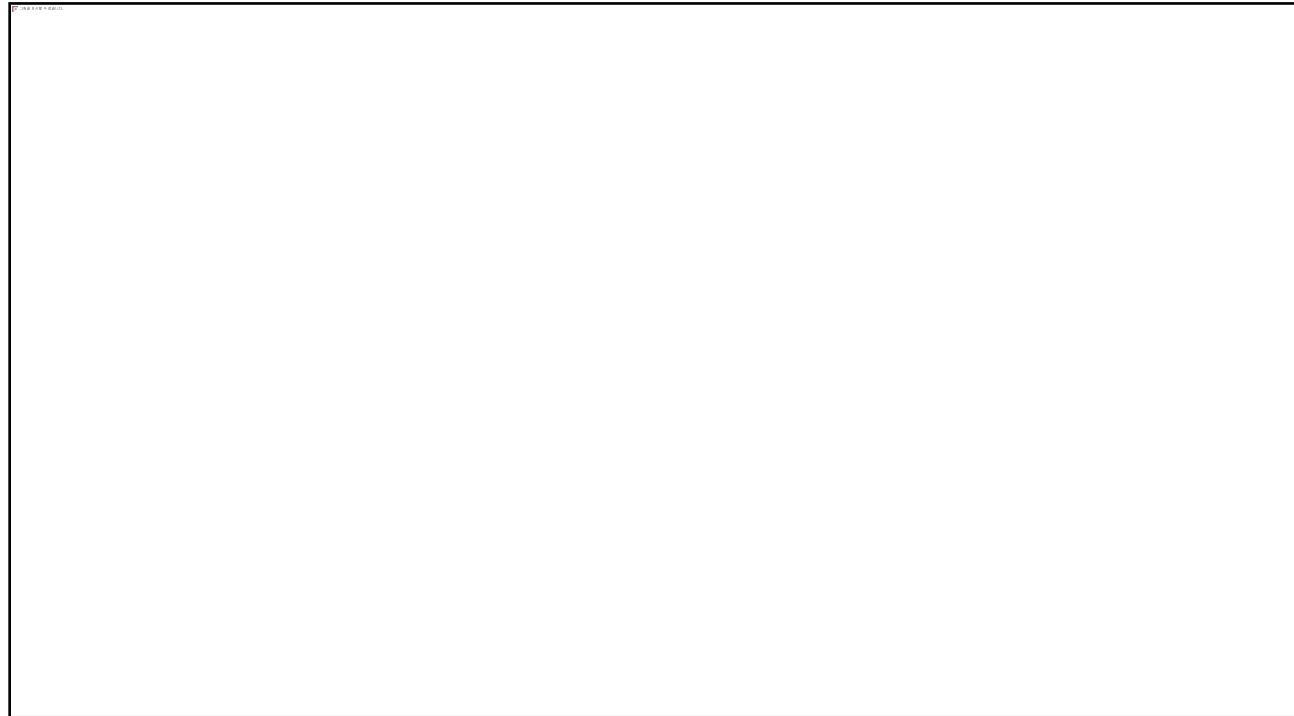


8.11.2 Reception Timeout Notification Setting

8.11.2.1 Configuration

8.11.2.1.1 Com Configuration

1. Select the relevant signal on the Com module setting screen and register the Timeout Notification function.



8.11.2.1.2 SW-C Configuration

In order to use Rx Timeout Notification, the contents of chapter 8.11.1.1 must be set. The description below proceeds under the assumption that 8.11.1.1 is set.

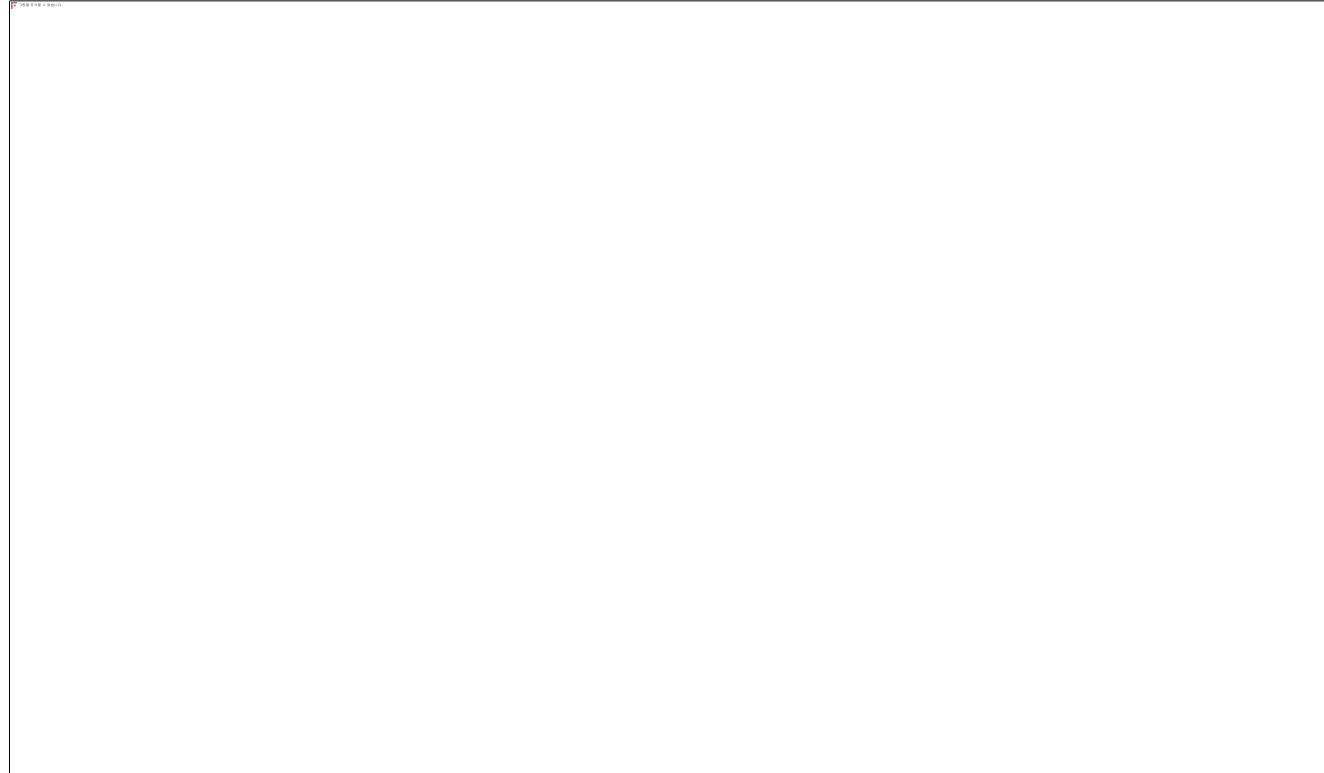
1. Select SW-C to process signal reception notification and move to Runnable tab.

DOCUMENT NUMBER (DOC NO)	SHT/SHTS 108 / 181
-----------------------------	-----------------------

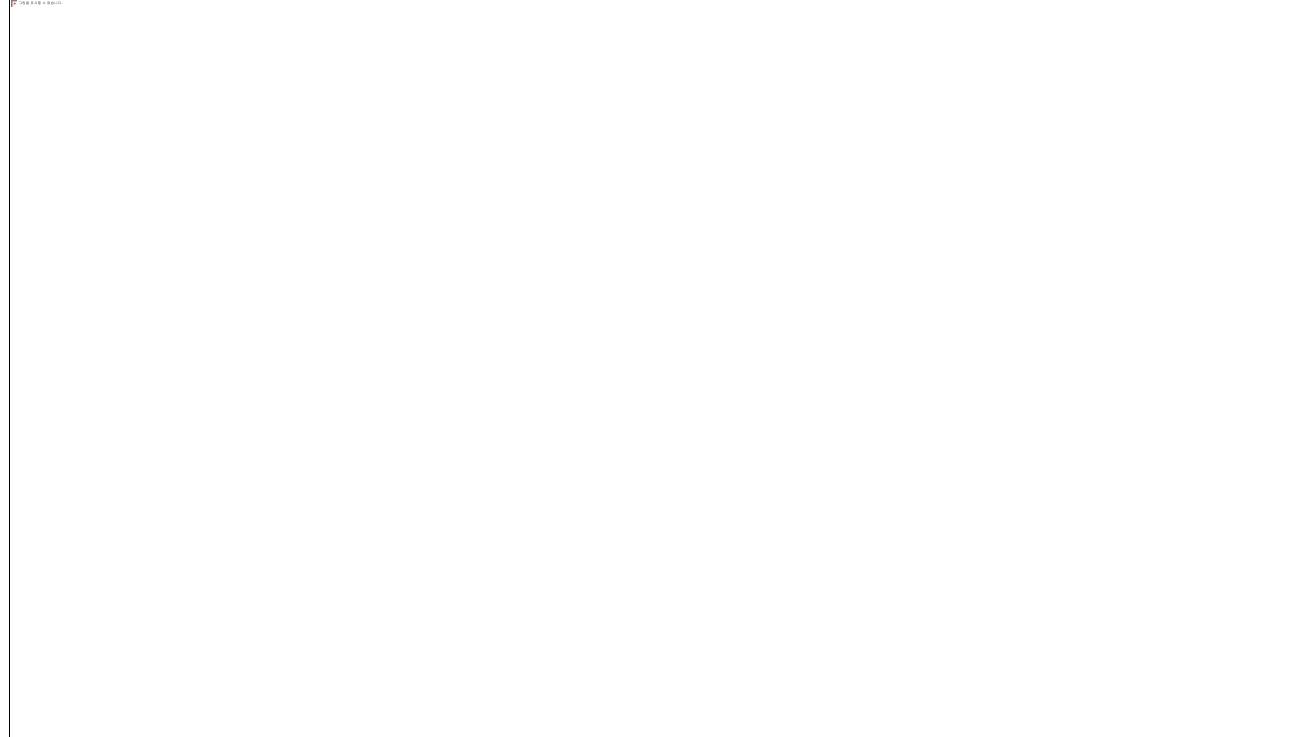
2. Add Runnable to be executed when Timeout Event and select RTE Event.



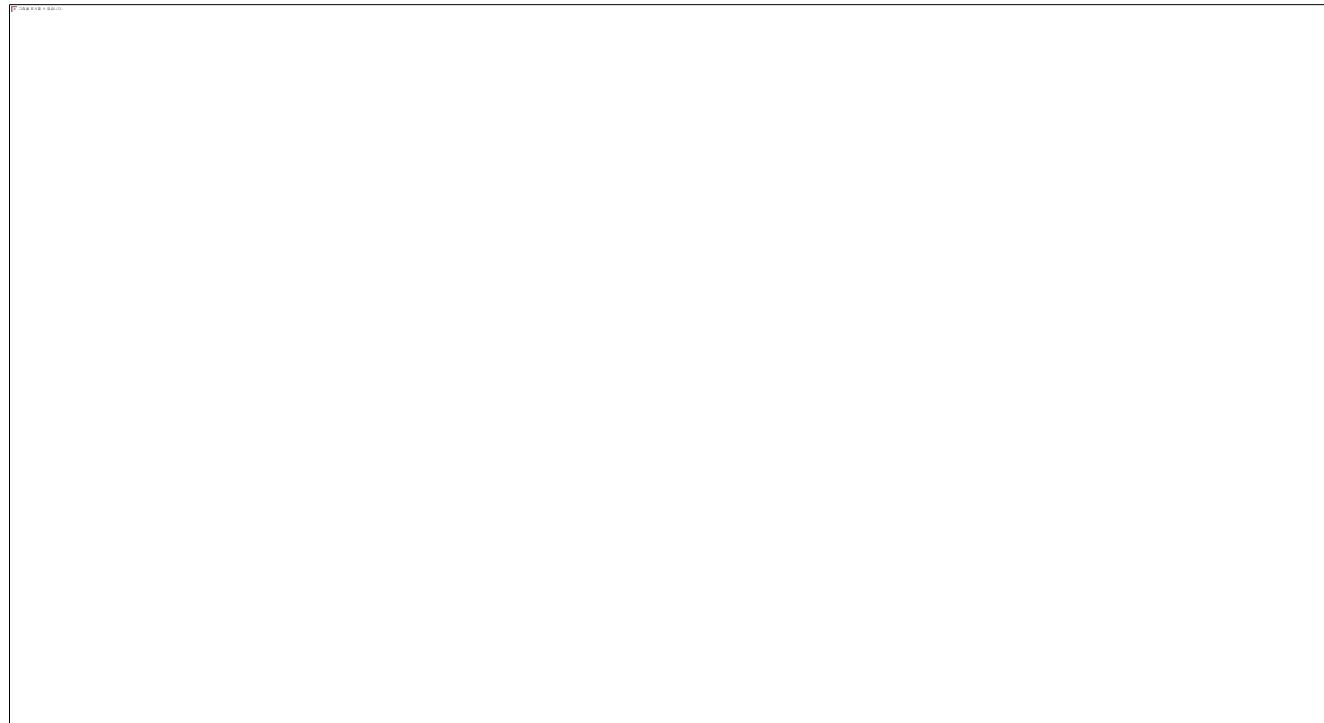
3. Select the applicable Signal of Data Receive Error Event and click the Add button.



4. Check if Event is registered.



5. Go to the All Contents tab and check if the settings are correct.



8.11.2.1.3 Rte/OS Configuration

1. In Rte's Task Mapping tab, select a Runnable to assign a task to, map the task and click the Add button.



2. Check if the assignment is normally done.



8.11.3 Reception Invalid Notification Setting

8.11.3.1 Configuration

Invalid Notification settings are the same as in Chapter 8.11.2 except for Com Configuration.

8.11.3.1.1 Com Configuration

1. Select the relevant signal in the COM module setting screen and register the Invalid Notification function.



8.11.3.1.2 SW-C Configuration

Same setting method as 8.11.2.1.2

8.11.3.1.3 Rte/OS Configuration

Same setting method as 8.11.2.1.3

8.12 Signal Port Connection and Transmission Signal Notification Settings

Reference:

The setting guide below is for Odin 2017a.

After Odin 2017b (SP1), it is set as a separate dedicated UI. (8.13,8.14)

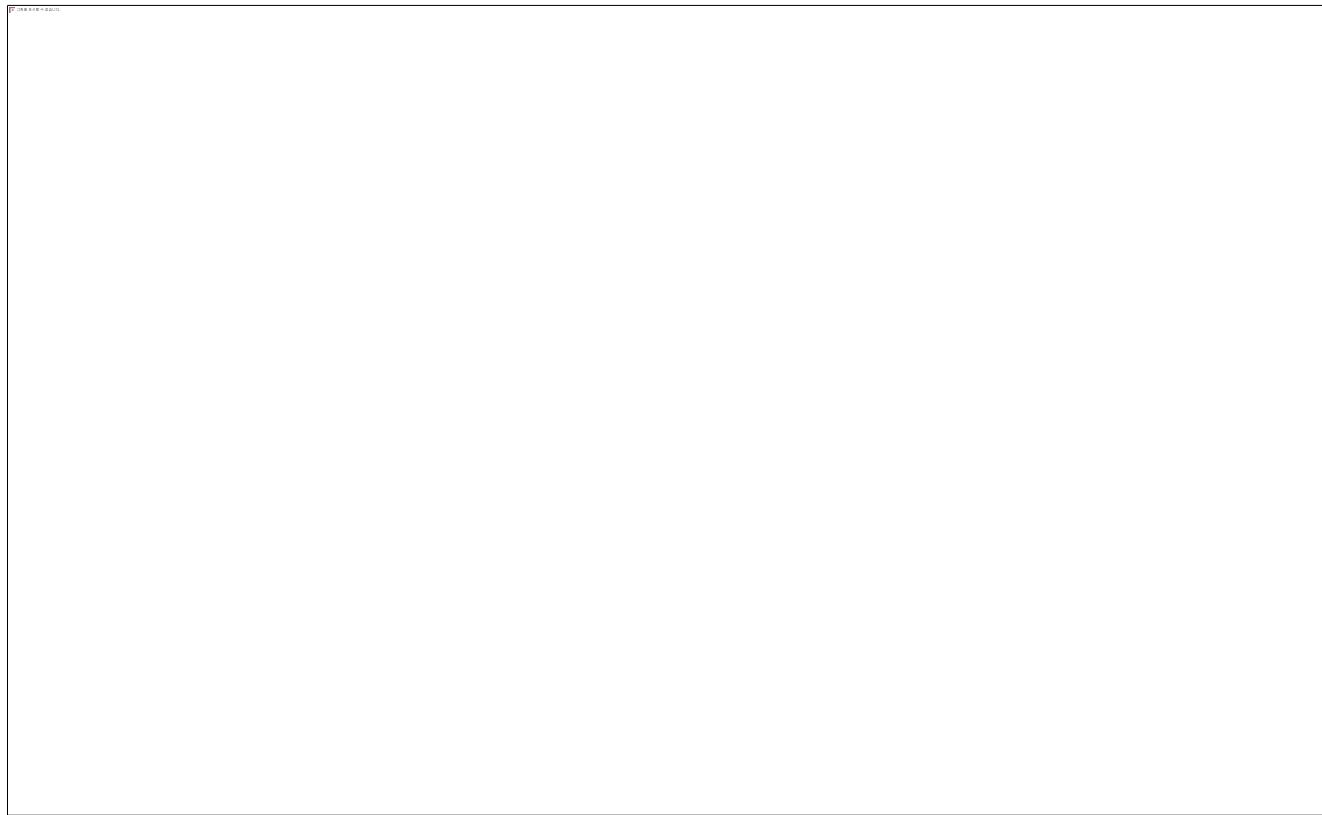
8.12.1 Transmission Notification Setting

