



Elektrobit

EB tresos[®] AutoCore Generic 8 COM Services documentation

product release 8.5.1



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

Technical support

Europe

Phone: +49 9131 7701 6060

Japan

Phone: +81 3 5577 6110

USA

Phone: +1 888 346 3813

Support URL

<https://www.elektrobit.com/support>

Legal notice

Confidential and proprietary information

ALL RIGHTS RESERVED. No part of this publication may be copied in any form, by photocopy, microfilm, retrieval system, or by any other means now known or hereafter invented without the prior written permission of Elektrobit Automotive GmbH.

ProOSEK®, tresos®, and street director® are registered trademarks of Elektrobit Automotive GmbH.

All brand names, trademarks and registered trademarks are property of their rightful owners and are used only for description.

Copyright 2018, Elektrobit Automotive GmbH.

Table of Contents

1. Overview of EB tresos AutoCore Generic 8 COM Services documentation	12
2. ACG8 COM Services release notes	13
2.1. Overview	13
2.2. Scope of the release	13
2.2.1. Configuration tool	13
2.2.2. AUTOSAR modules	13
2.2.3. EB (Elektrobit) modules	13
2.2.4. MCAL modules and EB tresos AutoCore OS	14
2.3. Module release notes	14
2.3.1. Com module release notes	14
2.3.1.1. Change log	14
2.3.1.2. New features	24
2.3.1.3. EB-specific enhancements	25
2.3.1.4. Deviations	27
2.3.1.5. Limitations	37
2.3.2. IpduM module release notes	40
2.3.2.1. Change log	40
2.3.2.2. New features	47
2.3.2.3. EB-specific enhancements	48
2.3.2.4. Deviations	50
2.3.2.5. Limitations	54
2.3.3. PduR module release notes	55
2.3.3.1. Change log	55
2.3.3.2. New features	63
2.3.3.3. EB-specific enhancements	63
2.3.3.4. Deviations	64
2.3.3.5. Limitations	73
3. ACG8 COM Services user's guide	75
3.1. Overview	75
3.2. Background information	75
3.2.1. Network-independent and network-dependent communication in AUTOSAR	75
3.2.1.1. Modules and dependencies of the network-independent communication stack... ..	75
3.2.1.2. Data transmission in the network-independent communication stack	76
3.2.1.2.1. Signals and signal groups	77
3.2.1.2.2. Transmission modes and transfer properties	77
3.2.1.2.3. I-PDU groups	78
4. ACG8 COM Services module references	79
4.1. Overview	79
4.1.1. Notation in EB module references	79

4.1.1.1. Default value of configuration parameters	79
4.1.1.2. Range information of configuration parameters	79
4.2. Com	80
4.2.1. Configuration parameters	80
4.2.1.1. ComDefensiveProgramming	81
4.2.1.2. ComConfig	84
4.2.1.3. ComGwMapping	85
4.2.1.4. ComGwDestination	85
4.2.1.5. ComGwDestinationDescription	85
4.2.1.6. ComFilter	88
4.2.1.7. ComGwSignal	91
4.2.1.8. ComGwSource	91
4.2.1.9. ComGwSignal	91
4.2.1.10. ComGwSourceDescription	92
4.2.1.11. ComIPdu	95
4.2.1.12. ComIPduCounter	99
4.2.1.13. ComIPduReplication	100
4.2.1.14. ComTxIPdu	101
4.2.1.15. ComTxModeFalse	103
4.2.1.16. ComTxMode	103
4.2.1.17. ComTxModeTrue	105
4.2.1.18. ComTxMode	105
4.2.1.19. ComIPduGroup	108
4.2.1.20. ComSignal	108
4.2.1.21. ComFilter	117
4.2.1.22. ComSignalGroup	119
4.2.1.23. ComGroupSignal	124
4.2.1.24. ComFilter	129
4.2.1.25. ComTimeBase	131
4.2.1.26. ComTxMainFunctions	133
4.2.1.27. Com_MainFunctionTx	133
4.2.1.28. ComRxMainFunctions	134
4.2.1.29. Com_MainFunctionRx	134
4.2.1.30. ComGeneral	135
4.2.1.31. VendorSpecific	137
4.2.1.32. ComGeneratedRxSignal	171
4.2.1.33. CommonPublishedInformation	173
4.2.1.34. PublishedInformation	176
4.2.2. Recommended configurations	176
4.2.2.1. ComRecConfigurationStandard	176
4.2.2.1.1. ComGeneral	177
4.2.2.1.2. VendorSpecific	177

4.2.2.2. ComRecConfigurationSmall	178
4.2.2.2.1. ComGeneral	179
4.2.2.2.2. VendorSpecific	179
4.2.2.3. ComRecConfigurationMedium	180
4.2.2.3.1. ComGeneral	181
4.2.2.3.2. VendorSpecific	181
4.2.2.4. ComRecConfigurationMax	182
4.2.2.4.1. ComGeneral	183
4.2.2.4.2. VendorSpecific	183
4.2.3. Application programming interface (API)	184
4.2.3.1. Type definitions	185
4.2.3.1.1. Com_IpduGroupIdType	185
4.2.3.1.2. Com_IpduGroupVector	185
4.2.3.1.3. Com_PduGroupIdType	185
4.2.3.1.4. Com_RxCalloutType	185
4.2.3.1.5. Com_ServiceIdType	185
4.2.3.1.6. Com_SignalGroupIdType	185
4.2.3.1.7. Com_SignalIdType	186
4.2.3.1.8. Com_StatusType	186
4.2.3.1.9. Com_TxCalloutType	186
4.2.3.2. Macro constants	186
4.2.3.2.1. COMServiceId_ClearIpduGroupVector	186
4.2.3.2.2. COMServiceId_CopyRxData	186
4.2.3.2.3. COMServiceId_CopyTxData	187
4.2.3.2.4. COMServiceId_DeInit	187
4.2.3.2.5. COMServiceId_GetConfigurationId	187
4.2.3.2.6. COMServiceId_GetRxIpduBuffer	187
4.2.3.2.7. COMServiceId_GetStatus	187
4.2.3.2.8. COMServiceId_GetVersionInfo	188
4.2.3.2.9. COMServiceId_Init	188
4.2.3.2.10. COMServiceId_InternalAPI	188
4.2.3.2.11. COMServiceId_InvalidateShadowSignal	188
4.2.3.2.12. COMServiceId_InvalidateSignal	188
4.2.3.2.13. COMServiceId_InvalidateSignalGroup	189
4.2.3.2.14. COMServiceId_IpduGroupControl	189
4.2.3.2.15. COMServiceId_MainFunctionRouteSignals	189
4.2.3.2.16. COMServiceId_MainFunctionRx	189
4.2.3.2.17. COMServiceId_MainFunctionTx	189
4.2.3.2.18. COMServiceId_ReceiveDynSignal	190
4.2.3.2.19. COMServiceId_ReceiveShadowSignal	190
4.2.3.2.20. COMServiceId_ReceiveSignal	190
4.2.3.2.21. COMServiceId_ReceiveSignalGroup	190

4.2.3.2.22. COMServiceld_ReceiveSignalGroupArray	190
4.2.3.2.23. COMServiceld_ReceptionDMControl	191
4.2.3.2.24. COMServiceld_RxIndication	191
4.2.3.2.25. COMServiceld_SendDynSignal	191
4.2.3.2.26. COMServiceld_SendSignal	191
4.2.3.2.27. COMServiceld_SendSignalGroup	191
4.2.3.2.28. COMServiceld_SendSignalGroupArray	192
4.2.3.2.29. COMServiceld_SetIpduGroup	192
4.2.3.2.30. COMServiceld_StartOfReception	192
4.2.3.2.31. COMServiceld_SwitchIpduTxMode	192
4.2.3.2.32. COMServiceld_TpRxIndication	192
4.2.3.2.33. COMServiceld_TpTxConfirmation	193
4.2.3.2.34. COMServiceld_TriggerIPDUSend	193
4.2.3.2.35. COMServiceld_TriggerTransmit	193
4.2.3.2.36. COMServiceld_TxConfirmation	193
4.2.3.2.37. COMServiceld_UpdateShadowSignal	193
4.2.3.2.38. COM_E_PARAM	194
4.2.3.2.39. COM_E_PARAM_POINTER	194
4.2.3.2.40. COM_E_SIGNAL_TOO_WIDE	194
4.2.3.2.41. COM_E_UNINIT	194
4.2.3.2.42. COM_INSTANCE_ID	194
4.2.3.3. Functions	195
4.2.3.3.1. Com_ClearIpduGroupVector	195
4.2.3.3.2. Com_CopyRxData	195
4.2.3.3.3. Com_CopyTxData	196
4.2.3.3.4. Com_DeInit	197
4.2.3.3.5. Com_GetConfigurationId	197
4.2.3.3.6. Com_GetRxIPduBuffer	197
4.2.3.3.7. Com_GetStatus	198
4.2.3.3.8. Com_GetVersionInfo	198
4.2.3.3.9. Com_Init	199
4.2.3.3.10. Com_IpduGroupControl	199
4.2.3.3.11. Com_IsValidConfig	200
4.2.3.3.12. Com_MainFunctionRouteSignals	200
4.2.3.3.13. Com_MainFunctionRx	201
4.2.3.3.14. Com_MainFunctionTx	201
4.2.3.3.15. Com_ReceiveDynSignal	201
4.2.3.3.16. Com_ReceiveShadowSignal	202
4.2.3.3.17. Com_ReceiveSignal	203
4.2.3.3.18. Com_ReceiveSignalGeneric	203
4.2.3.3.19. Com_ReceiveSignalGroup	204
4.2.3.3.20. Com_ReceiveSignalGroupArray	204

4.2.3.3.21. Com_ReceptionDMControl	205
4.2.3.3.22. Com_RxIndication	205
4.2.3.3.23. Com_SendDynSignal	206
4.2.3.3.24. Com_SendSignal	207
4.2.3.3.25. Com_SendSignalGroup	207
4.2.3.3.26. Com_SendSignalGroupArray	208
4.2.3.3.27. Com_SetIpduGroup	209
4.2.3.3.28. Com_StartOfReception	209
4.2.3.3.29. Com_SwitchIpduTxMode	210
4.2.3.3.30. Com_TpRxIndication	210
4.2.3.3.31. Com_TpTxConfirmation	211
4.2.3.3.32. Com_TriggerIPDUSend	211
4.2.3.3.33. Com_TriggerTransmit	212
4.2.3.3.34. Com_TxConfirmation	212
4.2.3.3.35. Com_UpdateShadowSignal	213
4.2.4. Integration notes	213
4.2.4.1. Exclusive areas	213
4.2.4.1.1. COM_EXCLUSIVE_AREA_0	213
4.2.4.1.2. COM_EXCLUSIVE_AREA_1	214
4.2.4.2. Production errors	214
4.2.4.3. Memory mapping	214
4.2.4.4. Integration requirements	215
4.2.4.4.1. Com.EB.IntReq.Preemption01	215
4.2.4.4.2. Com.EB.IntReq.Preemption02	216
4.2.4.4.3. Com.EB.IntReq.Preemption02.TP	217
4.2.4.4.4. Com.EB.IntReq.Preemption03	219
4.2.4.4.5. Com.EB.IntReq.MainRxSchedule04	219
4.2.4.4.6. Com.EB.IntReq.UpdateBit05	220
4.2.4.4.7. Com.EB.IntReq.Preemption06	220
4.2.4.4.8. Com.EB.IntReq.Preemption07	221
4.3. IpduM	221
4.3.1. Configuration parameters	222
4.3.1.1. CommonPublishedInformation	223
4.3.1.2. IpduMDefensiveProgramming	226
4.3.1.3. IpduMConfig	229
4.3.1.4. IpduMContainedRxPdu	229
4.3.1.5. IpduMContainedTxPdu	230
4.3.1.6. IpduMContainerRxPdu	233
4.3.1.7. IpduMContainerTxPdu	235
4.3.1.8. IpduMRxPathway	238
4.3.1.9. IpduMRxIndication	238
4.3.1.10. IpduMRxDynamicPart	240

4.3.1.11. IpduMSegment	241
4.3.1.12. IpduMRxStaticPart	242
4.3.1.13. IpduMSegment	242
4.3.1.14. IpduMSelectorFieldPosition	244
4.3.1.15. IpduMTxPathway	244
4.3.1.16. IpduMTxRequest	245
4.3.1.17. IpduMSelectorFieldPosition	248
4.3.1.18. IpduMTxDynamicPart	249
4.3.1.19. IpduMSegment	251
4.3.1.20. IpduMTxStaticPart	252
4.3.1.21. IpduMSegment	254
4.3.1.22. IpduMGeneral	255
4.3.1.23. IpduMRxProcessing	266
4.3.1.24. IpduMTxProcessing	267
4.3.1.25. IpduMPublishedInformation	268
4.3.1.26. IpduMRequestMessageConfiguration	268
4.3.1.27. IpduMRequestMessageMapInfo	269
4.3.1.28. PublishedInformation	270
4.3.2. Application programming interface (API)	271
4.3.2.1. Macro constants	271
4.3.2.1.1. IPDUM_E_GLOBAL_ECUID	271
4.3.2.1.2. IPDUM_SID_PROCESS_REQUEST_PDU	271
4.3.2.2. Functions	271
4.3.2.2.1. IpduM_GetVersionInfo	271
4.3.2.2.2. IpduM_Init	272
4.3.2.2.3. IpduM_MainFunctionRx	272
4.3.2.2.4. IpduM_MainFunctionTx	272
4.3.2.2.5. IpduM_ProcessRequestPdu	273
4.3.2.2.6. IpduM_RxIndication	273
4.3.2.2.7. IpduM_Transmit	274
4.3.2.2.8. IpduM_TriggerTransmit	274
4.3.2.2.9. IpduM_TxConfirmation	275
4.3.3. Integration notes	275
4.3.3.1. Exclusive areas	275
4.3.3.1.1. SCHM_IPDUM_EXCLUSIVE_AREA_0	276
4.3.3.2. Production errors	276
4.3.3.3. Memory mapping	276
4.3.3.4. Integration requirements	277
4.4. PduR	277
4.4.1. Configuration parameters	277
4.4.1.1. CommonPublishedInformation	278
4.4.1.2. PublishedInformation	281

4.4.1.3. PduRBSwModules	281
4.4.1.4. PduRGeneral	289
4.4.1.5. PduRRoutingTables	299
4.4.1.6. PduRRoutingPathGroup	300
4.4.1.7. PduRRoutingTable	301
4.4.1.8. PduRRoutingPath	302
4.4.1.9. PduRDestPdu	303
4.4.1.10. PduRDefaultValue	305
4.4.1.11. PduRDefaultValueElement	306
4.4.1.12. PduRSrcPdu	307
4.4.1.13. PduRTpBufferTable	307
4.4.1.14. PduRTpBuffer	308
4.4.1.15. PduRTxBufferTable	309
4.4.1.16. PduRTxBuffer	309
4.4.2. Recommended configurations	310
4.4.2.1. PduRRecConfigurationGatewayEcu	310
4.4.2.1.1. Com	311
4.4.2.1.2. Dcm	312
4.4.2.1.3. CanIf	312
4.4.2.1.4. LinIf	313
4.4.2.1.5. FrIf	313
4.4.2.1.6. CanTp	314
4.4.2.1.7. LinTp	315
4.4.2.1.8. FrTp	315
4.4.2.1.9. SoAd	316
4.4.2.1.10. DoIP	316
4.4.2.1.11. IpduM	317
4.4.2.1.12. PduRGeneral	318
4.4.2.2. PduRRecConfigurationFrEcu	318
4.4.2.2.1. Com	318
4.4.2.2.2. Dcm	319
4.4.2.2.3. FrIf	320
4.4.2.2.4. FrTp	320
4.4.2.2.5. IpduM	321
4.4.2.3. PduRRecConfigurationCanEcu	321
4.4.2.3.1. Com	322
4.4.2.3.2. Dcm	322
4.4.2.3.3. CanIf	323
4.4.2.3.4. CanTp	324
4.4.2.3.5. IpduM	324
4.4.2.4. PduRRecConfigurationLinEcu	325
4.4.2.4.1. Com	325

4.4.2.4.2. Dcm	326
4.4.2.4.3. LinIf	326
4.4.2.4.4. LinTp	327
4.4.2.4.5. IpduM	328
4.4.2.5. PduRRecConfigurationEthernetEcu	328
4.4.2.5.1. Com	329
4.4.2.5.2. Dcm	329
4.4.2.5.3. SoAd	330
4.4.2.5.4. DoIP	330
4.4.2.5.5. IpduM	331
4.4.3. Application programming interface (API)	332
4.4.3.1. Macro constants	332
4.4.3.1.1. PDUR_E_CONFIG_PTR_INVALID	332
4.4.3.1.2. PDUR_E_INVALID_REQUEST	332
4.4.3.1.3. PDUR_E_NULL_POINTER	332
4.4.3.1.4. PDUR_E_PDU_ID_INVALID	332
4.4.3.1.5. PDUR_E_PDU_INSTANCES_LOST	332
4.4.3.1.6. PDUR_E_TP_TX_REQ_REJECTED	333
4.4.3.1.7. PDUR_INSTANCE_ID	333
4.4.3.1.8. PDUR_INVALID_CONFIGURATION_ID	333
4.4.3.1.9. PDUR_SID_GET_CONF_ID	333
4.4.3.1.10. PDUR_SID_GET_VER_INF	333
4.4.3.1.11. PDUR_SID_IFGW_RXIND_DF	333
4.4.3.1.12. PDUR_SID_IFGW_RXIND_SB	334
4.4.3.1.13. PDUR_SID_IFGW_RXIND_TF	334
4.4.3.1.14. PDUR_SID_IFGW_TRIGTX_SB	334
4.4.3.1.15. PDUR_SID_IFGW_TRIGTX_TF	334
4.4.3.1.16. PDUR_SID_INIT	334
4.4.3.1.17. PDUR_SID_LOTP_COPY_RX_DATA	334
4.4.3.1.18. PDUR_SID_LOTP_COPY_TX_DATA	335
4.4.3.1.19. PDUR_SID_LOTP_RXIND	335
4.4.3.1.20. PDUR_SID_LOTP_STRT_OF_RCPTN	335
4.4.3.1.21. PDUR_SID_LOTP_TX_CONF	335
4.4.3.1.22. PDUR_SID_LO_RXIND	335
4.4.3.1.23. PDUR_SID_LO_TRIGTX	335
4.4.3.1.24. PDUR_SID_LO_TXCONF	336
4.4.3.1.25. PDUR_SID_UP_CANCEL_RXREQ	336
4.4.3.1.26. PDUR_SID_UP_CANCEL_TXREQ	336
4.4.3.1.27. PDUR_SID_UP_CHANGE_PARAREQ	336
4.4.3.1.28. PDUR_SID_UP_TX	336
4.4.3.1.29. PduR_GetVersionInfo	336
4.4.3.2. Objects	337

4.4.3.2.1. PduR_State	337
4.4.3.2.2. PduR_gConfigPtr	337
4.4.3.3. Functions	337
4.4.3.3.1. PduR_GetConfigurationId	337
4.4.3.3.2. PduR_Init	338
4.4.3.3.3. PduR_IsValidConfig	338
4.4.3.3.4. PduR_LoRxIndication	339
4.4.3.3.5. PduR_LoTpCopyRxData	339
4.4.3.3.6. PduR_LoTpCopyTxData	340
4.4.3.3.7. PduR_LoTpRxIndication	341
4.4.3.3.8. PduR_LoTpStartOfReception	342
4.4.3.3.9. PduR_LoTpTxConfirmation	343
4.4.3.3.10. PduR_LoTriggerTransmit	343
4.4.3.3.11. PduR_LoTxConfirmation	344
4.4.3.3.12. PduR_UpCancelReceive	344
4.4.3.3.13. PduR_UpCancelTransmit	344
4.4.3.3.14. PduR_UpChangeParameter	345
4.4.3.3.15. PduR_UpTransmit	345
4.4.4. Integration notes	346
4.4.4.1. Exclusive areas	346
4.4.4.1.1. SCHM_PDUR_EXCLUSIVE_AREA_0	346
4.4.4.2. Production errors	346
4.4.4.3. Memory mapping	346
4.4.4.4. Integration requirements	347
4.4.4.4.1. PduR.EB.IntReq.RestrictTpGwToSF	347
4.4.4.4.2. PduR.EB.IntReq.BlockLoTpCopyTxDataForTpMulticast	348
4.4.4.4.3. PduR.EB.IntReq.BlockLoTpCopyTxDataForDirectTpGw	348
4.4.4.4.4. PduR.EB.IntReq.RestrictNto1toSingleActivatedRPath	348
5. Bibliography	349



1. Overview of EB tresos AutoCore Generic 8 COM Services documentation

Welcome to the EB tresos AutoCore Generic 8 COM Services (ACG8 COM Services) product documentation.

This document provides:

- ▶ [Chapter 2, “ACG8 COM Services release notes”](#): release notes for the ACG8 COM Services modules
- ▶ [Chapter 3, “ACG8 COM Services user's guide”](#): containing background information and instructions
- ▶ [Chapter 4, “ACG8 COM Services module references”](#): information about configuration parameters and the application programming interface

2. ACG8 COM Services release notes

2.1. Overview

This chapter provides the ACG8 COM Services product specific release notes. General release notes that are applicable to all products are provided in the EB tresos AutoCore Generic documentation. Refer to the general release notes in addition to the product release notes documented here.

2.2. Scope of the release

2.2.1. Configuration tool

Your release of EB tresos AutoCore is compatible with the release of the EB tresos Studio configuration tool:

- ▶ EB tresos Studio: 25.0.0 b180628-0943

2.2.2. AUTOSAR modules

The following table lists the AUTOSAR modules that are part of this ACG8 COM Services release.