8.12.1.1 Configuration

8.12.1.1.1 Com Configuration

1. Select the relevant signal in the Com module setting screen and register the Notification function.

DOCUMENT NUMBER (DOC NO)	SHT/SHTS 114 / 181
-----------------------------	-----------------------



8.12.1.1.2 SW-C Configuration

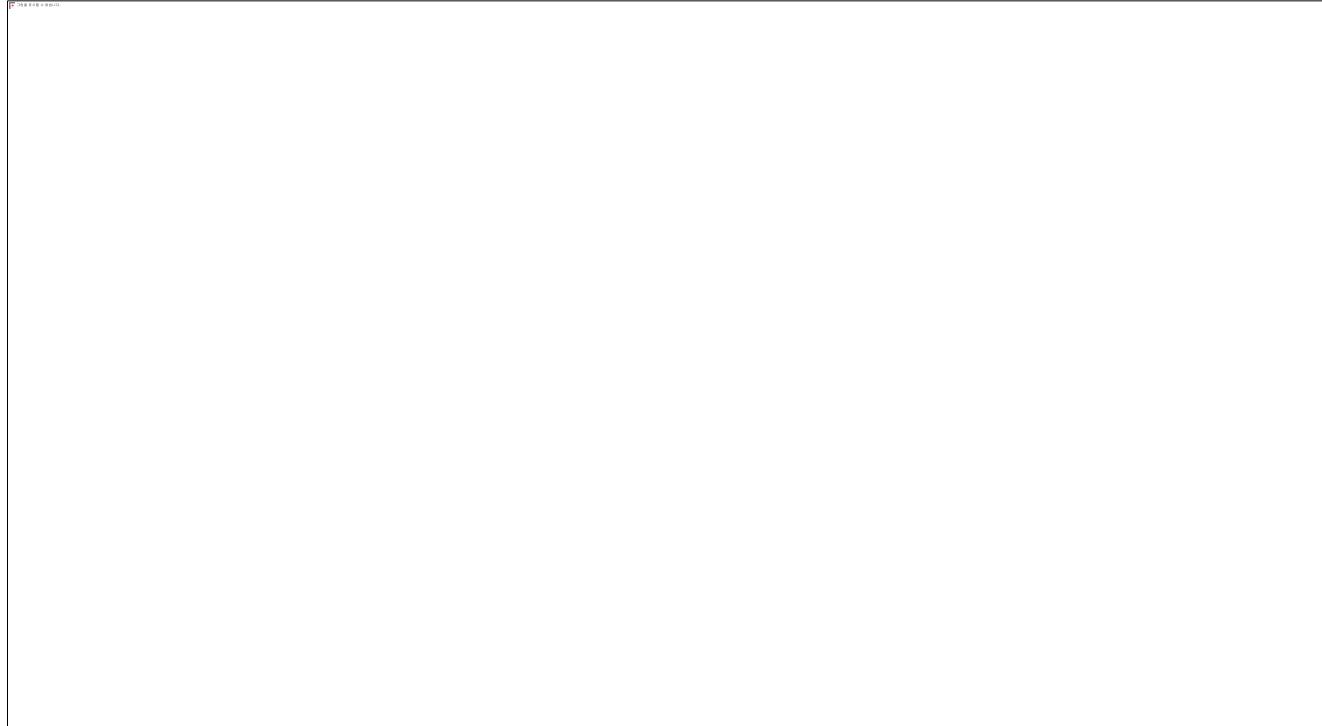
1. Select SW-C to process signal and create Runnable.



2. Go to the Port tab and click the Add button.



3. Select the signal to register the port, and select the runnable to use the port in the select box below.



4. Check if it has been added normally in the All Contents tab.



5. Go to Port tab, select signal details, and check Transmission Acknowledge. If only Notification is required, the Timeout value can be set to 0.



6. In the Runnable tab, select the corresponding signal of Data Send Completed Event and click the Add button.



7. Check if the event is registered.



8. Go to the All Contents tab and check whether the event is set normally.



8.12.1.1.3 Rte/OS Configuration

1. Go to Rte settings and go to Task Mapping tab. Select Sw Component Instance and select SW-C set above in the select box.



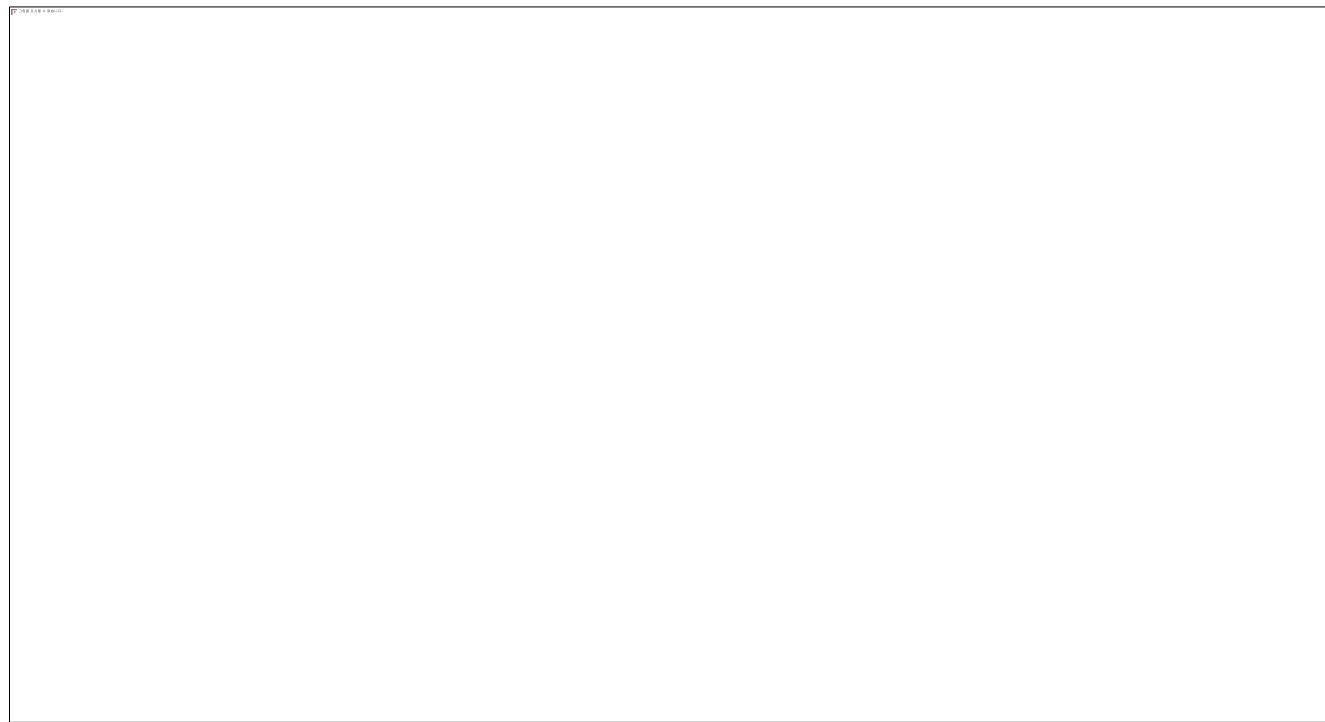
2. In the Event Table, select the Runnable to be mapped, and if there is a previously created task in the OsTask Mapping Table, select it through the select box. If not, add the task by clicking the Add button on the right.

DOCUMENT NUMBER (DOC NO)	SHT/SHTS 123 / 181
-----------------------------	-----------------------

Select a task and click the Add button.[Refer to 9.9.1.1.3 for how to add a task.]



3. Extract is performed by clicking ECU Software Components Mapping in the Overview tab of EcuExtract.



DOCUMENT NUMBER (DOC NO)	SHT/SHTS 125 / 181
-----------------------------	-----------------------

4. Go to the Data Mappings tab of EcuExtract, select the added port to add, or click the Add All button at the bottom right to add the added ports in bulk.

[In the case of adding with the Add button, the corresponding Signal on the right must be selected accurately.]



If Data Mapping already exists, delete all DataMappings from All Contents and perform Add All. In case of adding after selecting individually, there is no need to delete.



9. Check if Data Mapping is done normally.



8.12.2 Transmission Timeout Notification Settings

8.12.2.1 Configuration

In order to process Timeout Notification, Transmission Notification must be set by default. For related settings, refer to 8.12.1.1.

8.12.2.1.1 Com Configuration

1. Select the relevant signal on the Com module setting screen and set Timeout Notification and Timeout time.



8.12.2.1.2 SW-C Configuration

1. Select SW-C to process signal, move to Port tab, and set Timeout value to the value set in Com module. Runnable and Port creation are explained under the assumption that it has already been created.



2. Timeout Event uses the same Event as Transmission Notification, so there is no additional setting.

8.12.2.1.3 Rte/OS Configuration

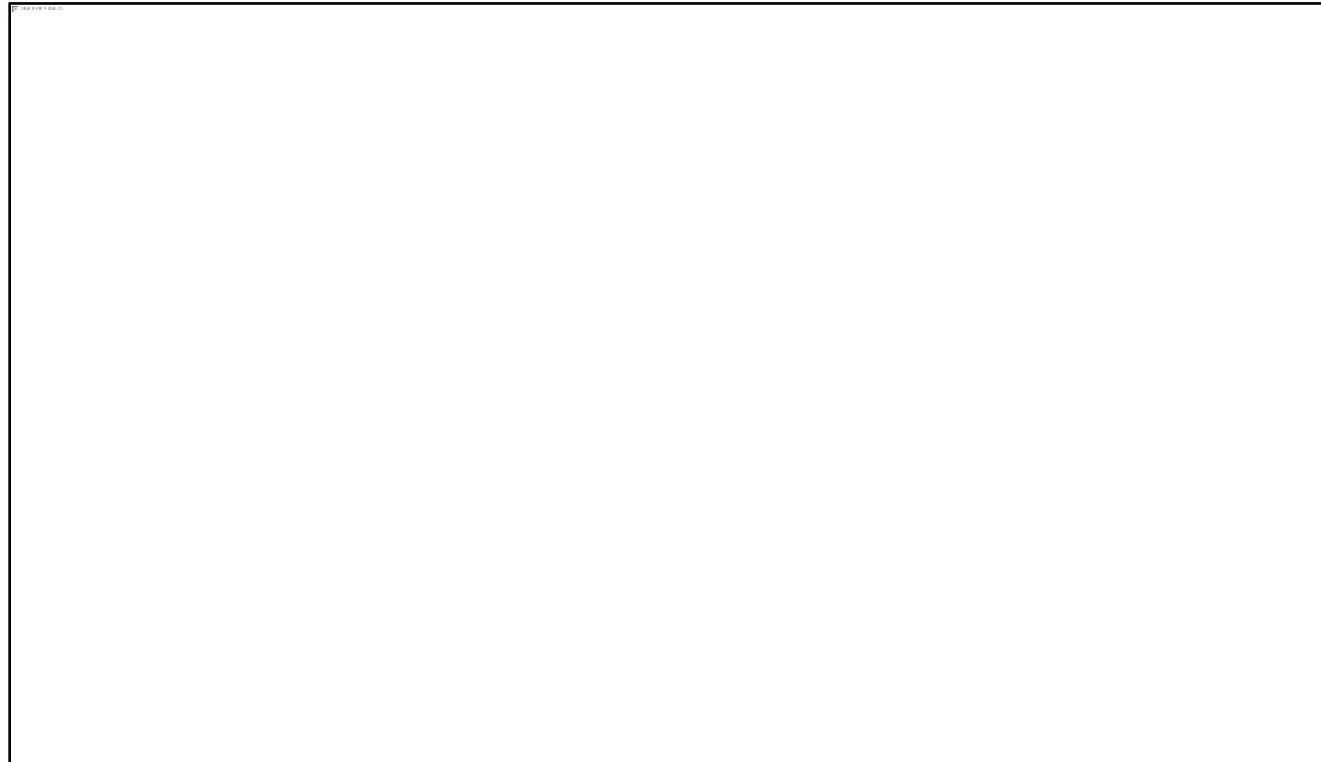
Since Rte/OS setting has already been set in the process of 8.12.1.1, there is no additional work to be done.

8.13 Alternative settings for 8.10, 8.11 (Rx) using Create Ports From Network

The setting guide below is a setting that replaces the contents of 8.10 and 8.11.

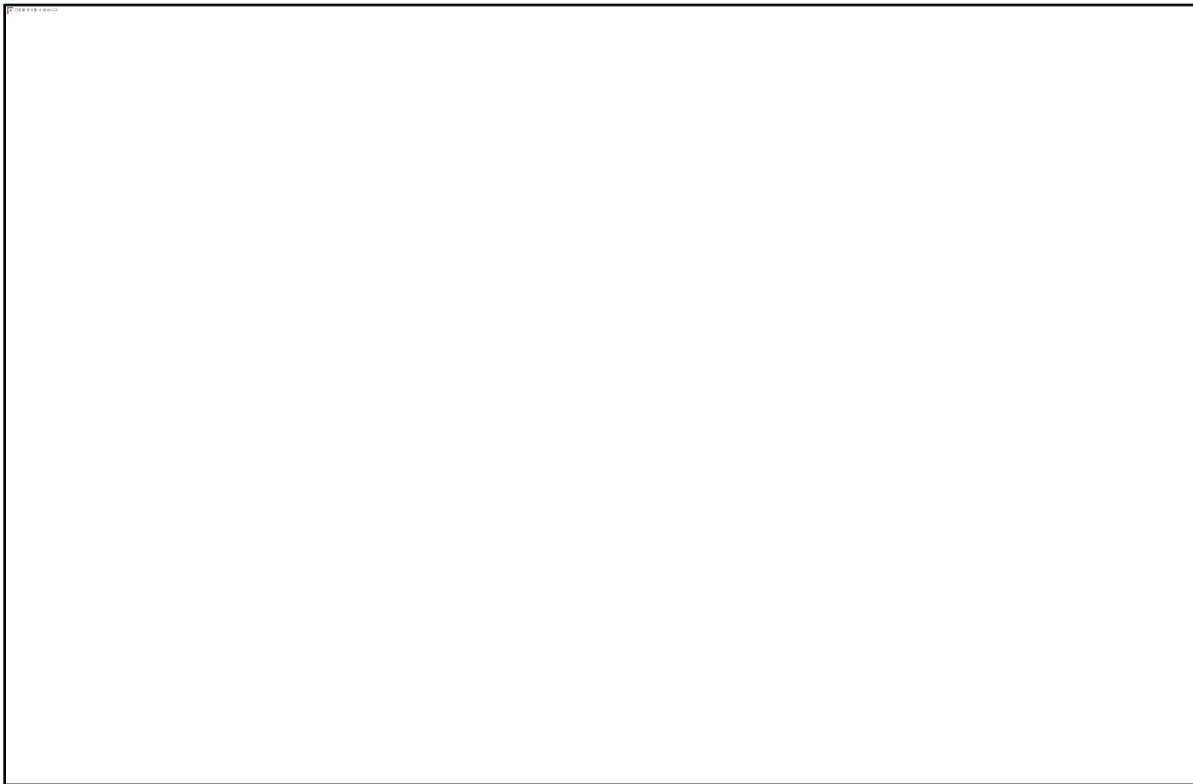
8.13.1 Com Harmonize

Click the Configure ECU and Generate Code button at the top to open the EcucValueCollection editor, click Generate ECU Configuration to run the Com module harmonize.



8.13.2 Port and Comspec Creation

Click Configure System and Extract ECU to open ECUExtract editor, select Data Mappings.