Module name	AUTOSAR version and revision	SWS version and revision	Module version	Supplier
Com	4.0.3 []	4.2.0 [0000]	6.3.37	Elektrobit Automotive GmbH
IpduM	4.0.3 []	2.2.0 [0000]	3.3.11	Elektrobit Automotive GmbH
PduR	4.0.3 []	3.2.0 [0000]	5.3.30	Elektrobit Automotive GmbH

Table 2.1. Hardware-Independent Modules specified by the AUTOSAR standard

2.2.3. EB (Elektrobit) modules

The following table lists all modules which are part of this release but are not specified by the AUTOSAR standard. These modules include tooling developed by EB or they may hold files shared by all other modules.

Module name	Module version	Supplier
No EB modules available		

Table 2.2. Modules not specified by the AUTOSAR standard

2.2.4. MCAL modules and EB tresos AutoCore OS

For information about MCAL modules and OS, refer to the respective documentation, which is available as PDF at `$TRESOS_BASE/doc/3.0_EB_tresos_AutoCore_OS` and `$TRESOS_BASE/doc/5.0_MCAL_modules`¹. It is also available in the online help in EB tresos Studio. Browse to the folders `EB tresos AutoCore OS` and `MCAL modules`.

2.3. Module release notes

2.3.1. Com module release notes

- ▶ AUTOSAR R4.0 Rev 3
- ▶ AUTOSAR SWS document version: 4.2.0
- ▶ Module version: 6.3.37.B206164
- ▶ Supplier: Elektrobit Automotive GmbH

2.3.1.1. Change log

This chapter lists the changes between different versions.

Module version 6.3.37

2018-06-22

- ▶ Improved handle ID wizard for Tx-I-PDUs which takes the priority of CAN messages into account
- ▶ Implemented defer Com transmission into Tx main function support
- ▶ Improved usage of critical sections for `Com_SendDynSignal`
- ▶ Implemented version compatibility check for EcuC library

¹`$TRESOS_BASE` is the location at which you installed EB tresos Studio.

Module version 6.3.36

2018-05-25

- ▶ Implemented COM TP-API support
- ▶ Implemented COM API Com_GetRxIPduBuffer which returns information about the receive IPdu buffer of a Rx IPdu

Module version 6.3.35

2018-05-07

- ▶ ASCCOM-2420 Fixed known issue: Wrongly generated code after reopening the project in EB tresos Studio

Module version 6.3.34

2018-04-20

- ▶ Implemented non-functional code improvements to fix Misra violations
- ▶ Implemented improvements for flexible allocation of PDUs to multiple Tx respectively Rx main functions including signal gateway
- ▶ ASCCOM-2410 Fixed known issue: Wrong COM_EXCLUSIVE_AREA is used

Module version 6.3.33

2018-03-16

- ▶ Implemented improvements for PduLengthType uint32 support

Module version 6.3.32

2018-02-16

- ▶ Implemented improvements for flexible allocation of PDUs to multiple Tx respectively Rx main functions
- ▶ ASCCOM-2359 Fixed known issue: Group signals are not routed by signal gateway
- ▶ ASCCOM-2360 Fixed known issue: Group signals are not routed by signal gateway
- ▶ ASCCOM-2375 Fixed known issue: Fragmented/interlaced signal groups are embedded wrong in Tx-I-PDUs
- ▶ Implemented improvements for routing of I-PDUs with different unused area values in between fragmented/interlaced signal groups
- ▶ Implemented improvements for routing of group signals where the signal group has parameter ComSignalGroupArrayAccess set to true

- ▶ Implemented improvements for flexible allocation of PDUs to multiple Tx respectively Rx main functions including signal gateway

Module version 6.3.31

2017-12-15

- ▶ ASCCOM-2299 Fixed known issue: Compilation error of Com_MainFunctionRouteSignals.c

Module version 6.3.30

2017-09-22

- ▶ Implemented improvements for flexible allocation of PDUs to several Tx respectively Rx main functions including signal gateway
- ▶ Switch from MISRA-C:2004 to MISRA-C:2012
- ▶ Introduced basic support for float64, uint64 and sint64 signal types

Module version 6.3.29

2017-08-25

- ▶ ASCCOM-2256 Fixed known issue: Wrong signal packing behaviour for 8-bit unaligned (group) signals with big endian

Module version 6.3.28

2017-07-28

- ▶ Implemented non-functional code improvements
- ▶ Implemented improvements for flexible allocation of PDUs to several Tx respectively Rx main functions (without singal gateway)

Module version 6.3.27

2017-06-30

- ▶ Implemented non-functional code improvements
- ▶ Added support for additional ComTransferProperties TRIGGERED_ON_CHANGE_WITHOUT_-REPETITION and TRIGGERED_WITHOUT_REPETITION
- ▶ Added support for flexible allocation of PDUs to several Tx respectively Rx main functions

Module version 6.3.26

2017-03-31

- ▶ Implemented non-functional code improvements

Module version 6.3.25

2017-03-03

- ▶ Improved description of configuration parameter ComTxModeTimeOffset
- ▶ Implemented non-functional code improvements
- ▶ ASCCOM-2202 Fixed known issue: Wrong invocation of ComNotification callbacks on transmission side (for non-AUTOSAR use-case only)
- ▶ Improved usage of critical sections

Module version 6.3.24

2017-02-03

- ▶ Implemented non-functional code improvements
- ▶ Implemented non-functional code improvements for optimizations
- ▶ Implemented non-functional code improvements

Module version 6.3.23

2016-12-02

- ▶ Implemented non-functional code improvements

Module version 6.3.22

2016-11-04

- ▶ ASCCOM-2157 Fixed known issue: Out of bounds access of unaligned Tx 16 bit signals / group signals
- ▶ Implemented non-functional code improvements
- ▶ Implemented non-functional code improvements to avoid compiler warnings

Module version 6.3.21

2016-09-23

- ▶ Added support for routing of fragmented/interlaced signal groups

Module version 6.3.20

2016-09-09

- ▶ Added support for fragmented/interlaced signal groups
- ▶ Adapted resource file for the scheduling of main functions to the split of `IpduM_MainFunction()` into `IpduM_MainFunctionRx()` and `IpduM_MainFunctionTx()`.

Module version 6.3.19

2016-08-05

- ▶ Implemented non-functional code improvements

Module version 6.3.18

2016-07-01

- ▶ Implemented non-functional code improvements

Module version 6.3.17

2016-05-25

- ▶ ASCCOM-2104 Fixed known issue: Unintended restarting of reception deadline monitoring with `Com_IpduGroupControl()`

Module version 6.3.16

2016-04-01

- ▶ Implemented non-functional code improvements

Module version 6.3.15

2016-02-05

- ▶ ASCCOM-2084 Fixed known issue: Nested MemMap section if `TS_MERGED_COMPILE` is activated
- ▶ Added support for Debug & Trace with custom header file configurable via parameter `BaseDbgHeaderFile`

Module version 6.3.14

2015-11-06



- ▶ ASCCOM-2071 Fixed known issue: Missing includes in source files in case option TS_MERGED_COM-PILE is disabled

Module version 6.3.13

2015-10-09

- ▶ Implemented non-functional code improvements to avoid compiler warnings (Green Hills compiler for RH850 derivative) and static code analysis tools warnings
- ▶ ASCCOM-2054 Fixed known issue: Com receives incorrect values for Big-endian (group) signals if their msb is set to a multiple of eight
- ▶ Implemented non-functional code improvements to avoid compiler warnings

Module version 6.3.12

2015-06-19

- ▶ Implemented non-functional code improvements to avoid compiler warnings for specific optimization configurations
- ▶ ASCCOM-2021 Fixed known issue: Wrong signal handling on CPUs with big endianness architecture

Module version 6.3.11

2015-05-22

- ▶ ASCCOM-2015 Fixed known issue: Compilation error due to wrong usage of MemMap
- ▶ ASCCOM-2018 Fixed known issue: Wrong API name in integration requirement EB_INTREQ_Com_0001

Module version 6.3.10

2015-04-24

- ▶ Added support for ACG7 Transformer (COM)

Note: With this feature, the handle ID policy for signals has changed. The conversion to the new handle ID policy can be simply applied by calling the *Calculate Handle IDs wizard* as described in *EB tresos Studio user's guide*.

Module version 6.3.9

2015-02-20

- ▶ Use AUTOSAR 4.x compliant memory section names for section CONFIG_DATA_UNSPECIFIED
- ▶ ASCCOM-1981 Fixed known issue: Invalid length calculation for dynamic length signals
- ▶ Implemented non-functional code improvements to fix Misra violations
- ▶ Implemented non-functional code improvements to ease readability

Module version 6.3.8

2015-01-07

- ▶ Added support for configurable mapping of PduR_IsValidConfig function to dedicated memory section
- ▶ Implemented non-functional code improvements for optimizations
- ▶ Implemented non-functional code improvements to fix Misra violations
- ▶ Implemented non-functional code improvements and update integration requirement EB_INTREQ_Com_-0002
- ▶ Implemented that reception filter MASKED_NEW_DIFFERS_MASKED_OLD always passes the filter criteria after a reception deadline monitoring timeout. For further details please refer to AUTOSAR Com SWS 4.1.1 SWS_Com_00793 and Bugzilla #52102
- ▶ Added support for ComInitialValueOnly
- ▶ Implemented range limitations for filter parameters. For further details please refer to AUTOSAR Com SWS 4.1.1, Bugzilla #52038 and #67828
- ▶ Changed signature of Com_RxIpduCallout and Com_TxIpduCallout. For further details please refer to AUTOSAR Com SWS 4.1.1, Bugzilla RfC #52342
- ▶ Removed AUTOSAR 3.x compliant symbolic name value macros and updated the logic to only provide AUTOSAR 4.0.2 compliant macros if macro COM_PROVIDE_LEGACY_SYMBOLIC_NAMES is defined

Module version 6.3.7

2014-10-03

- ▶ Added support for dynamic length signals

Module version 6.3.6

2014-08-07

- ▶ ASCCOM-1814 Fixed known issue: Build error due to missing file `Com_PBcfg.c` if code generation for Com is disabled and only post-build configuration is compiled
- ▶ Improved optimization: Allow disabling of Transmission Mode Selection
- ▶ Added support for signal group array access.

- ▶ ASCCOM-1856 Fixed known issue: Transmission of a signal gateway destination I-PDU fails if I-PDU shall be transmitted because of a change of signal group values
- ▶ ASCCOM-1836 Fixed known issue: Initial TMS evaluation fails for filter parameter with values larger than 0x7FFFFFFF
- ▶ ASCCOM-1874 Fixed known issue: Wrong warning if a byte array is larger than 8 bytes
- ▶ ASCCOM-1875 Fixed known issue: Error is issued when config time support is enabled and ComIPduTriggerTransmitCallout is configured
- ▶ Added support I-PDUs larger than 254 Bytes

Module version 6.3.5

2014-04-25

- ▶ ASCCOM-1770 Fixed known issue: Build fails if source files shall be built separately
- ▶ ASCCOM-1781 Fixed known issue: Sign extension for Rx-signals may fail if generated Rx-signal API is used
- ▶ ASCCOM-1785 Fixed known issue: Com module configuration generator may generate incorrect compiler abstractions for type definitions of module internal data types
- ▶ Implemented non-functional code improvements to fix Misra violations
- ▶ ASCCOM-1790 Fixed known issue: Missing checks of configuration parameters for transmission modes
- ▶ Implemented non-functional code improvements to avoid compiler warning in `Com_MainFunction-RouteSignals.c`
- ▶ ASCCOM-1803 Fixed known issue: Nested `MemMap` section if `TS_MERGED_COMPILE` is activated
- ▶ ASCCOM-1813 Fixed known issue: Choice container `ComGwDestination` is not set to changeable at post-build time

Module version 6.3.4

2013-10-11

- ▶ Implemented non-functional code improvements to use critical sections symmetrically
- ▶ Implemented non-functional code improvements to reduce function parameter in order to meet HIS metrics
- ▶ Implemented non-functional code improvements to fix Misra violations
- ▶ Replaced Rte memory sections and compile abstractions with Com memory sections and compile abstractions
- ▶ Improved allocation of post-build memory to ensure proper alignment
- ▶ Improved configuration checks of `ComTransferProperty`

- ▶ Implemented non-functional code improvements to defensive programming
- ▶ Implemented non-functional code improvements for optimizations
- ▶ Changed data type of `Com_StatusCode` from `uint8` to an enumeration (used by API `Com_GetStatus()`)
- ▶ Updated behavior of `Com_SendSignalGroup()` regarding the calculation of the transmission mode of the related I-PDU
- ▶ Updated checks for timing parameters that resulting number of ticks matches are exact to configured values (according to TPS_ECUC_08010 of Specification of ECU Configuration AUTOSAR 4.1.1)
- ▶ Updated calculation of mask for filter `MASKED_NEW_DIFFERS_MASKED_OLD`
- ▶ Improved MCG to generate XML code for Binary Code Generation
- ▶ Added consistency checks for Com configuration
- ▶ Implemented non-functional code improvements to clean up service IDs
- ▶ Changed VSMDs to adhere to additional VSMD rules specified for AUTOSAR 4.1.1 related to attribute post-build changeable

Module version 6.3.3

2013-06-28

- ▶ ASCCOM-1685 Fixed known issue: `Com_TriggerTransmit()` incorrectly returns `E_OK` when all I-PDU groups to which an I-PDU belongs are stopped
- ▶ Changed timing behavior of starting of periodic I-PDUs according to the clarification in Bugzilla #52352
- ▶ Implemented a default value 0 for `ComTxModeTimeOffset` according to the clarification in Bugzilla #52352
- ▶ ASCCOM-1666 Fixed known issue: Restriction on I-PDU Trigger Transmit Callout
- ▶ ASCCOM-1699 Fixed known issue: `Com_RxIndication()` accesses invalid memory if called while uninitialized

Module version 6.3.2

2013-05-10

- ▶ Implemented check of published information signature to prevent loading of incompatible post-build configuration

Module version 6.3.1

2013-02-08

- ▶ Updated default value of filter of Tx-signals according to COM676 and COM677

Module version 6.3.0

2012-10-12

- ▶ Updated to AUTOSAR 4.0 Handle ID policy
- ▶ Added support of configuration parameter `ComRetryFailedTransmitRequests`
- ▶ Changed the top-level structure of the SWC description in the arxml files from `/AUTOSAR/Com` to `/AUTOSAR_Com`
- ▶ Added support for extended handling of configuration parameter `ComFirstTimeout`
- ▶ Added support of configuration parameter `ComEnableMDTForCyclicTransmission` (disable MDT for cyclic transmission)
- ▶ Added new API `Com_SwitchIpduTxMode()`
- ▶ Added separate I-PDU callout for `Com_TriggerTransmit()`
- ▶ Added support of reception of shorter I-PDUs (see deviation *Restricted support of small Rx-I-PDUs*)

Module version 6.2.2

2012-08-17

- ▶ Added definition of Exclusive Area Activation in Basic Software Module Description

Module version 6.2.1

2012-06-20

- ▶ Added support of usage of PbcfgM module

Module version 6.2.0

2012-03-16

- ▶ Modified `SchM_Enter/Exit()` calls to match AUTOSAR 4.0
- ▶ Added support for Tx-timeout handling for transmission mode `NONE`
- ▶ Updated naming scheme for `#defines` for symbolic name values to AUTOSAR 4.0 Rev 3 naming scheme
- ▶ Updated initial value for Rx-signals and signal groups when Rx Deadline Monitoring expired and I-PDU group is stopped
- ▶ Updated Com configuration to AUTOSAR 4.0 Rev 3
- ▶ Improved error message in case invalid references are configured

Module version 6.1.1

2012-02-17

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 6.1.0

2012-01-20

- ▶ Changed I-PDU group control API (removed AUTOSAR 3.x API, introduced AUTOSAR 4.x API)
- ▶ Updated Minimum Delay Time Monitoring behavior according to AUTOSAR 4.0
- ▶ ASCCOM-1435 Fixed known issue: Minimum Delay Time Monitoring is not canceled when Transmission Deadline Monitoring expires
- ▶ ASCCOM-1437 Fixed known issue: Possible omission of transmission of an I-PDU if `ComTransferProperty = TRIGGERED_ON_CHANGE` is used
- ▶ Added generation of BSWMD

Module version 6.0.1

2011-09-30

- ▶ ASCCOM-1318 Fixed known issue: Compilation fails when `Com.h` and `PduR.h` is included in the same file
- ▶ Changed optimization configuration from `ComTxFilterMaskedNewDiffersMaskOldEnable` to `Com_TxF_MaskNewDiffersMaskOld_En`
- ▶ Added reception of Rx-PDUs which are longer than configured

Module version 6.0.0

2011-09-02

- ▶ Initial AUTOSAR 4.0 version

2.3.1.2. New features

- ▶ Implemented optimization for flexible allocation of PDUs to multiple Tx respectively Rx main functions.
- ▶ Implemented COM TP-API support.
- ▶ Implemented COM API `Com_GetRxIPduBuffer` which returns information about the receive IPdu buffer of a Rx IPdu.
- ▶ Improved usage of critical sections for `Com_SendDynSignal`.

- ▶ Implemented defer Com transmission into Tx main function support.
- ▶ Improved handle ID wizard for Tx-I-PDUs which takes the priority of CAN messages into account.

2.3.1.3. EB-specific enhancements

This chapter lists the enhancements provided by the module.

- ▶ [HisCom0029] Compile-time signal endianness optimization (extension to AUTOSAR specification)

Description:

If (at compile time) the `ComSignalEndianness` (COM157) is identical for every `ComSignal` of the COM module, the Com module does the following:

- ▶ It uses only the code required for this endianness.
 - ▶ It does not make runtime checks for the signal endianness.
 - ▶ It does not store the endianness of every `ComSignal` individually.
- ▶ [HisCom0009] The `Com_ReceiveSignal` API as defined in COM198 is implemented as access macro

Description:

If this optimization is used, macros and a function is generated which extract the value of a signal from the I-PDU.

Rationale:

If these macros or the generated `Com_ReceiveSignal` API are used, the access to the value of the signal is faster.

- ▶ Optional reception filter for Signal Gateway

Description:

According to AUTOSAR, an Rx-signal is always gated via the Com Signal Gateway if a gateway relation for that signal exists. The update-bit is not considered here. In this implementation the AUTOSAR-conform behavior is achieved when the vendor-specific parameter `ComSigGwRxFilterEnable` is set to false which is the default value. However, when `ComSigGwRxFilterEnable` is set to true, the signal is only gated via the Com Signal Gateway, when the filter of the Rx-signal evaluates to true.

Rationale:

Gated signals can be filtered.

- ▶ Optional Tx-signals with size zero

Description:

A zero size signal is a signal which is not represented in an I-PDU. However, an application can send a value which is evaluated by a filter configured for that signal. This feature is enabled when the vendor-specific parameter `ComTxZeroSignalEnable` is set to true and the default value is false.

Rationale:

Trigger-sending of an I-PDU without changing a value within the I-PDU.

- Support for signal group array access

Description:

The new APIs `Com_SendSignalGroupArray()` and `Com_ReceiveSignalGroupArray()` access the signal group value in the I-Pdu buffer.

Rationale:

Allows efficient access to signal groups, especially if the serialized data are also provide/required by another module, e.g. E2E module.

- Support for I-PDUs larger than specified by AUTOSAR

Description:

In contrast to AUTOSAR which restricts the configuration of signals / group signals into I-PDUs to at most 254 Bytes / large I-PDUs to at most 4095 bytes, the COM module supports I-PDUs up to 8191 Bytes. Further, the COM module allows the configuration of opaque signals / group signals (`ComSignalType` equals OPAQUE) with a length up to 8191 Bytes.

With EcuC parameter `PduLengthTypeEnum` configured to `UINT32`, the module is basically allowed to handle PDUs with user data of more than 64 KiB. With the length restriction imposed by 8191 bytes neither the `UINT16` nor the `UINT32` range can be exploited.

Rationale:

Communication with Ethernet frames requires increased length of I-PDUs.

- Support for ACG7 Transformer (COM)

Description:

The ACG7 Transformer (COM) uses the post-build configuration and the serialization / de-serialization functions of the Com module.

Rationale:

Ensures consistent configuration between the Com and ComXf and allows the efficient serialization / de-serialization for signals and group signals due to the updated read / write library.

2.3.1.4. Deviations

This chapter lists the deviations of the module from the AUTOSAR standard.

- Only post-build configuration is supported

Description:

The Com module only supports configuration variant `VARIANT-POST-BUILD`. `VARIANT-PRE-COMPILE` and `VARIANT-LINK-TIME` are not supported.

Requirements:

COM606, COM607

- Signal invalidation is not supported (but is supported via RTE) (reference to product description: ASCPD-15)

Description:

Signal invalidation is not supported. However, the EB tresos AutoCore RTE is extended in order to provide the signal invalidation functionality based on the configuration of the `Com` module.

Requirements:

COM099, COM286, COM680, COM681, COM736, COM683, COM737, COM717, COM718, COM334, COM024, COM203, COM642, COM643, COM288, COM644, COM557, COM645, COM536, COM315_Conf, COM391_Conf, COM314_Conf, COM738, COM682, COM483, COM396, COM005, COM731

- Signal-based gateway: Optimization issue rate conversion not supported

Description:

COM386: Optimization issue: An I-PDU can be configured to be handled en bloc if it contains signals to be routed completely via a transmit I-PDU through a retention of the signal order and the signals endianness (related use case: rate conversion).

The implementation lacks this optimization since AUTOSAR defines no configuration parameter to define to handle the I-PDU en bloc. Workaround: In case the I-PDU which should be gated is not used on the ECU, the whole I-PDU could be defined as a array. Note: If the size is greater than 8 bytes, the `init` value can not be defined with the AUTOSAR configuration parameters. If you use such a configuration, the PDU is directly copied to the Tx-PDU. Nevertheless, in case the signals shall be extracted by the Com module for an application, rate conversion can also be done. But in this case all signals are extracted from the Rx-PDU and packed into the Tx-PDU.