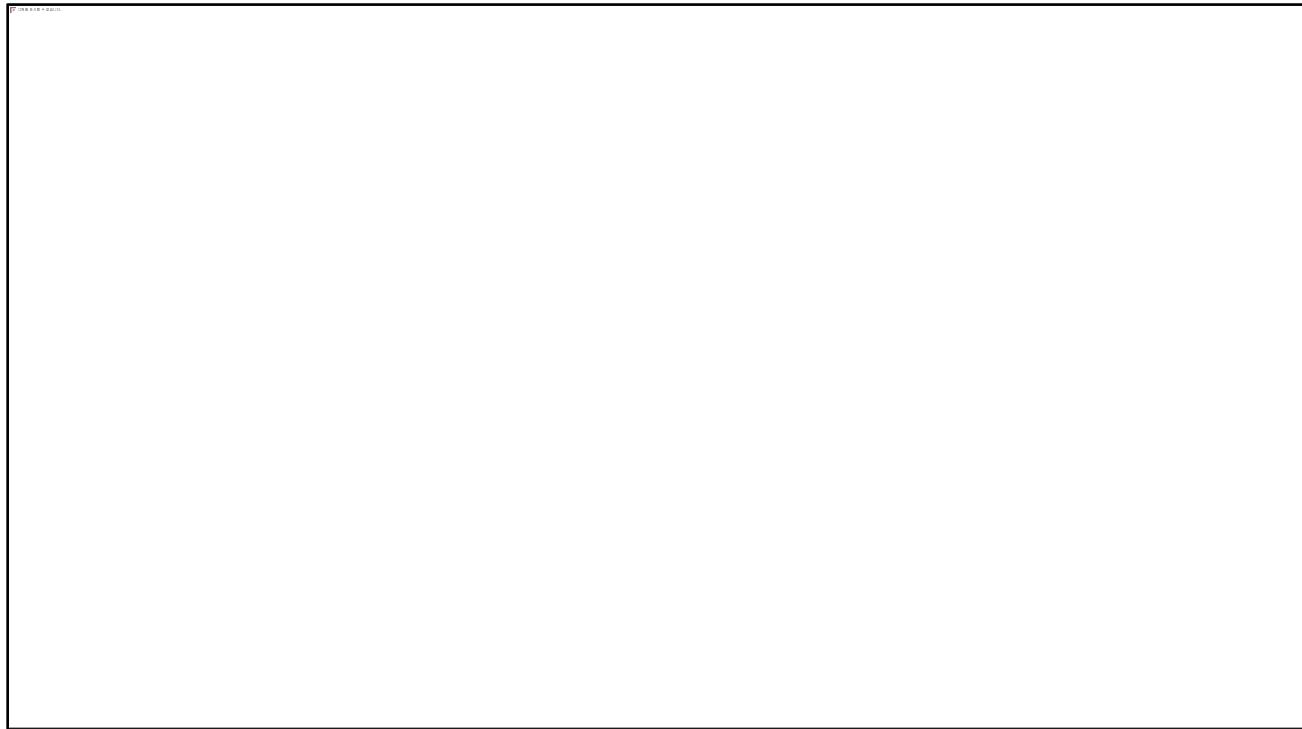


8.13.2.1 Select SwComponent to process Signal in Port& (Data Elements: mapped to ComSpec) column.

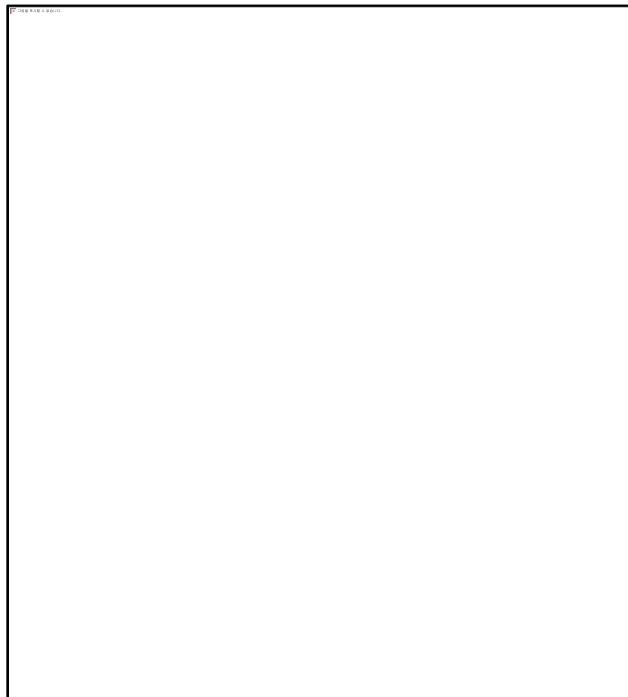
If you select a SwComponent that has already been registered, refer to 8.13.2.2.

If you want to create Port and ComSpec after creating a new SwComponent, refer to the following 9.11.2.1.1~9.11.2.1.5.

8.13.2.1.1 Double-click RootSwCompositionPrototype.



8.13.2.1.2 When the Composition Editor opens as follows, click the + button to add the newly created SwComponent.



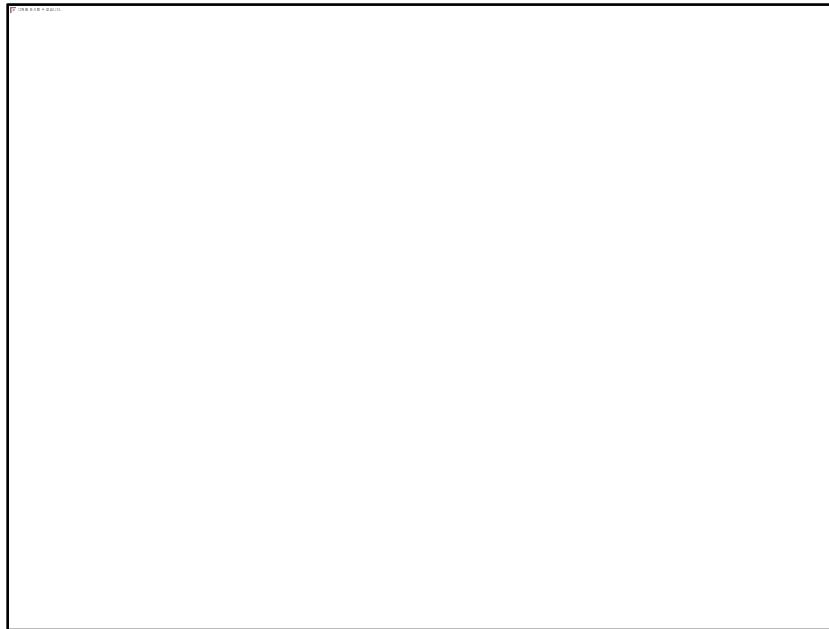
8.13.2.1.3 Execute ECU Extract through ECU Software Components Mapping in OverView page of EcuExtract editor.



8.13.2.1.4 Return to the DataMapping page, and click the Refresh button in the upper right corner.



8.13.2.1.5 A new SWC is registered as follows

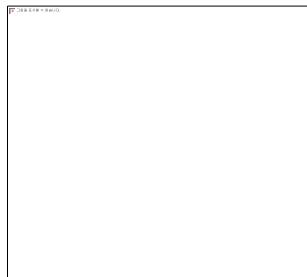


8.13.2.2 Double-click the selected SWC. Alternatively, click and then click the Create Ports From Network button at the bottom left.

In the example, we double-clicked ApplictionSwComponentType0.

8.13.2.3 When the Cluster Selection dialog appears, select the Cluster with the processed signal.

In this example, only the Cluster called Project is selected.



8.13.2.4 Select the desired item in the following editor to create the signal to be processed and the port to be connected.

When the Editor opens, select the Rx tab.



In the example, the signal to be processed is ComISignal_ECU2_Msg_P1_ECU2_Msg_P1_Sig1.

I checked as below.



It is useful to use the function below to find the text you want.

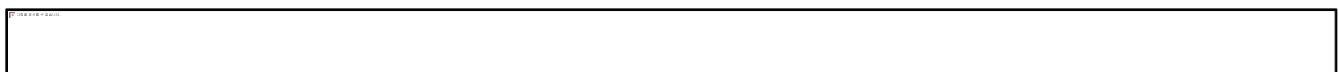
-When clicking the first column (:Sw Component~) Header, it is possible to sort forward/reverse the name value.

- When the second column (:Position) header is clicked, the alignment of the first column is fixed and the position value is

Forward/reverse sorting is possible.

-If any Cell of Editor has focus, you can find the desired text value by pressing Ctrl + F.

If you select a cell, a Validation Message appears at the bottom about the current state of the selected cell.



In the case of the example, since DataMapping is not connected, the contents of Timeout, Data Access Runnable, DSP, and RTE Event cannot be modified. Only Port and Comspec corresponding to the selected name can be created.

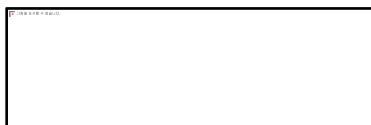
8.13.2.5 Click the OK button.

8.13.3 Data Mapping

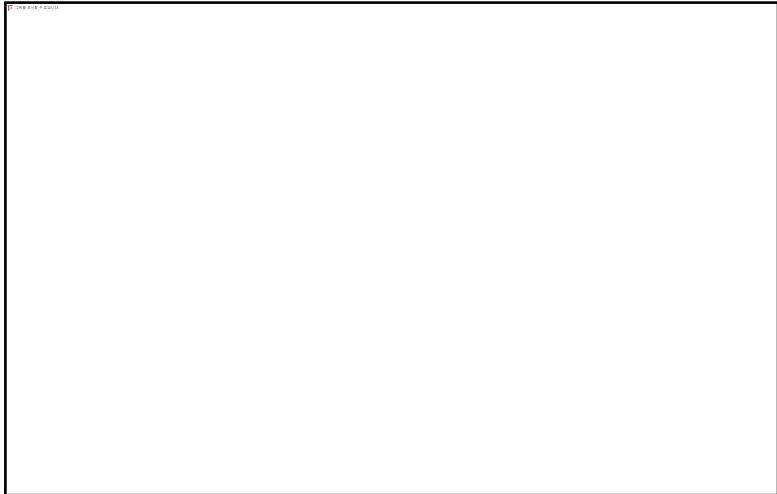
8.13.3.1 Click the data element ECU2_Msg_P1_Sig1 under the created Port. As a result, a connectable ISignal appears as follows.



This is set as the default option, but if you do not want it, click the lower button (:Select Options) in the upper right



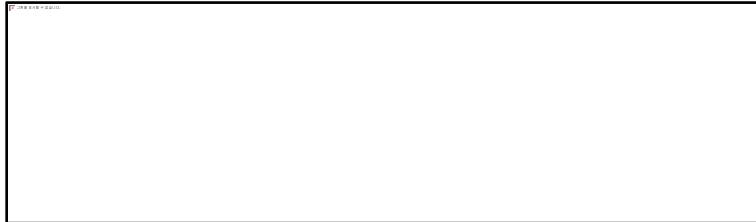
Cancel the Search Option in the pop-up window that appears as follows.



8.13.3.2 Create DataMapping through Add or Add All.



When DataMapping is created, the background color is changed as follows, and the number in [] is changed.



8.13.4 Runnable Setting

Double-click SwComponent on the DataMapping page.

In the example, double-click ApplicationSwComponentType0[1/1] and select cluster as the project.

Select the Rx tab.



The background color becomes gray, and the contents of Timeout, Data Access Runnable, DSP, and RTE Event can be modified. If there is no Runnable in SWC, only Default Runnable named NewRunnable appears.

If a Runnable has already been created in SWC, when you click it, you will see candidates from which you can select from Runnable in SWC.

Since there is currently no Runnable, only NewRunnable is shown.



To create a new Runnable, double-click or double-click the Data Access Runnable column in succession.

When the focus is activated, a Text value is entered.

In the example, Test_P1_Sig1_Notification is entered.



8.13.5 DataReceivePointByArgument/DataReceivePointByValue Setting

Select the Runnable name and click on the DRPBA DRPBV column, Value, you will see DRPBA, DRPBV, NONE, etc.

When you want to create DataReceivePointByArgument in the selected Runnable, DRPBA,

When you want to create DataReceivePointByValue in the selected Runnable, DRPBV,

Select NONE to delete all Variable Access to the selected Runnable.



In our example, we selected DRPBA and clicked OK.

8.13.6 Check your Runnable settings

Double-click on a port to move to the SwComponent editor.



You can check the creation as follows.

(If you create a Runnable through Create Ports From Network, make the Short Name and Symbol the same name.
generate)

8.13.7 DataReceivedEvent settings

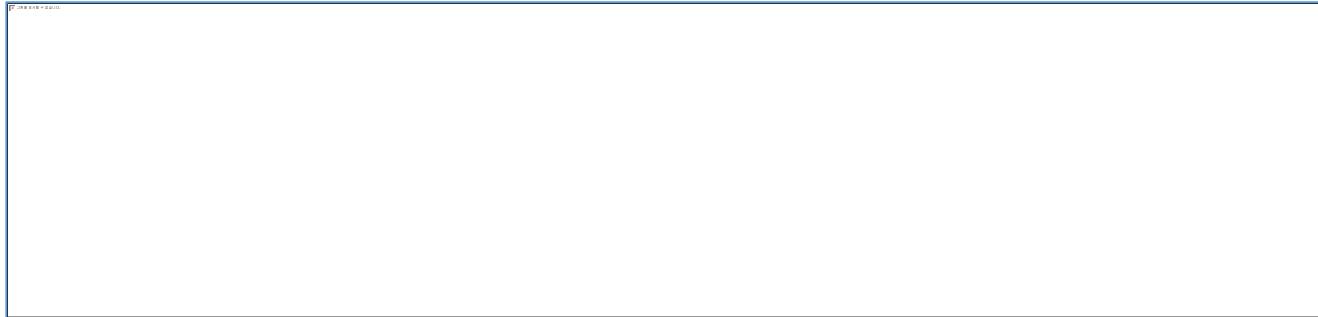
After closing the editor by clicking the OK button, double-click ApplicationSwComponentType0[1/1] on the DataMapping page and select the cluster as the project. Select the Rx tab.

Since it was already created in 8.13.6, it opens with a DRPBA state like this:

If you want to create DRE, select Runnable to handle event in Start on DRE item.

In the example, since there are no other Runnables in SwComponent, the runnable created in 9.11.4 is selected.

If another Runnable has been created, it is shown as a candidate. However, in the case of DRE, the function to directly create a set Runnable is not currently supported, as in 9.11.4, which directly creates a Runnable. (Odin 2017b SP1 version)



Click the OK button.

8.13.8 Checking the DataReceivedEvent

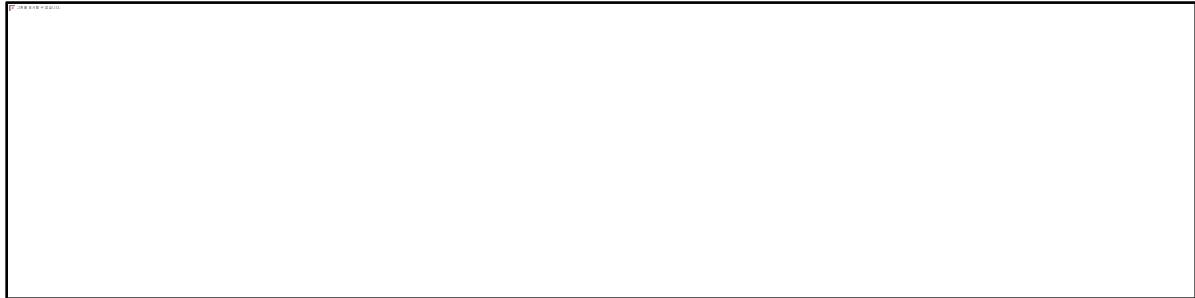
In this section, check how the settings up to 8.13.7 are reflected.

If you double-click the port item in the DataMappings page, the ApplicationSwComponentType0 editor

Go to the All Contents page. You can check the settings on the page as follows.

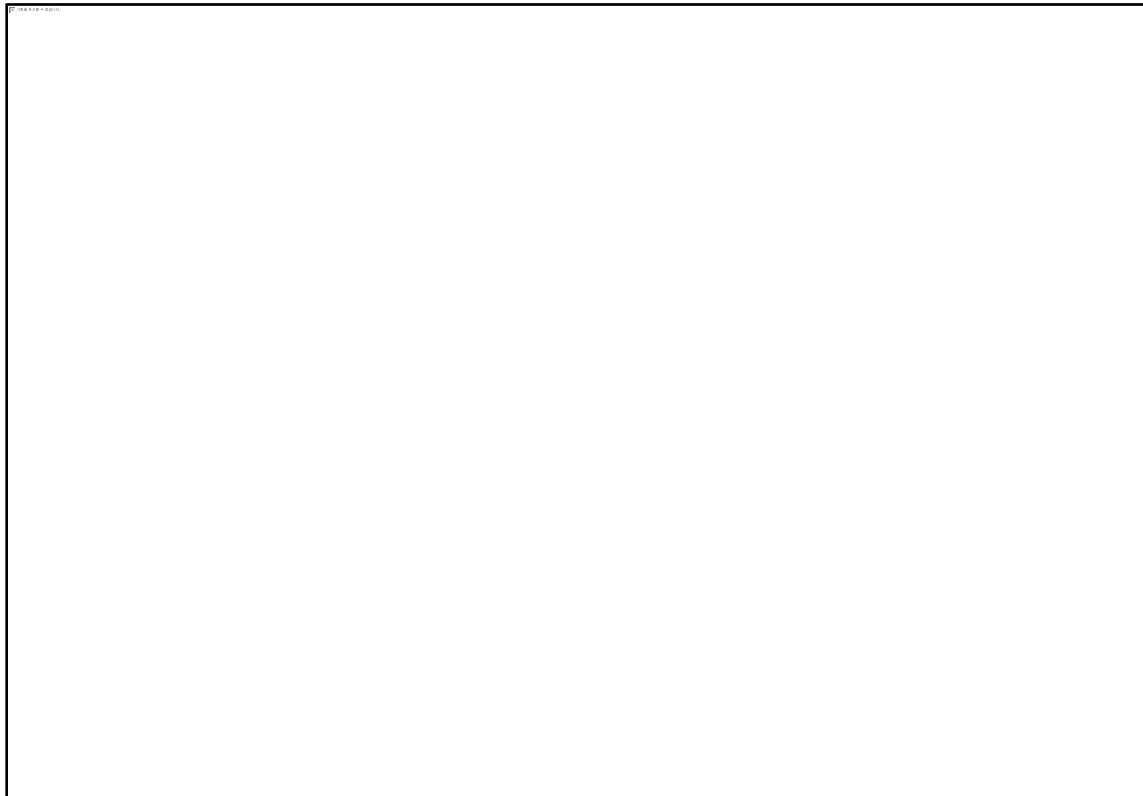
-DRE, DRPBA, Runnable, Port, Comspec





If you go to the COM module editor and check it, the call back contents are filled as follows.

-Com: Notification



8.13.9 Setting DataReceiveErrorEvent

On the DataMapping page, double-click ApplicationSwComponentType0[1/1] and select cluster as project. Select the Rx tab.

Since there are already created items while performing up to 8.13.8,

Data Access Runnable, DRPBV, and DRE are set and displayed.

If you want to create DREE, select Runnable to process event from Start on DREE item.

In the example, since there are no other Runnables in SwComponent, the runnable created in 9.11.4 is selected.

If another Runnable has been created, it is shown as a candidate. However, in the case of DREE, the function to directly create a set Runnable is not currently supported, as in 9.11.4, which directly creates a Runnable. (Odin 2017b SP1 version)

Leave the Timeout value as 0.7.

Click the OK button.

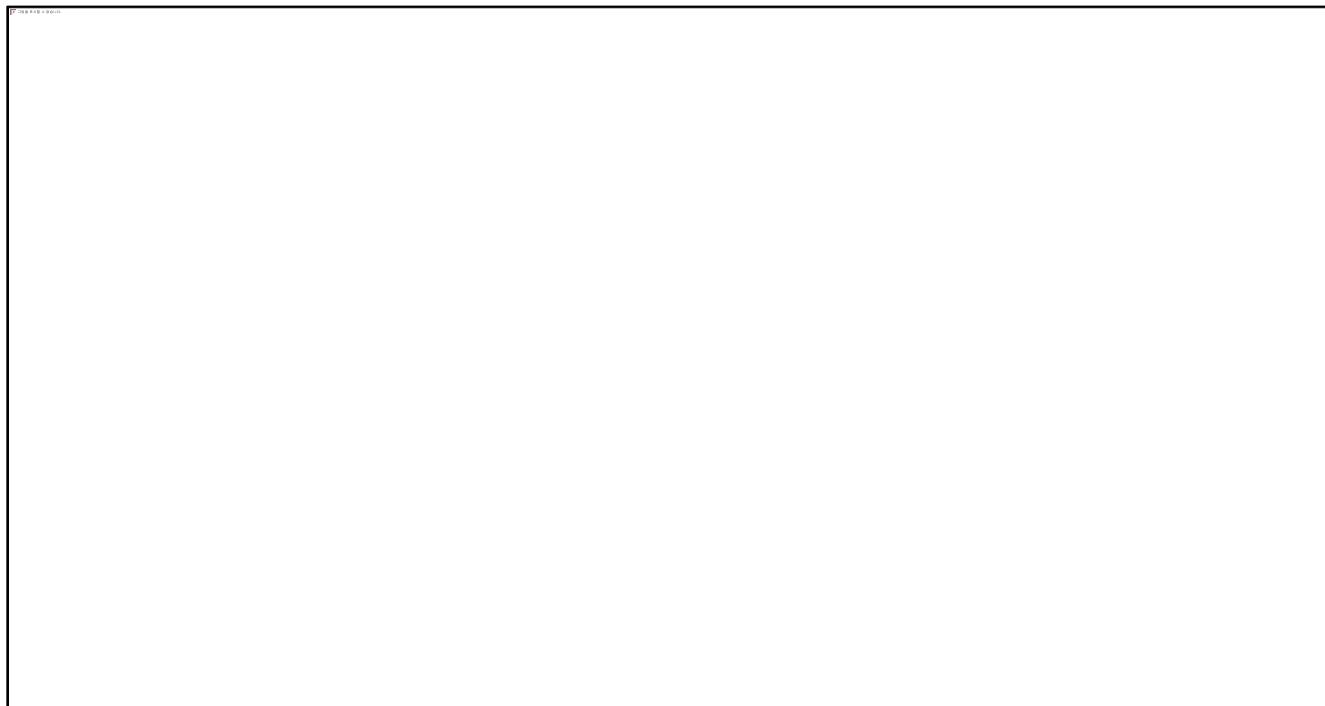
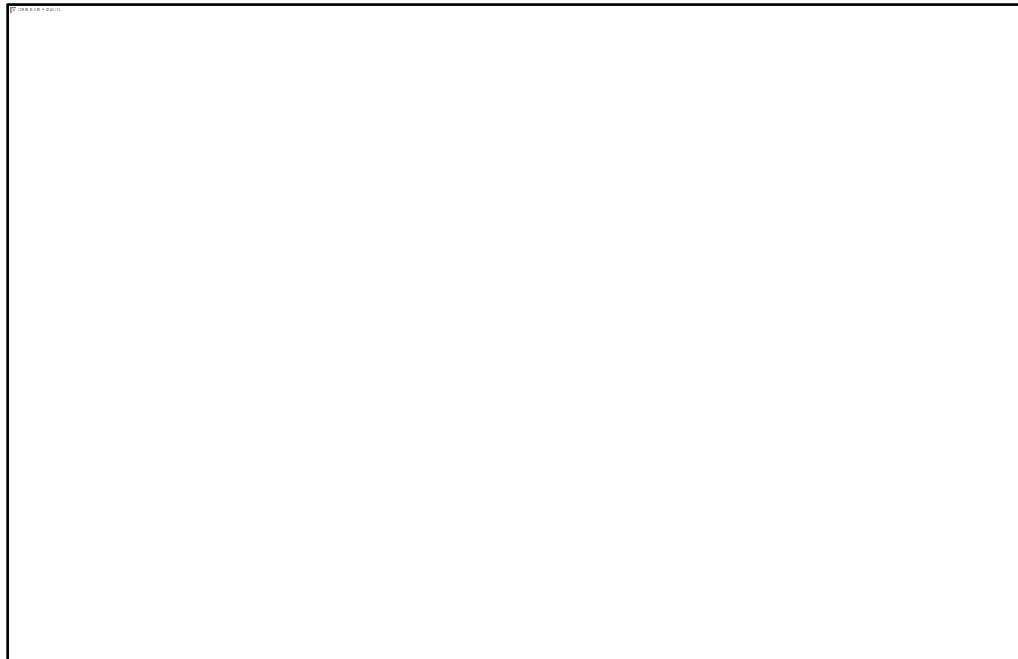
8.13.10 Check the DataReceiveErrorEvent setting

In this section, check how the settings up to 8.13.9 are reflected.

If you double-click the port item in the DataMappings page, the ApplicationSwComponentType0 editor

Go to the All Contents page. You can check the setting as follows.

-DREE, DRE, DRPBA, Runnable, Port, Comspec, Alive Timeout



Because the DREE value is created (a required condition), it synchronizes the timeout value of the COM module and the timeout value of the SWC.

Since the timeout value of the Com module was already set to the DB value of 0.7 through Com Harmonize of 9.11.1, the Alive timeout was set to 700ms with the same value on the SWC side.

If you go to the COM module editor and check it, the call back contents are filled as follows.

-Com: Notification, Timeout Notification, Timeout (already existed)



8.13.11 Timeout setting and confirmation

On the DataMapping page, double-click ApplicationSwComponentType0[1/1] and select cluster as project. Select the Rx tab.

After performing up to 9.11.9, the following is displayed.

When you click the Timeout value, you can change it as follows.



If activation is not performed as above when clicking on the Timeout value, refer to the Validation content at the bottom.

In the above case, there is no problem with the setting value, so ‘-’ appears as a result.

If there is a problem, a text message appears in Validation. Please refer to the contents of the message and modify it.

Synchronization of timeout values is performed when there is DREE. If you click OK after changing the value to 1.5 as follows

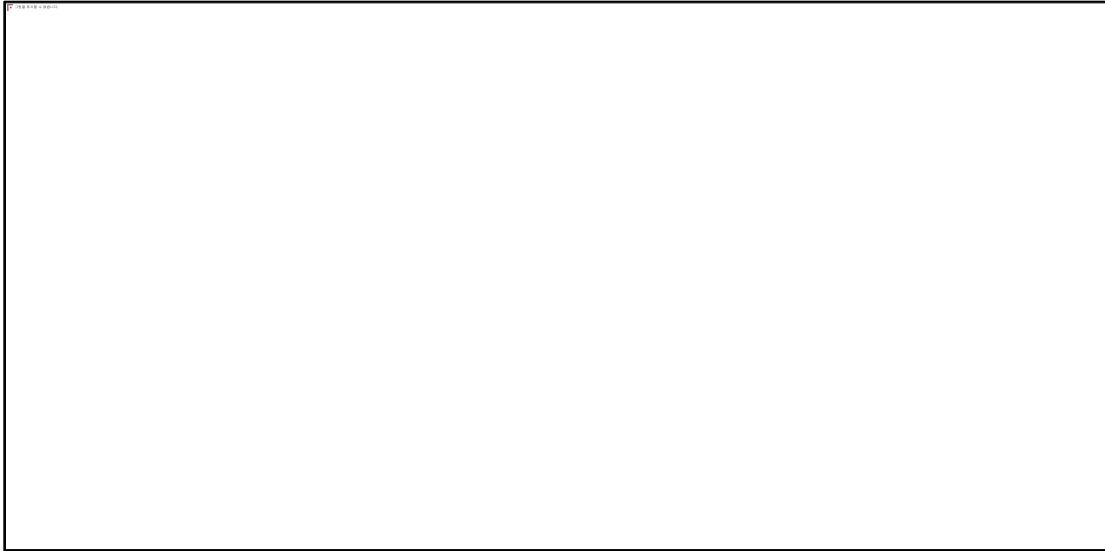
If you check the reflected value, the Alive timeout value on the SWC side becomes 1.5s as follows.

If the timeout value is changed without DREE in the following case,

The timeout value of the Com module is also set to 1.5.

If the timeout value is changed without DREE in the following case,

The Alive timeout value on SWC side disappears (you can see that the DREE event is also deleted).



Only the timeout value on the COM side is changed as follows, and the timeout Notification value is also deleted.



8.13.12 TaskMapping

Click the button below to move to the Task Mapping page and map the event to the Task.



For setting details, refer to 8.10.1.1.3. The method is the same.

8.14 Alternative settings for 8.10, 8.12 (Tx) using 8.14 Create Ports From Network

The setting guide below is a method of replacing the contents set in 8.10 and 8.12 using Create Ports From Network.

8.14.1 Com Harmonize

It is the same as in 8.13.1.

8.14.2 Creating Ports and Comspecs

It is the same as the contents of 8.13.2.1~8.13.2.3.

8.14.2.1 Select the desired item in the following editor to create the signal to be processed and the port to be connected.

In the example, the signal to be processed is ComISignal_ECU1_Msg_GW1_ECU1_Msg_GW1_Sig2.

I checked as below.



It is useful to use the function below to find the text you want.

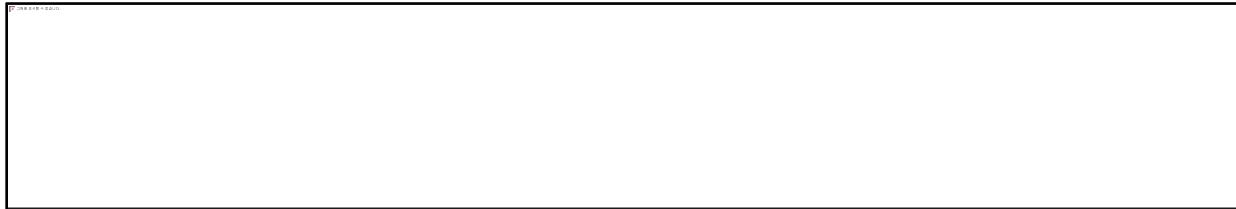
-When clicking the first column (Sw Component~) Header, it is possible to sort forward/reverse the name value.

- When the second column (Position) header is clicked, the alignment of the first column is fixed based on the Position value.

Forward/reverse sorting is possible.

-If any Cell of Editor has focus, you can find the desired text value by pressing Ctrl + F.

If you select a cell, a Validation Message appears at the bottom about the current state of the selected cell.



In the case of the example, since DataMapping is not connected, the contents of Timeout, Data Access Runnable, DSP, and RTE Event cannot be modified. Only Port and Comspec corresponding to the selected name can be created.

8.14.2.2 Click the OK button.



A notification window like the one above appears. If you do not want the current modifications to be reflected, click Cancel or X.

If you have already clicked OK and the result has been reflected, you can cancel the reflection through Undo on the menu bar as follows.



8.14.2.3 The created content is reflected and the following appears.



8.14.3 Data Mapping

8.14.3.1 Click the data element ECU1_Msg_GW1_Sig2 under the created Port. As a result, an ISignal that can be connected as follows appears.



You can also hide connectable ISignal by referring to # of 8.11.3.1.

8.14.3.2 Create DataMapping through Add or Add All.



When DataMapping is created, the background color is changed as follows, and the number in [] is changed.



8.14.4 Runnable settings

Double-click SwComponent on the DataMapping page.

In the example, double-click ApplicationSwComponentType0[1/1] and select cluster as the project.



The background color becomes gray, and the contents of Timeout, Data Access Runnable, DSP, and RTE Event can be modified. If there is no Runnable in SWC, only Default Runnable named NewRunnable appears.

If a Runnable has already been created in SWC, when you click it, you will see candidates from which you can select from Runnable in SWC.

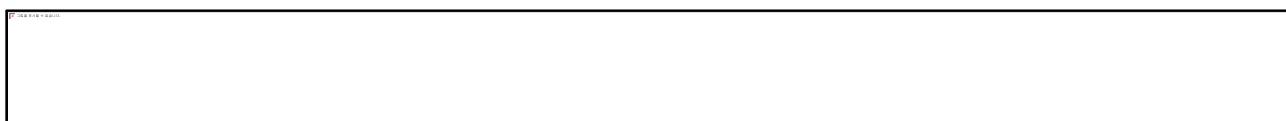
Since there is currently no Runnable, only NewRunnable is shown.



To create a new Runnable, double-click or double-click the Data Access Runnable column in succession.

When the focus is activated, a Text value is entered.

In the example, Test_GW1_Sig2_Notification is entered.



8.14.5 DataSendPoint settings.

Check DSP and click OK.



8.14.6 Runnable settings

Double-click on a port to move to the SwComponent editor.

In the example, we double-clicked on port ECU1_Msg_GW1[1/1].



You can check the creation as follows.



(If you create a Runnable through Create Ports From Network, create the same Short Name and Symbol names.)

8.14.7 Setting DataSendCompletedEvent

If you click the OK button to close and reopen the Editor, it opens with DSP checked as follows because it has already been created. Note) If you click OK after unchecking the same items again, the DSP will be erased.



If you want to create DSCE, select Runnable to handle event in Start on DSCE item.

In the example, since there are no other Runnables in SwComponent, the runnable created in 9.12.4 is selected.

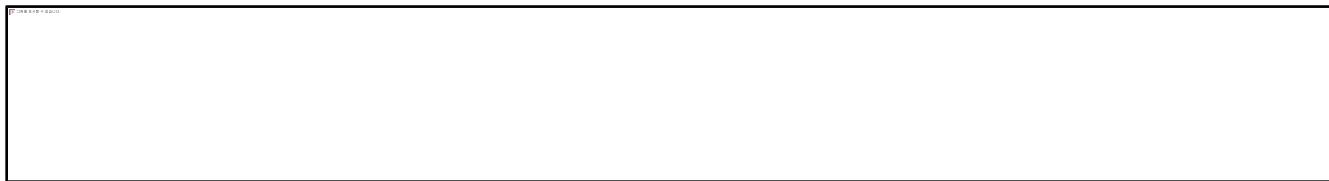
If another Runnable has been created, it is shown as a candidate. However, in the case of DSCE, the function to directly create a set Runnable is not currently supported, as in 9.12.4, which directly creates a Runnable. (Odin 2017b SP1 version)



8.14.8 Timeout setting

If DSCE value is set (required condition), Timeout value can be changed. In our example, we left it as 0.0.