Requirements:

COM386

- **Signal-based gateway:** The `ComBitSize` of a received `ComSignal` can differ from the routed `ComSignal`

Description:

In contrast to AUTOSAR which specifies that the `ComBitSize` of the received and the routed `ComSignal` shall not differ, the Com module allows a `ComBitSize` of the routed `ComSignal`. This `ComBitSize` is greater than the `ComBitSize` of the received `ComSignal` with the constraint that both must be of the same `DataType`.

Requirements:

COM384

- **SigGW:** `ComGwSourceDescription` and `ComGwDestinationDescription` are not supported (reference to product description: ASCPD-17)

Description:

AUTOSAR SWS COM548_Conf and COM549_Conf define a `ComGwSourceDescription` and `ComGwDestinationDescription`. These descriptions allow adding/changing gateway relations post-build without the configuration of new signals `ComGwSourceDescription` and `ComGwDestinationDescription` that are not supported.

Requirements:

COM548_Conf, COM550_Conf, COM549_Conf

- `Com_SendSignal()` does not return `COM_SERVICE_NOT_AVAILABLE` in case the value of the signal does not fit into the PDU

Description:

The function `Com_SendSignal()` does not return `COM_SERVICE_NOT_AVAILABLE` in case the value of the signal does not fit into the PDU, but an error is reported to DET. However, the SWS states: Return value: `E_OK` - service has been accepted `COM_SERVICE_NOT_AVAILABLE` - corresponding I-PDU group was stopped (or service failed due to development error). Therefore a `COM_SERVICE_NOT_AVAILABLE` should be returned.

Requirements:

COM197

- The content of unfiltered elements of `ComSignal` in I-PDUs which are received deferred is not preserved until the next call to `Com_MainFunctionRx`

Description

In case the upper layer calls `Com_ReceiveSignal` or `Com_ReceiveSignalGroup` after an I-PDU with `ComIPduProcessing deferred` was received (a lower layer called `RxIndication()`), but before the

deferred indications were signaled to the upper layer in `Com_MainFunctionRx`. This implementation does not behave like described in a note in the Com specification, but as follows: The content of the *old* I-PDU is not preserved until the next call to `Com_MainFunctionRx`. Immediately after the reception of the new I-PDU, the `Com_ReceiveSignal` and `Com_ReceiveSignalGroup` API provides the data of the new I-PDU's content. The only exception are non-group signals with a filter. As the filter is evaluated in the `Com_MainFunctionRx()`, the signal's value is retained until then.

Requirements:

COM198, COM201

- ▶ The Com does not check if an I-PDU is started if `Com_TxConfirmation` is called

Description

In contrast to AUTOSAR which states in Table 5 that a call to `Com_TxConfirmation()` shall be ignored in case of a stopped I-PDU, this version does not check if an I-PDU is started or stopped when the lower layer calls `Com_TxConfirmation()`.

Rationale:

In order not to lose speed to this check, it is assumed that no sporadic Tx-Confirmation appear once an Tx-I-PDU has been stopped.

Requirements:

COM124

- ▶ I-PDUs of gated signals are not sent out from the `Com_MainFunctionRouteSignals()` but from `Com_MainFunctionTx()`

Description:

In COM466 it is stated that I-PDUs that contain gated signals with triggered transfer property shall be sent from the `Com_MainFunctionRouteSignals()` according to their transmission modes. In the implementation, an I-PDU is never sent from the `Com_MainFunctionRouteSignals()`. In case a gated signal has the triggered transfer property, the I-PDU is sent out in the following `Com_MainFunctionTx()`. However, both `Com_MainFunctionRouteSignals()` and `Com_MainFunctionTx()` are scheduled functions which therefore do not have the *event-based* character as does `Com_SendSignal()` with the triggered transfer property.

Note: The transmission request from the signal gateway is also cleared, independent if a transmission request was issued and/or a transmission request failed if the following conditions are met:

- ▶ `ComRetryFailedTransmitRequest` is enabled.
- ▶ A transmission deadline monitoring is configured for the I-PDU and the deadline monitoring expires in the following `Com_MainFunctionTx()`.

Requirements:

COM539

- Data sequence control and Communication protection not supported (reference to product description: ASCPD-22, ASCPD-23)

Description:

Data sequence control (I-PDU counter) and communication protection (replication of I-PDUs) are not supported.

Requirements:

COM687, COM688, COM587, COM588, COM590, COM727, COM596, COM597, COM726, COM592_-Conf, COM003_Conf, COM593_Conf, COM594_Conf, COM595_Conf, COM599_Conf, COM600_Conf, COM601_Conf, partly COM787, COM731

- *Transmit Cancellation* not supported (reference to product description: ASCPD-24)

Description:

The cancelation of transmission requests is not supported.

Requirements:

COM708, COM670, COM709_Conf

- Restricted support of small Rx-I-PDUs

Description:

According to the AUTOSAR COM SWS chapter *Signal indication (Unpacking of I-PDUs)* it is specified that it is allowed that smaller than expected Rx-I-PDUs can be received (configured). In such a case partly or not received signals/signal groups shall not be updated and no notification via `ComNotification` shall take place.

However, the implementation behaves as follows:

- The received data length (`PduInfoPtr->SduLength`) is copied into the Com-internal I-PDU buffer. If a signal or signal groups are received only partly, these are also updated partly. Signals/signal groups which are not received at all are not updated. If the I-PDU contains a dynamic length signal the API `Com_ReceiveDynSignal()` does not copy data and 0 is returned in the length parameter. If a dynamic length signal is used within the signal gateway, the length of the corresponding Tx dynamic length signal is also set to 0.
- `ComNotification` is invoked for all signals or signal groups that belong to this Rx-I-PDU.

Workaround 1:

- ▶ For a smaller Rx-I-PDU, for which it is expected that a signal or signal group is only partly updated: Configure an I-PDU callout which updates the partly received signal/signal groups with a proper value (either last received value or initial value).
- ▶ Design the applications in a way that they can handle `ComNotification` of signals which are not or only partly received.

Workaround 2:

- ▶ Provide an application for each expected size of the Rx-I-PDU.
- ▶ For each expected size of the Rx-I-PDU configure a Rx-I-PDU in the Com module.
- ▶ Create a mapping between the additional applications and Rx-I-PDUs.
- ▶ Configure an Rx-I-PDU callout with the large I-PDU which invokes `Com_RxIndication` of the respective smaller Rx-I-PDU and returns `FALSE` in case a shorter I-PDU is received.

Rationale:

In general this limitation allows a more efficient implementation for I-PDUs which are received completely. Workarounds are available if this feature is required.

Requirements:

COM574, COM575

- ▶ No support of Debug & Trace

Description:

Tracing of global variables is not supported.

Requirements:

COM745, COM746, COM747, COM748

- ▶ Non-compliant deviations in vendor-specific module definition file

Description:

The vendor-specific module definition file (VSMD) has non-compliant deviations to the AUTOSAR specification:

Violations against Rule `EcucSws_6008`: The `LOWER-MULTIPLICITY` of an element in the VSMD must be bigger or equal and the `UPPER-MULTIPLICITY` must be equal or less than in the StMD

- ▶ `StMD-Node`: `/AUTOSAR/Com/ComConfig/ComIPdu/ComIPduTriggerTransmitCallout`

Rationale: Configuration shall be equal with `ComCallout`, see also http://www.autosar.org/bugzilla/show_bug.cgi?id=53200#c50.

- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComTimeBase

Rationale: Optionality of the ComTimeBase container is used to enable or disable the multiple main function support.

Violations against Rule EcucSws_1007: For integer and float parameters the MIN values must be >= and the MAX values <= as in the StMD.

- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComGwMapping/ComGwDestination/ComGwDestinationDescription/ComFilter/ComFilterMask
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComGwMapping/ComGwDestination/ComGwDestinationDescription/ComFilter/ComFilterMax
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComGwMapping/ComGwDestination/ComGwDestinationDescription/ComFilter/ComFilterMin
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComGwMapping/ComGwDestination/ComGwDestinationDescription/ComFilter/ComFilterOffset
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComGwMapping/ComGwDestination/ComGwDestinationDescription/ComFilter/ComFilterPeriod
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComGwMapping/ComGwDestination/ComGwDestinationDescription/ComFilter/ComFilterX
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComIPdu/ComTxIPdu/ComTxModeFalse/ComTxMode/ComTxModeTimeOffset
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComIPdu/ComTxIPdu/ComTxModeTrue/ComTxMode/ComTxModeTimeOffset
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignal/ComBitPosition
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignal/ComSignalLength
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignal/ComUpdateBitPosition
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignal/ComFilter/ComFilterMask
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignal/ComFilter/ComFilterMax
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignal/ComFilter/ComFilterMin
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignal/ComFilter/ComFilterX
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignalGroup/ComGroupSignal/ComUpdateBitPosition
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignalGroup/ComGroupSignal/ComBitPosition
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignalGroup/ComGroupSignal/ComSignalLength

- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignalGroup/ComGroupSignal/ComFilter/ComFilterMask
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignalGroup/ComGroupSignal/ComFilter/ComFilterMax
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignalGroup/ComGroupSignal/ComFilter/ComFilterMin
- ▶ **StMD-Node:** /AUTOSAR/Com/ComConfig/ComSignalGroup/ComGroupSignal/ComFilter/ComFilterX

Rationale: Limitations are necessary to ensure valid configuration of filters (see Limitation: Restriction on `ComFilter` values). The range of `ComTxModeTimeOffset` has been extended to allow a backward compatible configuration for starting of I-PDU groups. Value range for parameters `ComBitPosition`, `ComUpdateBitPosition`, and `ComSignalLength` of `ComSignals` and `ComSignalGroups` has been extended to support I-PDUs larger than 254 Bytes.

Violations against Rule `EcucSws_1014`: Additional vendor specific parameter definitions (using `ParameterTypes`), container definitions and references shall be added to the VSMD according to the alphabetical order.

- ▶ **StMD-Node:** /AUTOSAR/Com
- ▶ **StMD-Node:** /AUTOSAR/ComConfig/ComSignal
- ▶ **StMD-Node:** /AUTOSAR/ComGeneral

Rationale: Additional vendor specific parameter definitions have a specific order in the GUI which may differ to the alphabetical order.

- ▶ No consistency check between code files and header files

Description:

The inter-module version checks as specified by the Com SWS are not implemented.

Rationale:

- ▶ The required compile-time version checks would result in an inflexible, hardly integratable basic software stack.
- ▶ EB tresos AutoCore is an already integrated product.
- ▶ The project handling of EB tresos Studio provides means to enforce that only modules with the same EB tresos AutoCore release version can be added to the project.