Finally, the timeout value of the SWC and the timeout value of the Com module are synchronized with the value entered in the Timeout value.



Click OK.

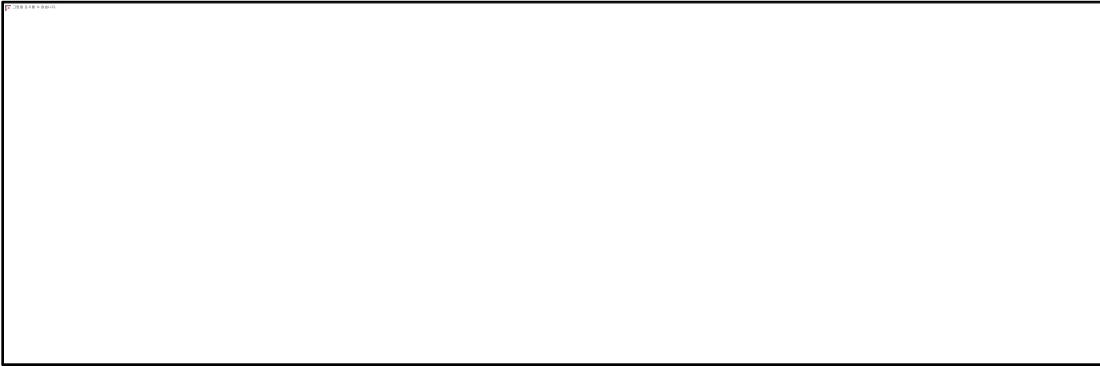
8.14.9 DataSendCompletedEvent, Timeout setting

In this section, check how the settings from 8.14.4 to 8.14.8 are reflected.

If you double-click the port item in the DataMappings page, the ApplicationSwComponentType0 editor

Go to the All Contents page. You can check the setting as follows.

-DSCE, DSP, Runnable, Port, Comspec

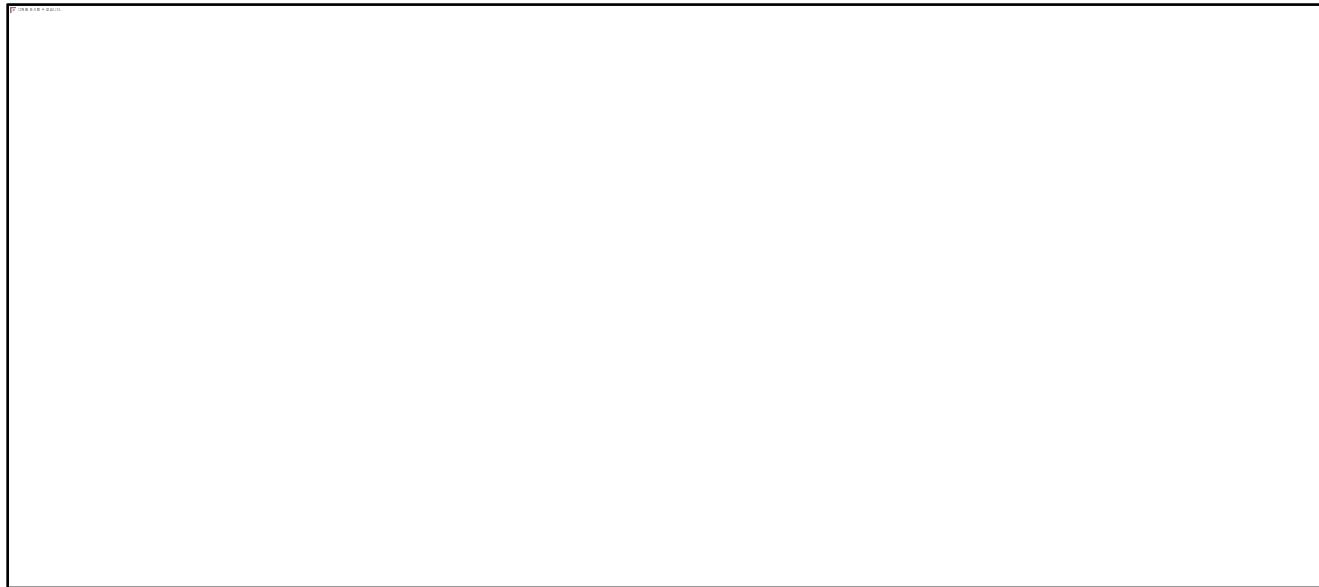


-Comspec: Transmission Ack, timeout



If you go to the COM module editor and check it, the call back contents are filled as follows.

-Com: Error Notification, Notification, Timeout



8.14.10 TaskMapping

The contents of 8.13.12 are the same.

8.15 UINT8_N, UINT8_DYN Signal setting

Reference:

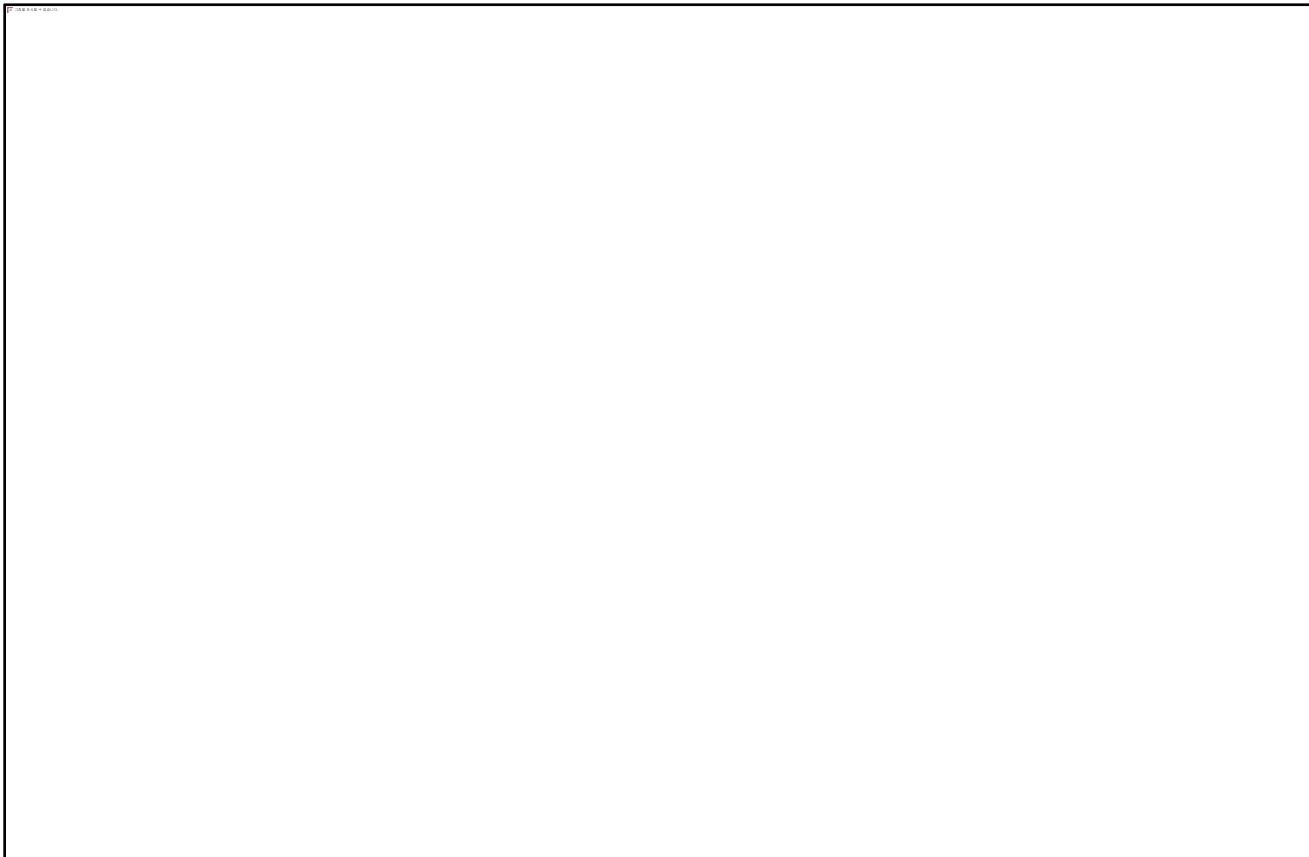
The following setting guide is for Odin 2017a.

After Odin 2017b (SP1), it is set as a separate dedicated UI. (8.13, 8.14)

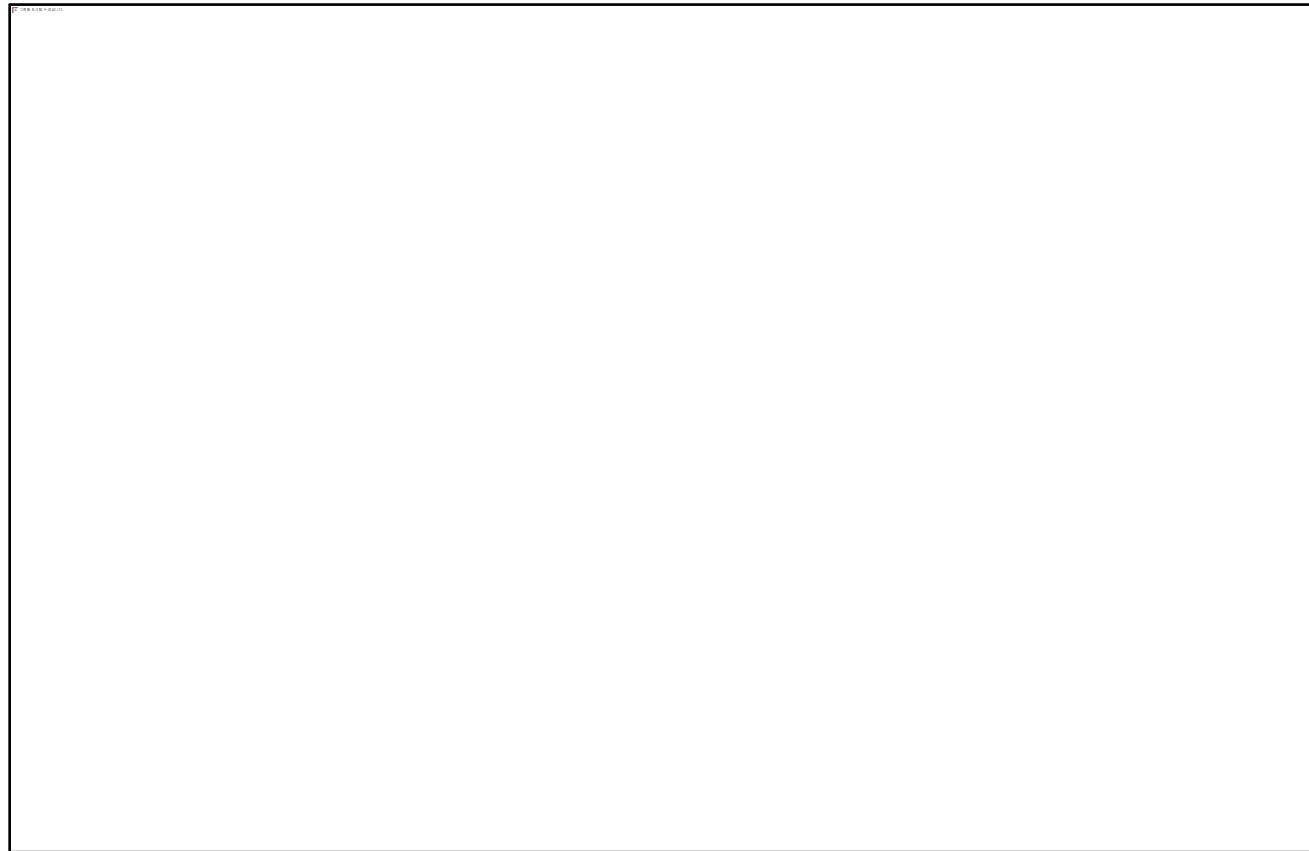
8.15.1 Configuration

Below is an example of Tx Signal, and Rx Signal setting is the same.

8.15.1.1 UINT8_N, UINT8_DYN type signal setting check



DOCUMENT NUMBER (DOC NO)	SHT/SHTS 158 / 181
-----------------------------	-----------------------



8.15.1.2 Port connection and event handling

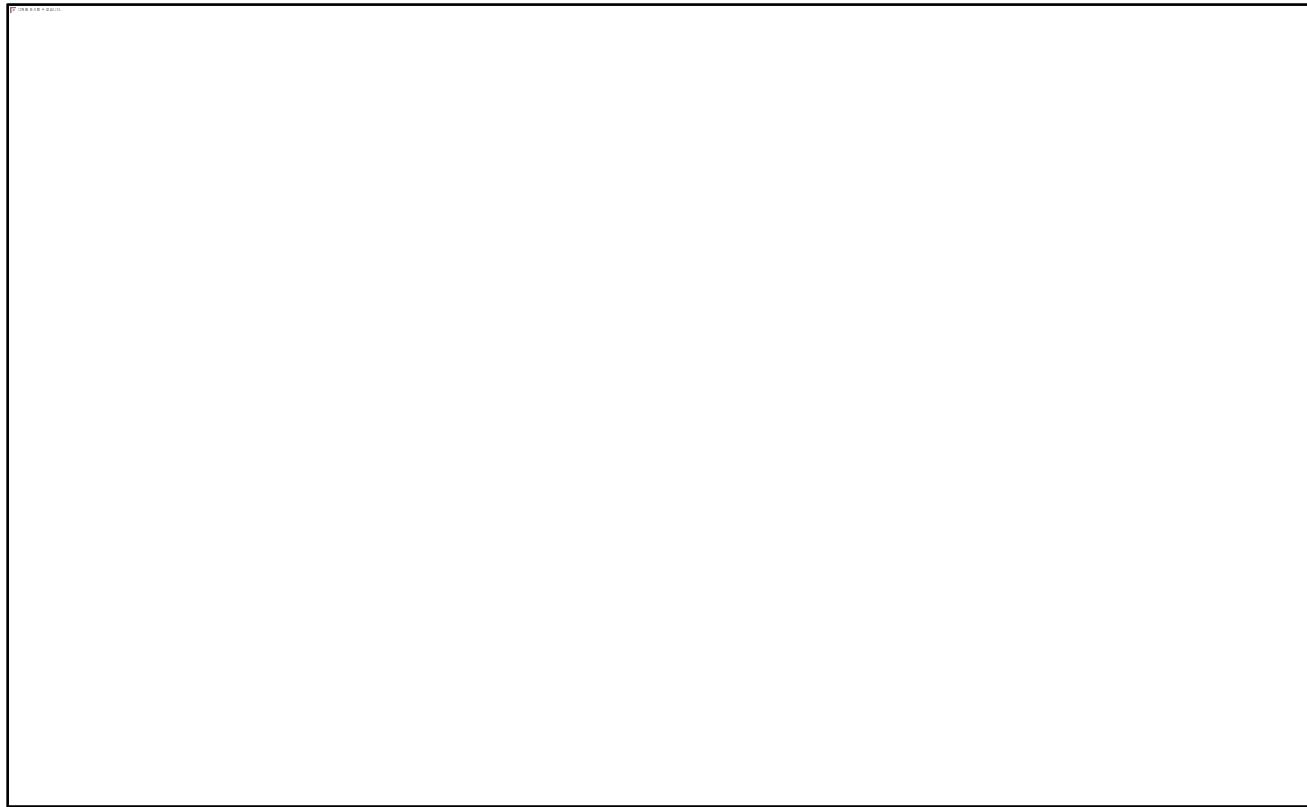
The general signal type, port connection, and event connection can be performed in the same way.

Task connection of Runnable is also the same. (Refer to 8.11, 0)

When creating a port, check whether it is created as a queued type.

DOCUMENT NUMBER (DOC NO)	SHT/SHTS 159 / 181
-----------------------------	-----------------------



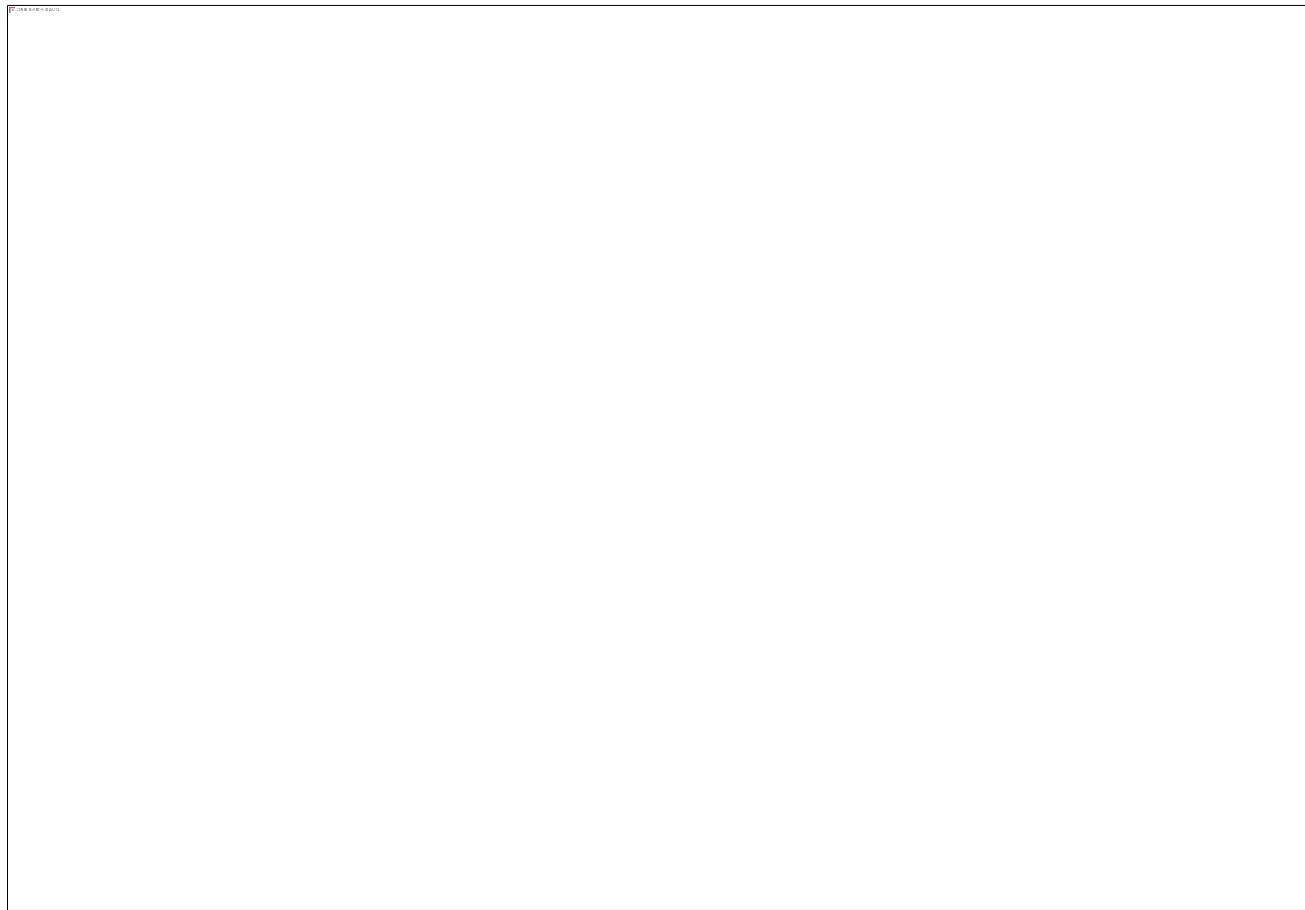


8.15.1.3 UINT8_DYN type Port Interface setting and system signal check

In the case of UINT8_DYN type, there are additional parts to check.

8.15.1.3.1 Port setting check

Click the Provided Interface link in the Port tab below.

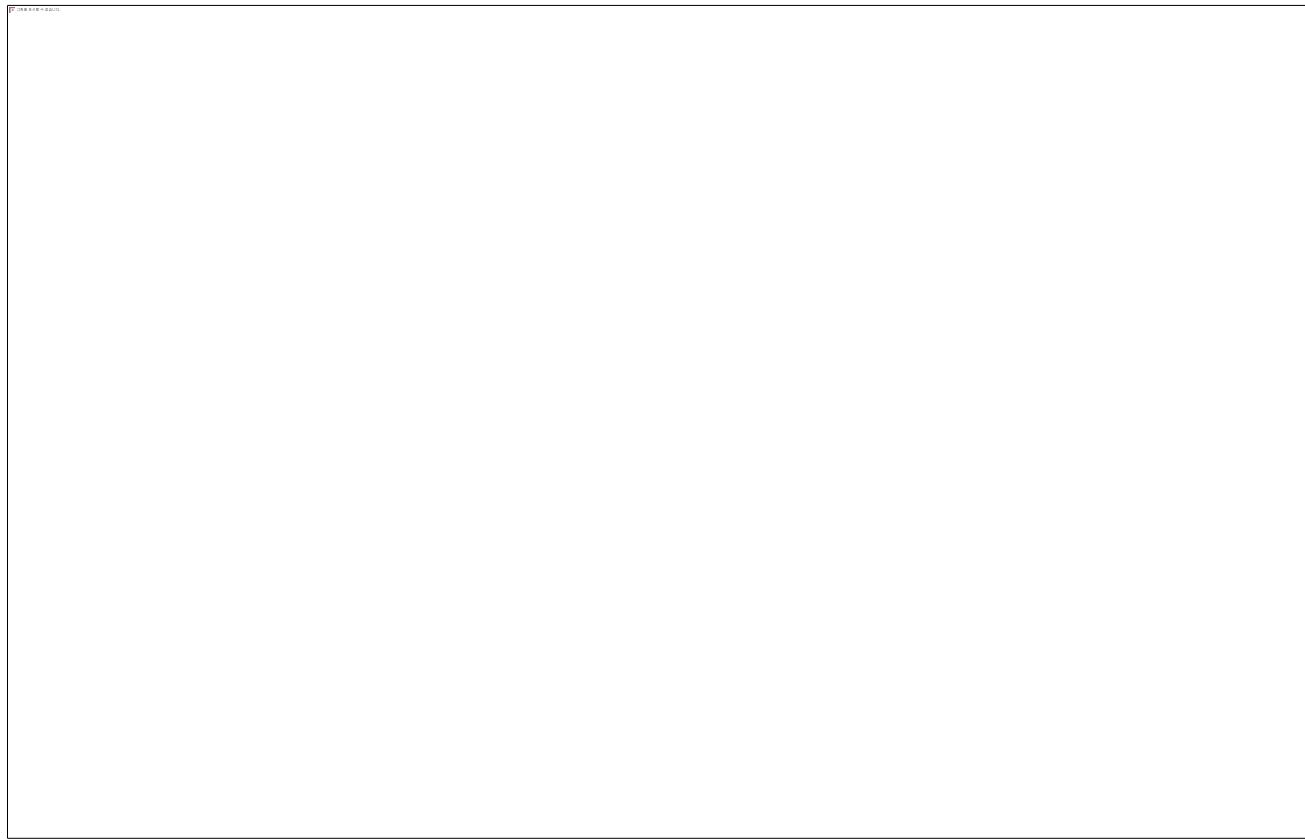


Select the relevant signal in the screen below and check the type by clicking the Edit Type button on the right.

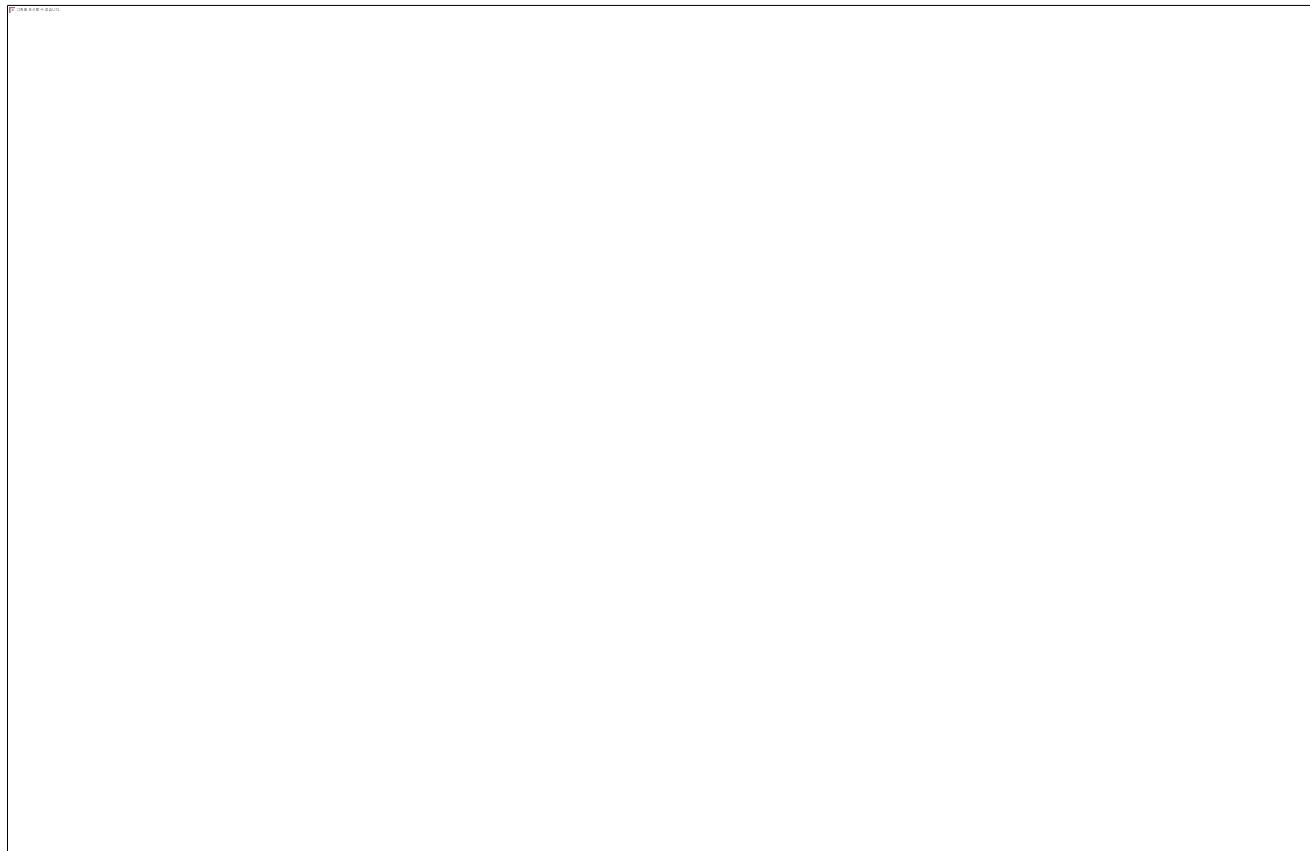
DOCUMENT NUMBER (DOC NO)	SHT/SHTS 162 / 181
-----------------------------	-----------------------

Check if the Category below the type setting is ARRAY. This part is the same for UINT8_N.

DOCUMENT NUMBER (DOC NO)	SHT/SHTS 163 / 181
-----------------------------	-----------------------

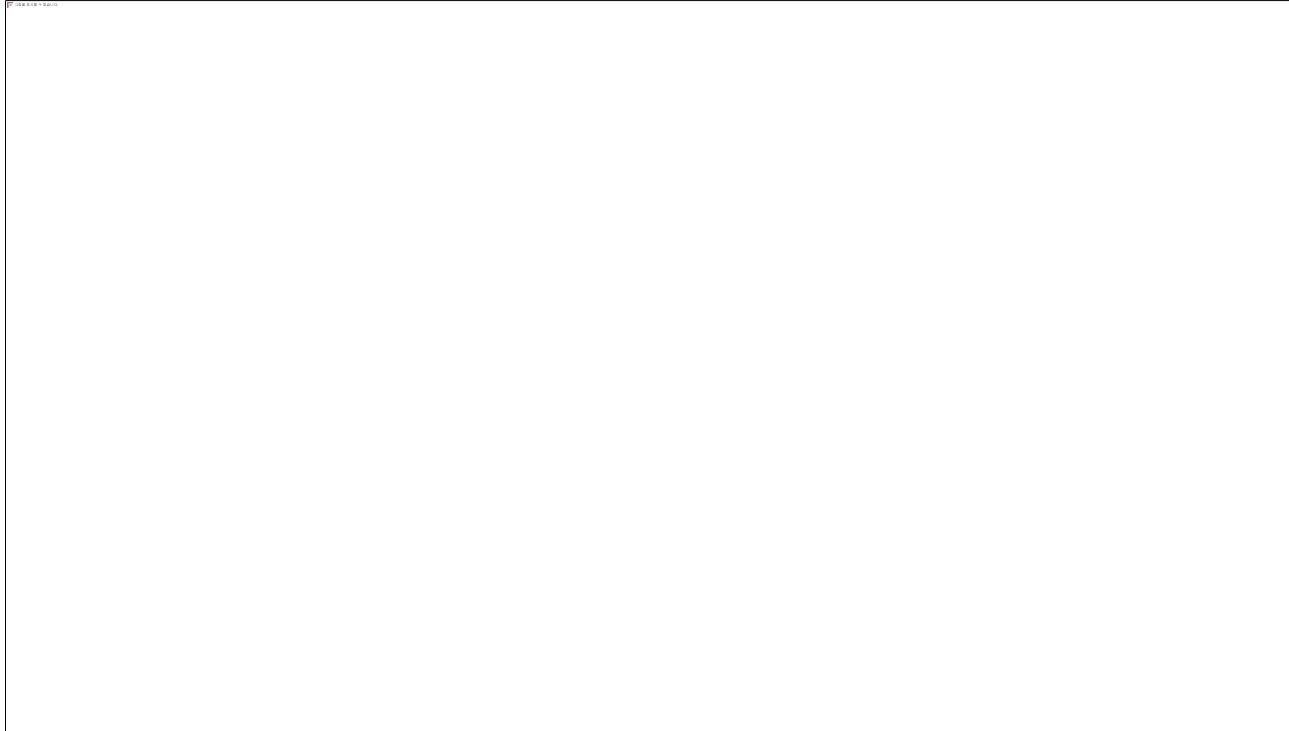


Go to Sub Elements and check the Array Size Semantics and Array Size values. In the case of `UINT8_N`, it is FIXED-SIZE, not VARIABLE-SIZE.

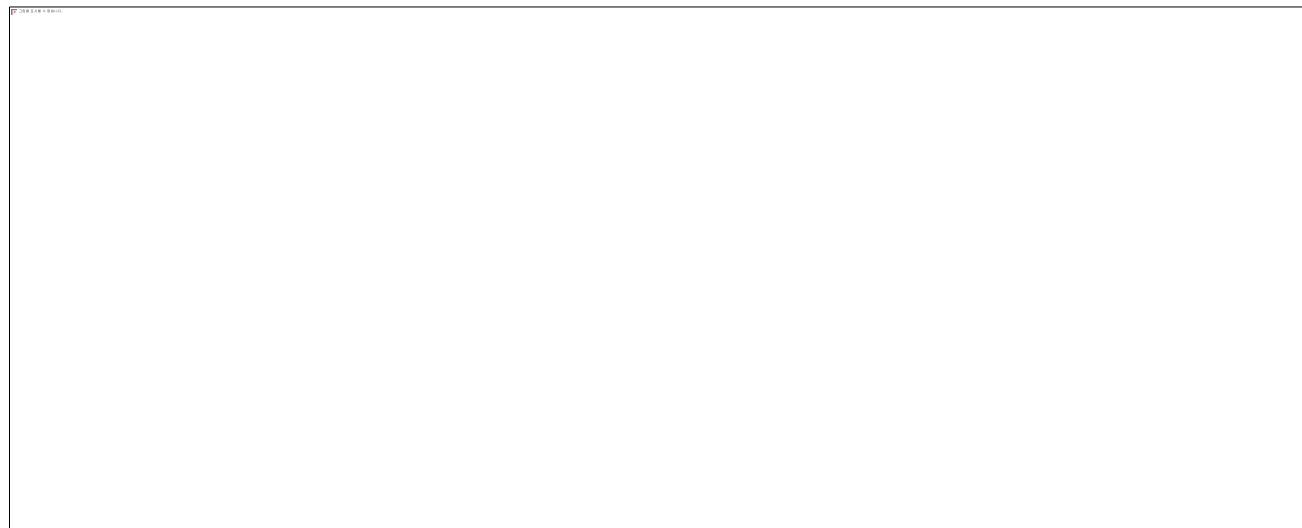
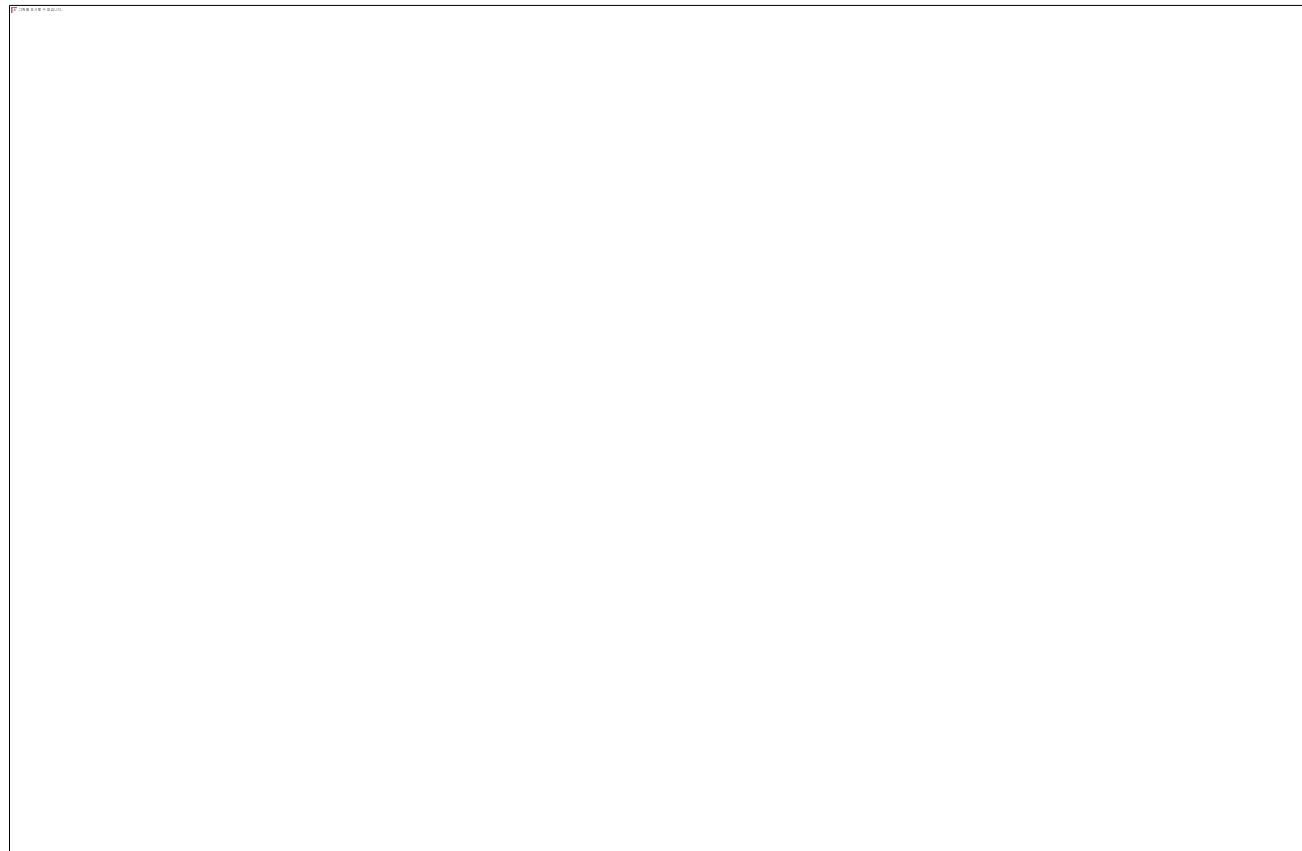