Requirements:

COM673

- ▶ Behavior of `Com_IpduGroupControl`

Description:

According to COM787, the Com module shall initialize the following attributes of an I-PDU as result of a call `Com_IpduGroupControl` (independent from parameter `Initialize`):

1. `ComMinimumDelayTime` of I-PDUs in transmission mode `DIRECT` or `MIXED`
2. Timeout attributes of I-PDUs for deadline monitoring aspect: all timeout timers (`ComFirstTimeout`, `ComTimeout`) shall restart.
3. All included update-bits shall be cleared.
4. Reset OCCURRENCE of filter with `ComFilterAlgorithm ONE_EVERY_N`.
5. Set the I-PDU counter to 0 for I-PDUs with `ComIPduDirection` configured to `SEND`.
6. Accept for I-PDUs with `ComIPduDirection` configured to `RECEIVED` any next incoming I-PDU counter.

However the implementation does not support I-PDU counter, therefore the items 5 and 6 are not initialized. See also deviation *Data sequence control and Communication protection not supported*. The items 1, 3 and 4 are only initialized as the result of a call `Com_IpduGroupControl` with parameter `Initialize` set to true. In contrast to COM222, the shadow buffers of included `RECEIVED` signal groups keeps unchanged if `Com_IpduGroupControl` is called (independent from parameter `Initialize`).

Additionally to the described behavior above, `ComTxModeTimePeriod` and `ComTxModeTimeOffset` of I-PDUs in `PERIODIC` or `MIXED` transmission mode are always respected (independent from parameter `Initialize`).

Rationale:

Requirement COM787 is not backward compatible, see also Bugzilla issue http://www.autosar.org/bugzilla/show_bug.cgi?id=48891.

The description of parameter `Initialize` of the API `Com_IpduGroupControl` (COM751) says: “flag to request initialization of the I-PDUs which are newly started”. That describes well the implemented behavior, but does not imply that I-PDU attributes are initialized, although parameter `Initialize` set to false.

The description of `ComTxModeTimeOffset` refers to `Com_IpduGroupControl` in general and is not limited to parameter `Initialize` set to true.

Requirements:

COM787, COM222

- No generation of symbolic name value into `Com_Cfg.h`

Description:

Several requirements claim that the symbolic names for the Com Handle IDs shall be published via `Com_Cfg.h`. However, the symbolic name values are provided in `Com_SymbolicNames_PBcfg.h` which is also included in `Com.h`.

Rationale:

- ▶ Requirement is a deviation against TPS_ECUC_02108 of Specification of ECU Configuration which says that the symbolic name values shall be generated into the module header file.
- ▶ Requirement is a deviation against SWS_BSW_00200 of SWS General Specification of Basic Software Modules AUTOSAR 4.1 Rev 1, which says that symbolic name values shall be imported through the header of the BSW module that provides the value.
- ▶ Shall be removed in future AUTOSAR releases, see http://www.autosar.org/bugzilla/show_bug.cgi?id=60888

Requirements:

COM174, COM126, COM163, COM044, COM521

- ▶ No support of dynamic length signals in signal groups

Description:

Dynamic length signals are only supported as signals. They are not supported in a group signal.

Rationale:

The implementation uses `Com_UpdateShadowSignal()` and `Com_ReceiveShadowSignal()` for the access of group signals. Since AUTOSAR does not define an equivalent API for access dynamic group signals, it is not possible to support dynamic length signals for group signals.

Requirements:

COM127

- ▶ No support of zero size signals / group signals with transfer property PENDING

Description:

In contrast to AUTOSAR which allows zero size signals / group signals for transfer properties TRIGGERED, PENDING, and TRIGGERED_WITHOUT_REPETITION, only transfer property TRIGGERED and TRIGGERED_WITHOUT_REPETITION is supported.

Requirements:

COM762

- ▶ No need for configuration of `ComTxModeTrue` or `ComTxModeFalse`

Description:

In contrast to AUTOSAR which states that every ComTxModeTrue or ComTxModeFalse that is a potential result of a configured/ calculated TMS must be configured, the COM module assumes ComTransmission-Mode NONE if one of these transmission modes is not configured but evaluated as a result of TMS. Note that at least one of the containers ComTxModeTrue or ComTxModeFalse must be configured at all.

Requirements:

COM465

- Overlapping of ComSignals / ComGroupSignals

Description:

In contrast to AUTOSAR which states that ComSignal / ComGroupSignal are not allowed to overlap each other, the COM module allows the configuration of overlapped ComSignals / ComGroupSignals.

Requirements:

COM102

- Configurable callback / callout functions are not provided in Com_Cbk.h

Description:

In contrast to AUTOSAR which states that the configurable callback and callout functions shall be provided in header file Com_Cbk.h, the COM module does not declare these functions. Instead, it declares and calls these external functions in an internal Com compilation unit.

Rationale:

These functions are usually generated / implemented by the Rte which also generates adequate function declarations. The linker then is able to resolve the function calls and the adequate function definitions in Rte. See also RTE Specification 4.2.1 Section 5.9.2.1 Call-backs for communication over AUTOSAR COM.

Requirements:

COM731

- Optimization parameter ComSignalGwEnable for scaling down signal gateway to no size

Description:

In contrast to AUTOSAR which states that the signal gateway of the AUTOSAR COM module shall scale down to no size if no signal routing functionality is needed, the integrator shall disable the vendor specific optimization parameter ComSignalGwEnable to get the same effect.

Requirements:

COM370

- ▶ Runtime error COM_E_SKIPPED_TRANSMISSION is not supported

Description:

In case a large I-PDU is currently transmitted and the same I-PDU is triggered for transmission again, the AUTOSAR COM skips the additionally send request but does not report the runtime error COM_E_SKIPPED_TRANSMISSION.

Requirements:

SWS_Com_00863

- ▶ PduR_ComTpTransmit is called for large I-PDUs

Description:

AUTOSAR specifies that for transmissions of large I-PDUs the PduR API PduR_ComTransmit has to be called. PduR_ComTpTransmit is used instead for large I-PDUs and PduR_ComTransmit for normal I-PDUs.

Requirements:

COM759, COM760, COM467, COM773, COM698

2.3.1.5. Limitations

This chapter lists the limitations of the module. Refer to the module references chapter *Integration notes*, subsection *Integration requirements* for requirements on integrating this module.

- ▶ Implementation-specific restrictions

Description:

There are some implementation-specific restrictions which are listed for completeness only, as they are most probably irrelevant for the intended use of the module:

- ▶ The maximum number of signals allowed is 65534.
- ▶ The maximum number of Rx/Tx I-PDUs allowed is 65534.
- ▶ The maximum number of callouts configured is 65534.
- ▶ The sum of the lengths of all byte-arrays which are sent via the Com module must not exceed 65535 bytes.
- ▶ The number of signals and signal group members, signal groups, notifications per I-PDU must not exceed 254.

- Discrepancy between ISO C90 standard and AUTOSAR ranges for signed integers

Description:

C90 only allows signed integer values to have the following range:

- sint8: $-127 / 127$, or $-(2^7 - 1) / 2^7 - 1$
- sint16: $-32767 / 32767$, or $-(2^{15} - 1) / 2^{15} + 1$
- sint32: $-2147483647 / 214743647$, or $-(2^{31} - 1) / 2^{31} - 1$

AUTOSAR, in the opposite, allows negative numbers to be one less:

- sint8: $-128 / 127$, or $-(2^7) / 2^7 - 1$
- sint16: $-32768 / 32767$, or $-(2^{15}) / 2^{15} + 1$
- sint32: $-2147483648 / 214743647$, or $-(2^{31}) / 2^{31} - 1$

Rationale:

In AUTOSAR, it is defined that AUTOSAR only supports platforms which use the 2's complement as basis for their architecture (and therefore support the AUTOSAR number range). This implementation implements the full AUTOSAR number range. C90-compliant compilers may legally facilitate code which shows undefined behavior in case the number-range as defined in C90 is left. That is if the minimum signed integer numbers as defined by AUTOSAR are used. Therefore this implementation relies upon the following:

- A platform is used which uses the 2's complement or a platform is used which can handle the whole AUTOSAR number range.
 - A C-compiler is used which can handle the AUTOSAR-defined minimum signed integer numbers and facilitates code which behaves properly.
- Limitation/extension on configuration of `ComFirstTimeout`

Description:

According to the AUTOSAR SWS Com requirement COM716, the AUTOSAR Com module shall not monitor the reception of this signal or of a signal group from the start of the corresponding I-PDU until the first reception. This behavior applies if the configuration parameter `ComFirstTimeout` for a signal or signal group is omitted or configured to 0.

The implementation behaves as following regarding the configuration parameter `ComFirstTimeout` for a signal or signal group:

- If configured to 0: as defined in COM716
- If omitted: `ComTimeout` is used for `ComFirstTimeout`

The default behavior for `ComFirstTimeout` is disabled.

Rationale:

- ▶ Eases configuration (otherwise `ComFirstTimeout` has to be configured for every signal/signal group)
- ▶ Does not restrict a use-case since configuration for starting of reception deadline monitoring with first reception of the I-PDU is possible.

Requirements:

COM716

- ▶ Restriction on `ComFilter` values

Description:

In contrast to AUTOSAR 4.0 Rev 3 where the `ComFilter` values (i.e. `ComFilterX`, `ComFilterMask`, `ComFilterMax`, `ComFilterMin`) for Com signals/signal groups shall have a configurable value range within [0, 18446744073709551615], the Com supports a value range within [0, 4294967295]. That is, only the least significant 32 bits are significant. Note: A Bugzilla item exists for this issue: http://www.autosar.org/bugzilla/show_bug.cgi?id=52038.

Rationale:

- ▶ Eases configuration
- ▶ There is no use-case for a filter `ONE_EVERY_N` where `ComFilterOffset` and `ComFilterPeriod` have to be higher than $2^{32}-1$.

Requirements:

COM147_Conf, COM235_Conf, COM317_Conf, COM318_Conf

- ▶ Restriction on 64 bit signals/group signals

Description:

The following restrictions for signals/group signals with `ComSignalType` configured to `UINT64` apply:

- ▶ The `ComBitPosition` is restricted to be byte aligned.
- ▶ The `ComBitSize` is restricted to be a multiple of 8 bit.
- ▶ The `ComFilterAlgorithm` is limited to `ALWAYS`, `NEVER` and `ONE_EVERY_N`.

The following restrictions for signals/group signals with `ComSignalType` configured to `SINT64` apply:

- ▶ The `ComBitPosition` is restricted to be byte aligned.
- ▶ The `ComBitSize` is restricted to be 64 bit.
- ▶ The `ComFilterAlgorithm` is limited to `ALWAYS`, `NEVER` and `ONE_EVERY_N`.

Requirements:

COM675, COM391_Conf, COM170_Conf

- ▶ Limitation on transmission behaviour for large Tx I-PDUs

Description:

The transmission behaviour of large Tx I-PDUs is limited to the following points:

- ▶ only one transmission mode can be enabled.
- ▶ only transmission mode `DIRECT` with no repetitions (`ComTxModeNumberOfRepetitions` set to 0) is allowed.
- ▶ all large Tx I-PDU transmission requests are deferred to the next execution of the Com transmission main function (parameter `ComDeferTx2MainFunc` must be enabled).
- ▶ all update-bits of all contained signals and signal groups of large Tx I-PDUs are only cleared if `PduR_ComTpTransmit` returned `E_OK` and the I-PDU was successfully confirmed (parameter `ComTxIPduClearUpdateBit` must be configured to Confirmation).