If the type does not exist, create a new one. You can create by copying an existing type, or right-click to create an Implementation Data Type as shown below. The Project.arxml file in the screen below is a file that is obtained by importing the DB file. You just need to create it in the arxml file where the

corresponding Signal of the project is located.

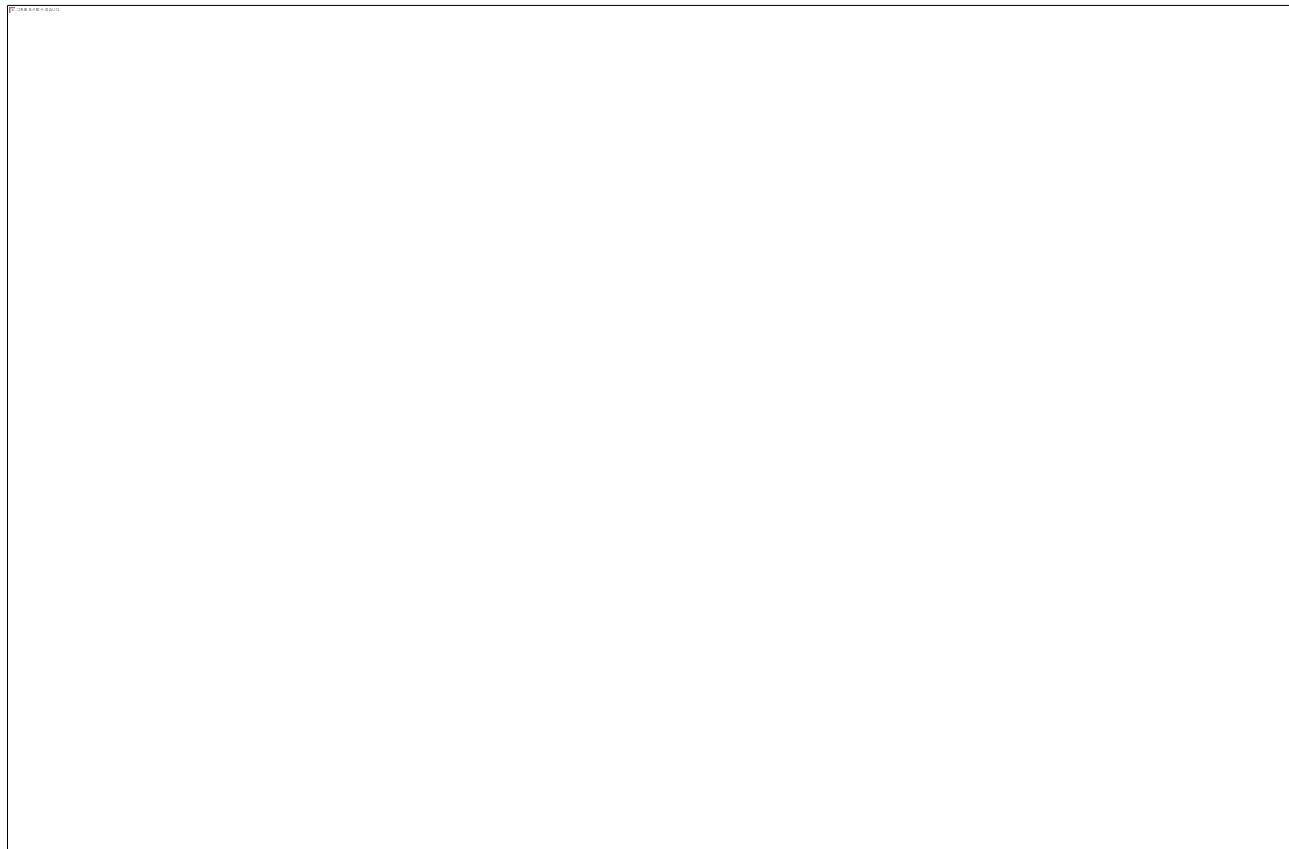


If necessary, set the type by pressing the button below on the Port Interface screen.

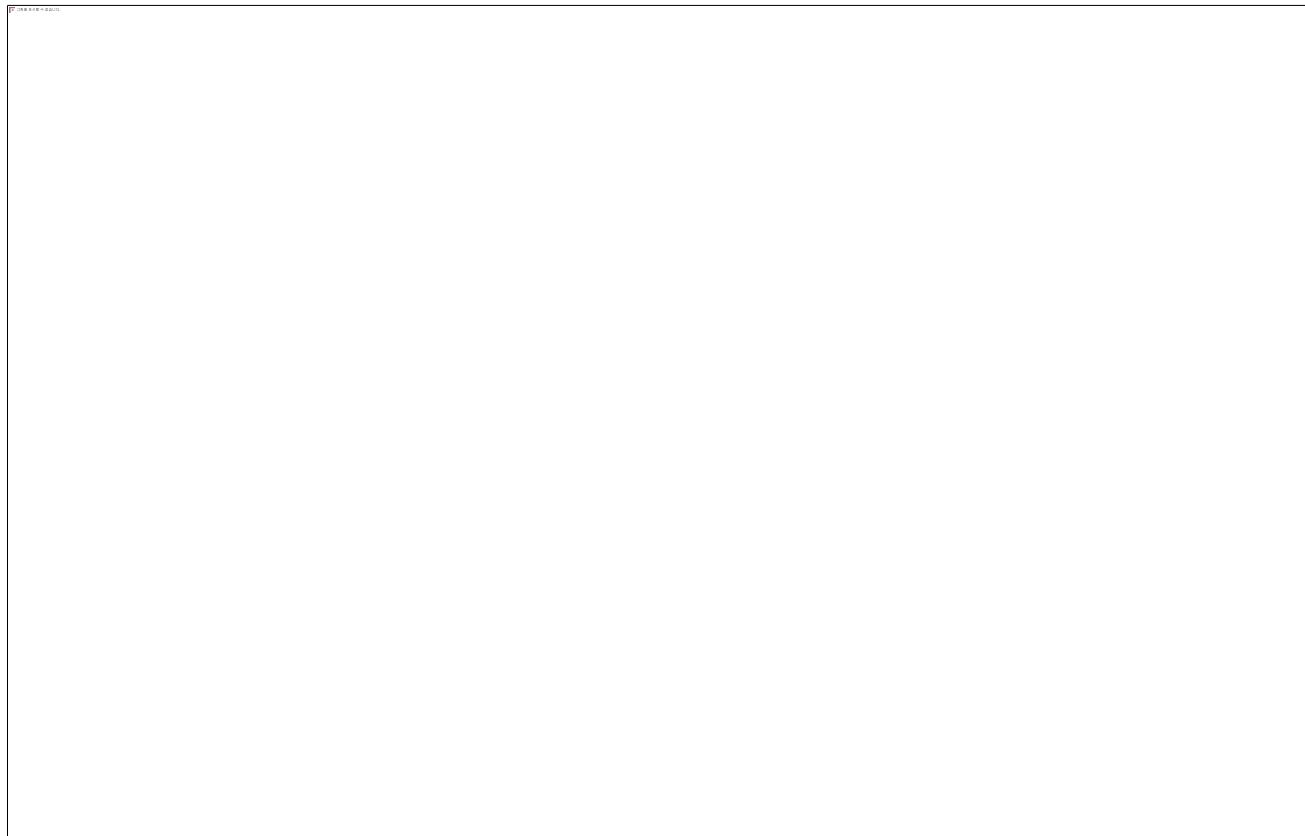


8.15.1.3.2 System Signal Signal Settings

Click the link in the box below on the Com module signal setting screen.

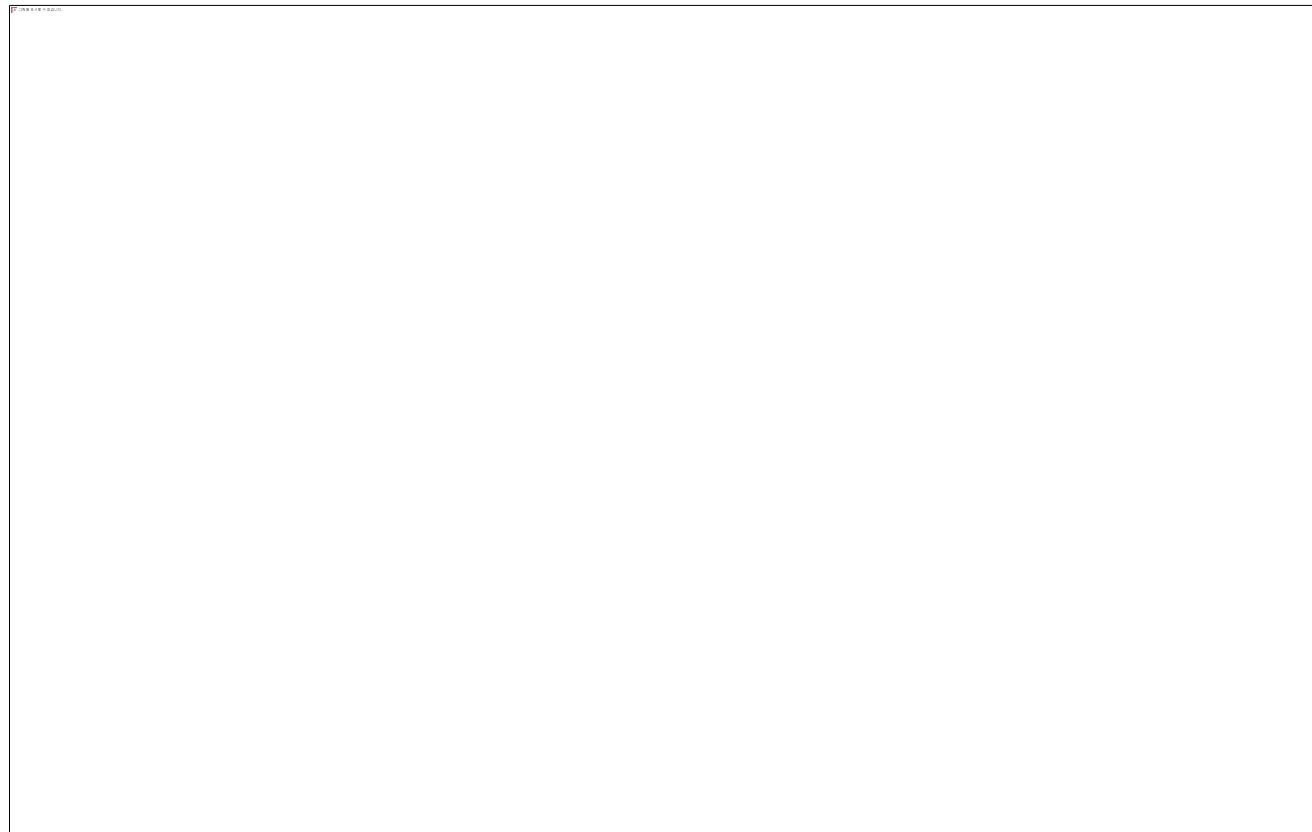


Click the ISignal link in the screen below.



Click the System Signal link in the screen below.

Check if Dynamic Length is set to TRUE.



8.16 Guide to check when Com communication is possible

8.16.1 Background

Requesting full communication does not mean that the COM module can process messages. After requesting Full Communication, ComM Module requests Full Communication to related modules (CanSM, LinSM), and after normal processing is completed, BswM starts Com Module I-PDU Group. Therefore, after confirming that the Com Module I-PDU Group is started in the application, the message should be transmitted. The following is a configuration that can check when a specific channel (or a specific group) of the Com Module is started.

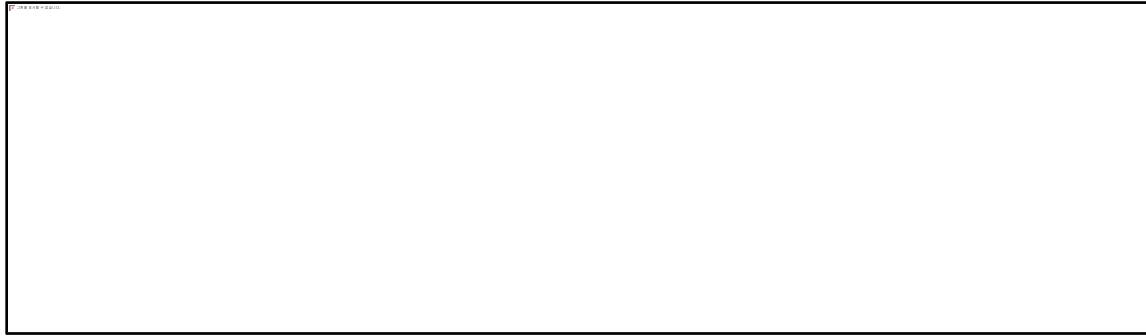
8.16.2 Setup Guide

Through Pdu Group Switch Notification, you can check the communication available time after Full Com request.

1. Modify App Mode SWC

- Check if the Service Needs for the Pdu Group to be used are registered. If not, perform BswM Harmonize after adding. If there is no Mode Switch Interface port for Pdu Group when New button is pressed, refer to 8.16.3.

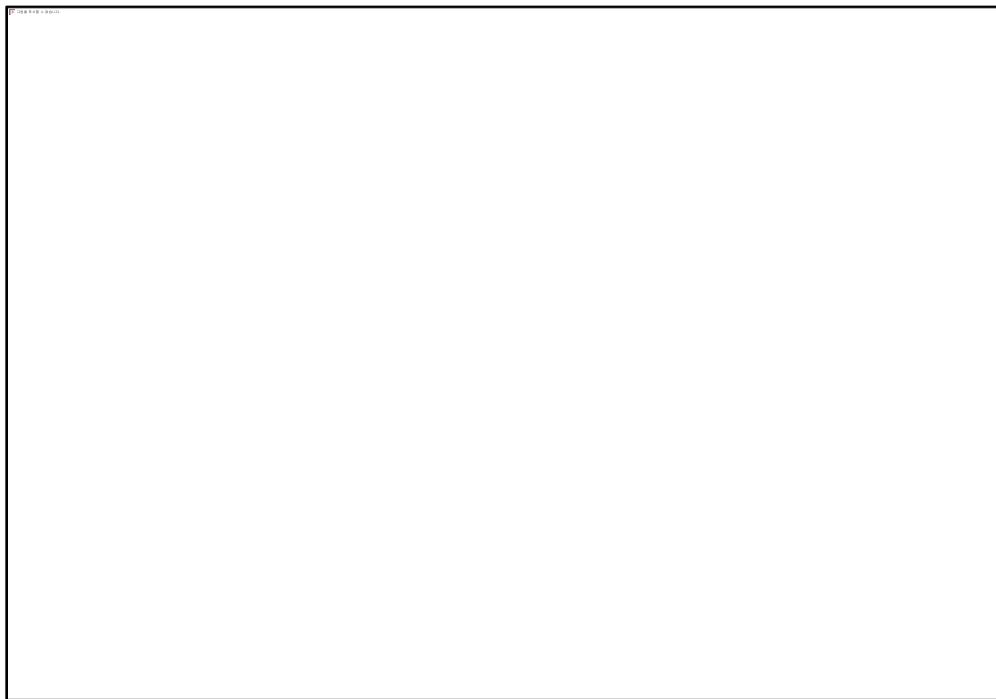
- Set Runnable and Event to execute Notification. Refer to the sample created in the RTU and set the Pdu group to receive the notification.