Due to that limitation no transmission mode selection based on Tx transmission filter evaluation is possible for large Tx I-PDUs.

Requirements:

COM694, COM602, COM325, COM380, COM439, COM231, COM330, COM767, COM734, COM768, COM762, COM135, COM741, COM769, COM742, COM770, COM326, COM676, COM678, COM679, COM245, COM763, COM238, COM244, COM495, COM582, COM467, COM279, COM305, COM494, COM392, COM776, COM787, COM222, COM223, COM228, COM229, COM789, COM696, COM308, COM739, COM388, COM492, COM784, COM813, COM605, COM032

2.3.2. IpduM module release notes

- ▶ AUTOSAR R4.0 Rev 3
- ▶ AUTOSAR SWS document version: 2.2.0
- ▶ Module version: 3.3.11.B206164
- ▶ Supplier: Elektrobit Automotive GmbH

2.3.2.1. Change log

This chapter lists the changes between different versions.

Module version 3.3.11

2018-06-22

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.3.10

2018-05-25

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.3.9

2018-04-20

- ▶ Internal module improvement. This module version update does not affect module functionality
- ▶ Added PbcfgM support
- ▶ Improved size announcement for trigger transmit Container Tx
- ▶ Add support for UINT32 PduLengthType.

Module version 3.3.8

2018-02-16

- ▶ Create per-Partition BswImplementation and BswInternalBehavior elements in BSWMD
- ▶ ASCIPDUM-885 Fixed known issue: Compilation fails for enabled IpduMDedicatedIpduProcessingSupport

Module version 3.3.7

2018-01-19

- ▶ ASCIPDUM-885 Fixed known issue: Out-of-bounds access for deferred container Rx PDUs
- ▶ Flexible allocation of PDUs to MainFunctions

Module version 3.3.6

2017-12-15

- ▶ ASCIPDUM-881 Fixed known issue: Container PDU is not transmitted when recovered from bus off situation
- ▶ Use BinarySearch for matching ContainedPdu ID
- ▶ ASCIPDUM-884 Fixed known issue: Generator error for timeout timers divisible by IpduMTxTimeBase

Module version 3.3.5

2017-11-17

- ▶ ASCIPDUM-872 Fixed known issue: Missing TxConfirmation of contained PDUs
- ▶ ASCIPDUM-873 Fixed known issue: Loss of data for bursts of contained PDUs
- ▶ Deferred finalization (frozen) for TriggerTransmit Container PDU

Module version 3.3.4

2017-10-20

- ▶ ASCIPDUM-871 Fixed known issue: IpduM_MainFunctionRx() blocks the interrupt too long

Module version 3.3.3

2017-09-22

- ▶ ASCIPDUM-856 Fixed known issue: Multi-PDU-to-container handling is not post-build capable

Module version 3.3.2

2017-08-25

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.3.1

2017-07-28

- ▶ Improve sending of container PDU due to send timeout trigger
- ▶ ASCIPDUM-837 Fixed known issue: Container PDU delayed with transmission confirmation timeout time
- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.3.0

2017-06-30

- ▶ Internal module improvement. This module version update does not affect module functionality
- ▶ Lower layer module isn't informed about the transmit request in case the trigger transmit mode is used for the container PDU

- ▶ ASCIPDUM-831 Fixed known issue: Transmission timer is not initialized when adding a contained I-PDU to a new container PDU

Module version 3.2.18

2017-06-02

- ▶ ASCIPDUM-804 Fixed known issue: Contained PDU is not transmitted in case of IPDUM_COLLECT_LAST_IS_BEST
- ▶ ASCIPDUM-805 Fixed known issue: Container PDU transmitted with wrong contained PDUs in case of IPDUM_COLLECT_LAST_IS_BEST
- ▶ ASCIPDUM-808 Fixed known issue: Container PDU is not transmitted on the network for a long period
- ▶ ASCIPDUM-811 Fixed known issue: Container PDU transmits wrong contained PDUs in case of IPDUM_COLLECT_QUEUED
- ▶ ASCIPDUM-814 Fixed known issue: Container PDU transmitted twice followed by the loss of the next instance of the container PDU
- ▶ ASCIPDUM-817 Fixed known issue: Corruption of run-time data during IpduM_TxConfirmation()
- ▶ IPDUM_GET_SHORT_HEADER_ID depends on CPU_BYTE_ORDER
- ▶ Removed restriction to little-endian byte order for contained I-PDU headers (Multiple-PDU-to-Container handling)

Module version 3.2.17

2017-05-05

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.2.16

2017-03-31

- ▶ ASCIPDUM-768 Fixed known issue: Invalid memory access when Container Tx PDU length exceeds configured PduLength
- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.2.15

2017-03-03

- ▶ ASCIPDUM-758 Fixed known issue: Header ID and DLC are processed incorrectly on big-endian platform
- ▶ Internal module improvement. This module version update does not affect module functionality

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.2.14

2017-02-03

- ▶ ASCIPDUM-749 Fixed known issue: IpduM_MainFunctionRx() and IpduM_MainFunctionTx() violate cycle time
- ▶ ASCIPDUM-754 Fixed known issue: Incorrect association of container PDU and contained PDUs
- ▶ ASCIPDUM-750 Fixed known issue: Dereferenced null pointer in IpduM_MainFunctionRx()
- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.2.13

2017-01-05

- ▶ ASCIPDUM-739 Fixed known issue: Out of bounds access during reception
- ▶ Added Support for queuing of container PDUs.

Module version 3.2.12

2016-11-04

- ▶ Corrected setting of transmission timer of container PDU

Module version 3.2.11

2016-09-23

- ▶ Incorporated Bugzilla RfC 71983: Introduce separate main functions for reception and transmission

Module version 3.2.10

2016-07-01

- ▶ Added *Multiple-PDU-to-Container* handling for Tx

Module version 3.2.9

2016-02-05

- ▶ Added support for Debug & Trace with custom header file configurable via parameter `BaseDbgHeaderFile`

Module version 3.2.8

2015-06-19

- ▶ Added *Multiple-PDU-to-Container* handling for Rx
- ▶ ASCIPDUM-586 Fixed known issue: The IpduM module reports an error for legal setting of `IpduMInitializationBySignalValue` and `IpduMEnableJitUpdate`

Module version 3.2.7

2015-01-07

- ▶ Removed AUTOSAR 3.x compliant symbolic name value macros and updated the logic to only provide AUTOSAR 4.0.2 compliant macros if macro `IPDUM_PROVIDE_LEGACY_SYMBOLIC_NAMES` is defined
- ▶ Changed signature of Com Rx callout `IpduM_ProcessRequestPdu` according to AUTOSAR bugzilla Rfc #52342

Module version 3.2.6

2014-10-02

- ▶ Implemented *Just-In-Time* update of parts
- ▶ Added initialization of multiplexed I-PDU with initialization data of dynamic and static part from COM module

Module version 3.2.5

2014-04-25

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.2.4

2013-10-11

- ▶ ASCIPDUM-452 Fixed known issue: EB-specific feature calls `Com_TriggerIPDUSend()` with wrong PDU-ID
- ▶ Changed the module structures for optimal memory usage
- ▶ Updated symbolic name value naming schema according to AUTOSAR 4.0 Rev 3

Module version 3.2.3

2013-06-14

- ▶ Added nonfunctional code improvements for shared data handling
- ▶ ASCIPDUM-425 Fixed known issue: If `IpduMTxConfirmationTimeout` is set to zero, confirmation calls to the PduR are blocked
- ▶ ASCIPDUM-429 Fixed known issue: The IpduM generates code even if the configured destination field does not fit in the I-PDU
- ▶ ASCIPDUM-412 Fixed known issue: The IpduM code generation fails if the IpduM configuration refers to a PduR destination PDU that has disabled confirmation PDU-ID and disabled IpduM Tx confirmation
- ▶ ASCIPDUM-437 Fixed known issue: The IpduM generates erroneous code if `IpduMDestinationBit` is not byte-aligned while byte copy is enabled or `IpduMDestinationBit` is not set to zero while zero copy is enabled

Module version 3.2.2

2013-02-07

- ▶ ASCIPDUM-342 Fixed known issue: The PduR IpduM transmission confirmation function is called with the wrong PDU-ID

Module version 3.2.1

2012-10-12

- ▶ Changed the top-level structure of the software-component description in the ARXML files from `/AUTOSAR/IpduM` to `/AUTOSAR_IpduM`
- ▶ Updated to AUTOSAR 4.0 Rev 3

Module version 3.2.0

2012-09-28

- ▶ ASCIPDUM-315 Fixed known issue: Transmission of incorrect data in case of zero size transmit queues
- ▶ Implemented AUTOSAR 4.0 ComStack Handle ID policy

Module version 3.1.2

2012-08-17

- ▶ Implemented definition of *Exclusive Area* in Basic Software Module Description

Module version 3.1.1

2012-06-22

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.1.0

2012-03-16

- ▶ Updated the include structure regarding the symbolic name value header files
- ▶ Updated `SchM_Enter/Exit()` calls to match AUTOSAR 4.0

Module version 3.0.4

2012-02-17

- ▶ Added BSWMD support

Module version 3.0.3

2012-01-20

- ▶ Improved speed of the template generator

Module version 3.0.2

2011-12-09

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 3.0.1

2011-09-30

- ▶ ASCIPDUM-210 Fixed known issue: The unattended wizard *Calculate Handle IDs* does not generate Handle IDs for the IpduM

Module version 3.0.0

2011-09-02

- ▶ Initial AUTOSAR 4.0 version

2.3.2.2. New features

- ▶ Added PbcfgM support

2.3.2.3. EB-specific enhancements

This chapter lists the enhancements provided by the module.

- ▶ Priority queuing for transmission of dynamic PDUs

Description:

The IpduM is able to provide a priority queue for each transmit path. `IpduMQueueSize` specifies the queue size. A value of 0 means not using a queue at all. `IpduMTxDynamicPriority` defines the priority of each PDU. 0 stands for the highest priority.

- ▶ Requesting service messages to request the transmission of a specific PDU from another ECU

Description:

To support a special type of multiplexed messages called requesting service messages, the functionality of the AUTOSAR IpduM is extended. A requesting service message is identified by a selector value set to 1. On reception of a requesting service message, the ECU sends out the requested Com I-PDU. The requested Com I-PDU is specified by a global PDU-ID in the data field of the dynamic part of the requesting service message.

- ▶ Automatic selector for *automatic setting of the selector value* by the IpduM

Description:

IpduM is extended to support *automatic setting of selector value*. The configuration parameter `IpduMTx-AutomaticSelector` is used to support this feature. If this parameter is enabled, the selector values for the transmit PDUs are set by the IpduM itself and if disabled, the selector value is not set by the IpduM. The selector value is also configurable using `IpduMTxSelectorValue`. During reception, the dynamic part is accepted and assembled only if the selector value is valid.