2. BswM – Add Connection between App Mode SWCs

Connection setting for Pdu Group added in Ecuc Value Collection Service and I/O

- Select the connection to add and click the + button



- Connection

- Code fix

Add function as below in App_Mode.c

XXX: Depends on the Pdu Group name used. Rte generation function reference

```
FUNC(void,AppMode_CODE) AppMode_PduGroupSwitched_XXX(void)
{
    Rte_ModeType_MDG_PduGroup LddPrevPduState, LddNextPduState;
    Rte_Mode_modeNotificationPort_PduGroupXX_XX_PduGroupXX_XX(&LddPrevPduState,
&LddNextPduState);

    /* Conditions when PduGroup started */
    if (LddNextPduState == RTE_MODE_MDG_PduGroup_STOP)
    {
        /* Write what to do when the Pdu Group is stopped */
    }

    /* Conditions when PduGroup completed */
    else if (LddNextPduState == RTE_MODE_MDG_PduGroup_START)
    {
        /* Write what to do when starting the Pdu Group */
    }
}
```

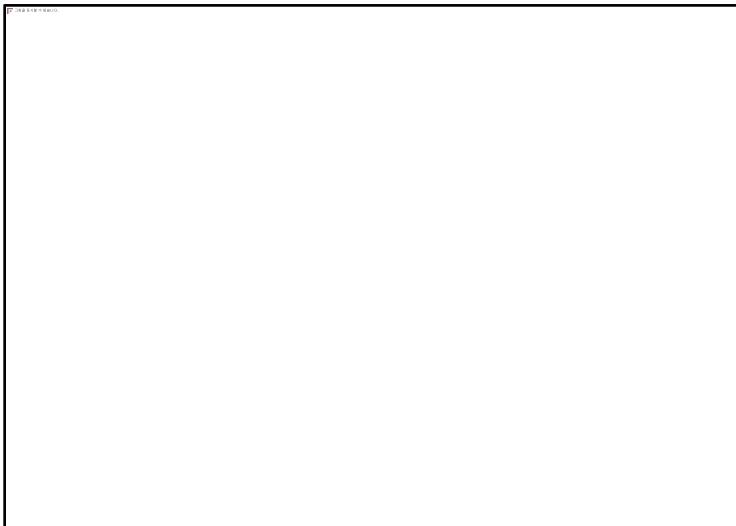
```
}
```

```
}
```

8.16.3 Mode Switch Interface Port Creation (Option)

When setting in 8.16.2, you only need to perform the following if the Mode Switch Interface port does not exist.

Select Mode Switch Interface → Mode Receiver.



In the screen below, select the ModeSwitchInterface of the desired Pdu Group.



Modify ShortName below if necessary. Check Enable Required Com Specs, and set Enhanced Mode API and Supports Asynchronous Mode Switch to true.

(The screen below may differ depending on the Odin version)



8.17 Group Signal IdsM Signal Value Check Setting Guide

8.17.1 Background

From Odin 2018 Patch 3, IdsM related settings Harmonize for Group Signal is supported. Therefore, in the case of the previous version, refer to this guide and proceed with manual setting. This guide assumes that the IdsM module is integrated.

8.17.2 Setup Guide

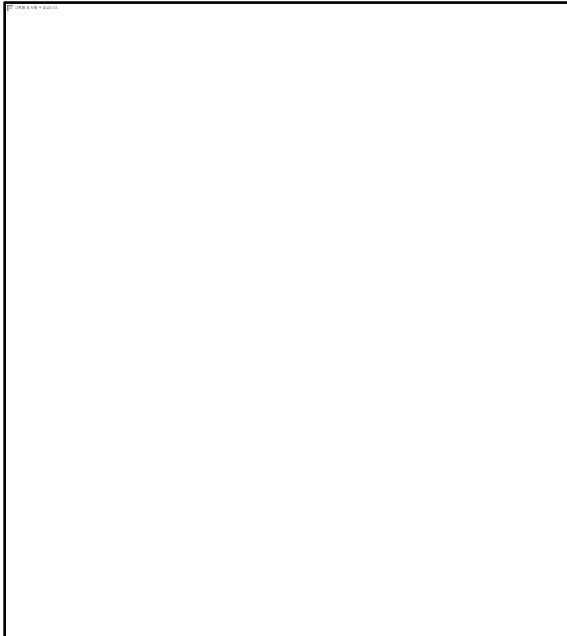
8.17.2.1 IdsM Settings

Refer to 5.1 IdsMGeneral setting and 5.3 IdsMSignalValueCheckFunc setting of AUTRON_AUTOSAR_Cdd_Idsm_UM.pdf document to perform Signal Value Check related settings.

8.17.2.2 Com Settings

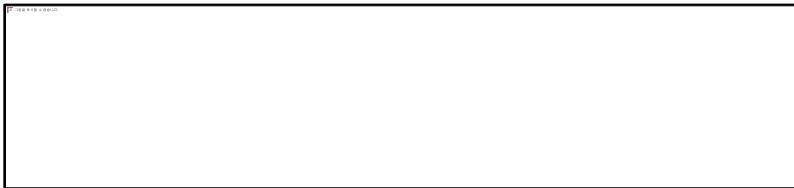
8.17.2.2.1 ComConfig-ComSignalGroup-ComGroupSignal setting

Set the SigReportIdsMEnable item of the Signal Value Check target group signal to ‘true’.

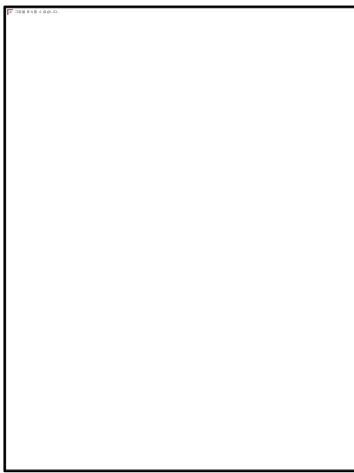


8.17.2.2.2 ComConfig-ComSignalGroup-ComGroupSignal-ComFilter settings

Right-click the Group Signal with the SigReportIdsMEnable item set to ‘true’ and add the ComFilter setting.



Refer to 5.11 and proceed with filter related settings. For example, when the value of the group signal is valid only when the value of the group signal is 0 or more and 10 or less, the filter setting is as follows.



8.18 Setting Guide for adding application condition to Pdu Group Switch condition

The Pdu Group Switch function is a function that enables or disables the status of a specific Pdu Group to control the communication available status of Com I-PDUs belonging to the corresponding group.

Currently, in Autron, rules to perform Pdu Group Switch for TX and RX Pdu Group for each communication channel are defined, and Pdu Group Switch is basically performed according to CanCM, Dcm, and ComM channel related conditions.

This guide is a setting guide for the case of performing Pdu Group Switch by additionally reflecting the mode conditions defined in the application to the basic rule conditions above. It is not allowed to make an application separate Rule without following the guide.

When executing this guide, the defined application mode transmits the BswM mode as a parameter of Rte_Write_[mode request application SW-C name]_[corresponding mode interface name]_[data element name of the corresponding mode interface], so Pdu Group Switch execution conditions It can be used for judgment.

8.18.1 Mode creation and interface definition

In the application, define the mode that the user will request for Pdu Group Switch, and create the relevant interface to create the port to request the mode. Create Mode and Interface by referring to step ‘Data definition, Interface definition’ in 8.1.4.1.1 SWC setting guide of AUTOSAR_AUTRON_BswM_UM.pdf.

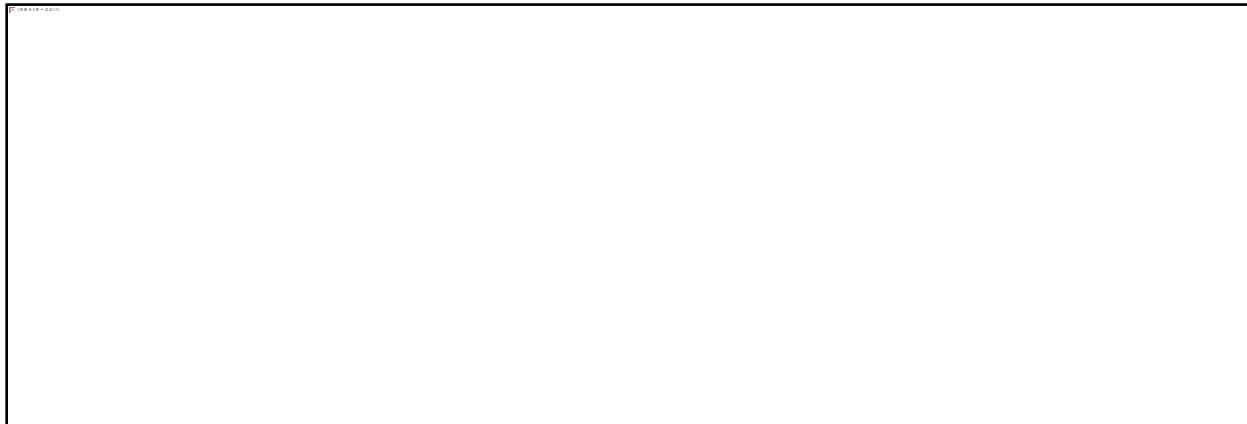
8.18.2 Mode Request related Application SW-C setting

You need to create a P-Port to request the mode defined above in Application SW-C and add Runnable, Data Send Point and Data Type Mapping. For related details, refer to the ‘Mode Requestor Definition’ step of 8.1.4.1.1 SWC Setting Guide of AUTOSAR_AUTRON_BswM_UM.pdf.

8.18.3 Service Needs Setting

In order to generate R-Port and Data Received Event for the P-Port created above in Swcd_Bsw_BswM.arxml, Service Needs setting is required.

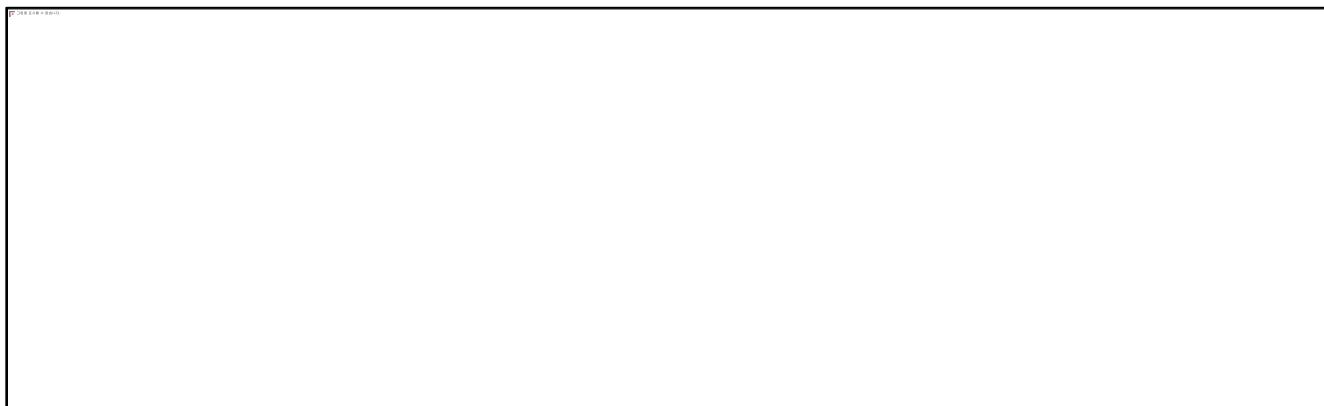
In the Service Needs tab of Application SW-C, double-click the following Service Needs > Bsw Mgr Needs and click the New button to add the created P-Port to the Assigned Port.



Next, click Generate ECU Configuration in the EcuCValueCollection.arxml file to harmonize the BswM module.



After selecting the BswM module and selecting ‘Auto Configuration for only used Mode’ to perform harmonization, the ‘Mode Manager setting’ step of 8.1.4.1.1 SWC setting guide of AUTOSAR_AUTRON_BswM_UM.pdf in Swcd_Bsw_BswM.arxml is automatically performed.



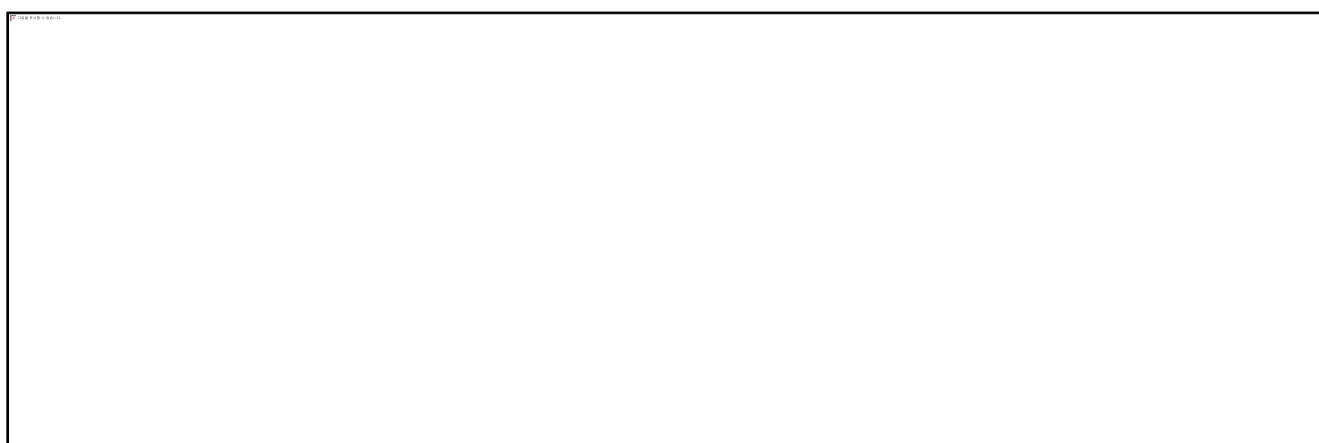
8.18.4 P-Port, R-Port Connection

In the Service and I/O tab of EcuCValueCollection.arxml, click BswM Component, click the '+' button marked with a red box, and R-Port related to the Application Definition Mode created in the BswM Component and P- related to the corresponding Mode of Application SW-C Connect the port.



8.18.5 Create ModeRequestPort, ModeCondition

ModeRequestPort and ModeCondition for the mode defined in Application should be added to BswM. To add automatically, click Generate ECU Configuration in EcuCValueCollection.arxml, select the BswM module, and select ‘Auto configuration for all possible Mode’ to perform BswM harmonization to create ModeRequestPort and ModeCondition for application defined mode. BswM settings that are not actually used are additionally created, but are removed by adding BswM harmonization in the last step of this guide.



8.18.6 Creating LEs

By combining the ModeCondition created in the above step, create LEs for Pdu Group Switch related application defined conditions under ‘BswMConfig > BswMArbitration > Logical Expression’.

8.18.7 Add LE to RuleLE

Double-click Pdu Group Switch related RuleLE under ‘BswM > BswMArbitration > Logical Expression’, and click ‘Browse’ of Argument Ref* parameter to add the above User Condition LE.

8.18.8 BswM Harmonize

To harmonize the BswM module, click Generate ECU Configuration in the EcuCValueCollection.arxml file.



After selecting the BswM module, select ‘Auto Configuration for only used Mode’ to perform harmonization to remove unused BswM settings.



8.19 Guide PDU data initialization function guide according to I-PDU Group Control

8.19.1 Function overview

In case of I-PDU Group Start, it is possible to select whether to initialize the PDU data (I-PDU Buffer Shadow Buffer, Filter Old Value) belonging to the relevant I-PDU Group to the Init Value through BswM setting.

For the related BswM setting method, refer to the following chapter.

8.19.2 Configuration

- Set position

: BswMConfig > BswMModeControl > Actions > PduGroupSwitchEnable Action > BswMAvailableAction > PduGroupSwitch settings

- Set target

: ‘Reinit’ parameter

→ true: Initialize to PDU data Init Value at I-PDU Group Start

→ false: Maintains PDU data value at I-PDU Group Start, does not perform initialization