- ▶ Code and run-time optimizations

Description:

The code and run-time has been optimized. This optimization includes:

- ▶ Detection of development errors

Disabling this feature reduces the ROM consumption and reduces the execution time of the module code.

- ▶ Usage of static parts

Disabling this feature reduces the execution time and reduces the ROM consumption of the module code.

- ▶ Version information API

Disabling this API reduces the ROM consumption of the module code.

► Zero Copy

Enabling this feature reduces the execution time and reduces the ROM consumption of the module code.

► Byte-wise copy

Enabling this feature reduces the execution time and reduces the ROM consumption of the module code.

► Dynamic part queue

Disabling this feature reduces the execution time and reduces the ROM consumption of the module code.

► Automatic selector

Disabling this feature reduces the execution time and reduces the ROM consumption of the module code.

► Static memory allocation

Decreasing this parameter reduces the RAM consumption of the module configuration.

► Optional initialization of static and dynamic parts

Description:

For the enabled `IpduMInitializationBySignalValue`, the static and dynamic parts are initialized in retrieving signal values from the upper layer module by `IpduM_Init`. Otherwise the static and dynamic parts are only initialized by the unused area pattern configured.

Rationale:

The pre-compile switch was introduced to allow backward compatibility of the mandatory parameter `IpduMInitialDynamicPart`.

► Optional *Just-In-Time* update

Description:

For the enabled `IpduMEnableJitUpdate`, the *Just-In-Time* update functionality is provided in general. For the individual static and dynamic parts, the parameter `IpduMJitUpdate` has to be handled according to the SWS.

Rationale:

The pre-compile switch was introduced to allow backward compatibility of the mandatory parameter `IpduMInitialDynamicPart`.

- ▶ Possibility to select whether dequeuing happens in the context of `IpduM_MainFunctionTx()` or `IpduM_TxConfirmation()` by the configuration parameter `IpduMDequeueInTxConf`.
- ▶ Binary Search algorithm for matching contained PDU header ID

Description:

For matching contained PDUs header ID the `IpduM` module makes use of `Binary Search` algorithm in order to reduce runtime consumption. This is needed especially when a container PDU with `IpduMContainerRxAcceptContainedPdu` set to `IPDUM_ACCEPT_ALL` is received.

2.3.2.4. Deviations

This chapter lists the deviations of the module from the AUTOSAR standard.

- ▶ `IpduM` supports only little endian byte order for `IpduM` segments

Description:

For the configuration parameter `IpduMByteOrder` [`IPDUM162_Conf`] in the configuration container `IpduMRxIndication` and `IpduMTxRequest` only the value `LITTLE_ENDIAN` is allowed. This also violates [`IPDUM166`] which requests that `Com` and `IpduM` must have the same setting regarding the endianness.

Rationale:

The EB tresos Studio `Com` importer is able to convert `BIG_ENDIAN` segments to `LITTLE_ENDIAN` segments. For direct configuration and import from ECU configuration files this limitation has to be considered, but does not restrict any PDU layout.

Requirements:

`IPDUM166`, `IPDUM162_Conf`

- ▶ `IpduMRxDirectComInvocation` not supported

Description:

The configurable optimization *direct invocation of the COM module (bypassing the PduR)* as defined in `IPDUM140` is not implemented.

Rationale:

This optimization violates the AUTOSAR layered architecture.

Requirements:

`IPDUM140`, `IPDUM142_Conf`

- ▶ `PduR_IpduMRxIndication()`, `PduR_IpduMTransmit()`, `PduR_IpduMTriggerTransmit()`, and `PduR_IpduMTxConfirmation()` are mandatory

Description:

`PduR_IpduMRxIndication()`, `PduR_IpduMTransmit()`, `PduR_IpduMTriggerTransmit()`, and `PduR_IpduMTxConfirmation()` are mandatory and not optional interfaces as specified by requirement IPDUM105.

Rationale:

`PduR_IpduMRxIndication()`, `PduR_IpduMTransmit()`, `PduR_IpduMTriggerTransmit()`, and `PduR_IpduMTxConfirmation()` can only be optional when following optimizations are implemented:

- ▶ The IpduM Tx path can be disabled when PDUs are only received via the IpduM.
- ▶ The IpduM Rx path can be disabled when PDUs are only send via the IpduM.

These optimizations are not implemented by the IpduM.

Requirements:

IPDUM105, IPDUM104

- ▶ No AUTOSAR Debugging support

Description:

The requirements associated with AUTOSAR Debugging are not supported. This comprises all requirements mentioned within the section *Debugging*.

Rationale:

EB tresos Debug and Trace is intended to be used.

Requirements:

IPDUM144, IPDUM145, IPDUM146, IPDUM147

- ▶ No consistency check between code files and header files

Description:

The inter-module version checks as specified in the IpduM SWS are not implemented.

Rationale:

Module consistency check is not within the responsibility of the basic software, but part of the configuration management and delivery process.

Requirements:

IPDUM165, IPDUM170

- ▶ Support of configuration variant post-build (reference to product description: ASCPD-77)

Description:

The IpduM module only supports configuration variant post-build.

Requirements:

IPDUM095

- ▶ Usage of EB convention for file structure.

Description:

The IpduM module follows the EB-specific implementation method for file inclusion. Implementation is distributed over several implementation files.

Requirements:

IPDUM149, IPDUM150

- ▶ Configuration parameter `IpduMTxConfirmationPduId` is not OPTIONAL.

Description:

As per AUTOSAR_SWS_IPDUMultiplexer V4.0.3, `IpduMTxConfirmationPduId` can be optional. But it is also mentioned that the existence of this parameter is essential for the PduR generation tool to actually find a `symbolicNameValue` for the outgoing PDU. Also disabling the parameter demands lots of rework in the generator.

Requirements:

IPDUM158_Conf

- ▶ Non-compliant deviations in vendor-specific module definition file

Description:

The vendor-specific module definition file (VSMD) has non-compliant deviations to the AUTOSAR specification:

Violations against Rule EcucSws_1014: Additional vendor specific parameter definitions (using `ParameterTypes`), container definitions and references shall be added to the VSMD according to the alphabetical order.

- ▶ StMD-Node: `/AUTOSAR/IpduM`

- ▶ **StMD-Node:** /AUTOSAR/IpduM/IpduMConfig/IpduMRxPathway/IpduMRxIndication/IpduMRxDynamicPart/IpduMSegment
- ▶ **StMD-Node:** /AUTOSAR/IpduM/IpduMConfig/IpduMRxPathway/IpduMRxIndication/IpduMRxStaticPart/IpduMSegment
- ▶ **StMD-Node:** /AUTOSAR/IpduM/IpduMConfig/IpduMTxPathway/IpduMTxRequest
- ▶ **StMD-Node:** /AUTOSAR/IpduM/IpduMConfig/IpduMTxPathway/IpduMTxRequest/IpduMTxDynamicPart/IpduMSegment
- ▶ **StMD-Node:** /AUTOSAR/IpduM/IpduMConfig/IpduMTxPathway/IpduMTxRequest/IpduMTxStaticPart/IpduMSegment
- ▶ **StMD-Node:** /AUTOSAR/IpduM/IpduMGeneral

Rationale: Additional vendor specific parameter definitions have a specific order in the GUI which may differ to the alphabetical order.

- ▶ **Unsupported parameter IpduMConfigurationTimeBase**

Description:

This configuration parameter is replaced by the 2 configurable parameters IpduMRxTimeBase and IpduMTxTimeBase, and IpduMDedicatedIpduProcessingSupport is enabled, additionally by IpduMTimeBase.

See Bugzilla RfC 71983 for further information.

See ASCIPDUM-772.

Requirements:

IPDUM131_Conf

- ▶ **Unsupported API IpduM_MainFunction**

Description:

The IpduM_MainFunction is split into IpduM_MainFunctionRx and IpduM_MainFunctionTx.

See Bugzilla RfC 71983 for further information.

Requirements:

IPDUM103, IPDUM101

- ▶ **IpduMDequeueInTxConf selects when dequeuing happens**

Description:

If IpduMDequeueInTxConf is **FALSE**, dequeuing happens only in IpduM_MainFunctionTx().

Requirements:

IpduM.ASR42.SWS_IpduM_00190

- ▶ IpduMDequeueInTxConf selects when dequeuing happens

Description:

If IpduMDequeueInTxConf is **TRUE**, dequeuing happens also in IpduM_TxConfirmation().

Requirements:

IpduM.ASR42.SWS_IpduM_00190

2.3.2.5. Limitations

This chapter lists the limitations of the module. Refer to the module references chapter *Integration notes*, subsection *Integration requirements* for requirements on integrating this module.

- ▶ IpduM Handle ID assignment

Description:

- ▶ IpduMConfig/IpduMRxPathway/*/IpduMRxIndication/IpduMRxHandleId has to be assigned zero-based and dense.
- ▶ IpduMConfig/IpduMTxPathway/*/IpduMTxRequest/IpduMTxStaticPart/IpduMTxStaticHandleId has to be assigned zero-based and dense.
- ▶ IpduMConfig/IpduMTxPathway/*/IpduMTxRequest/IpduMTxDynamicPart/*/IpduMTxDynamicHandleId has to be assigned dense and start with `max(IpduMTxStaticHandleId)+1`.
- ▶ IpduMConfig/IpduMContainedTxPdu/*/IpduMContainedTxPduHandleId has to be assigned dense and start from the last IpduMTxPathway/IpduMTxRequest/IpduMTxDynamicPart/IpduMTxDynamicHandleId.
- ▶ IpduMConfig/IpduMContainerTxPdu/*/IpduMContainerTxHandleId has to be assigned dense and start from the number of IpduMTxPathways.
- ▶ IpduMConfig/IpduMContainerRxPdu/*/IpduMContainerRxHandleId has to be assigned dense and start from the last IpduMRxPathway/IpduMRxHandleId.

Rationale:

Code size reduction and run-time improvement.

- ▶ IpduM module expects restricted multiplicity of container PduRRoutingTable

Description:

The IpduM module expects the upper multiplicity of container `PduRRoutingTable` restricted to 1.

Rationale:

The EB tresos AutoCore does not support the post-build time selectable concept, stated within AutoCore_Generic_documentation.pdf in section 'Post-build time selectable concept'. Instead, the post-build time loadable concept is supported where multiple routing tables are not applied.

► Configuration separation

Description:

In the current version the configuration is generated in a monolithic manner, not separately for each core.

► Uniqueness of contained PDU header IDs

Description:

The `IpduMContainedPduHeaderIds` must be unique for contained Rx PDUs, respectively for contained Tx PDUs.

► Rx/Tx PathWay PduLengthType

Description:

The `PduLengthType` is limited to `uint16` for Rx and Tx PathWays. Even though the `PduLengthType` can be set to `uint32` the `IpduM` module does not offer support for Rx and Tx PathWay PDUs of size greater than `uint16`.

2.3.3. PduR module release notes

- AUTOSAR R4.0 Rev 3
- AUTOSAR SWS document version: 3.2.0
- Module version: 5.3.30.B206164
- Supplier: Elektrobit Automotive GmbH

2.3.3.1. Change log

This chapter lists the changes between different versions.

Module version 5.3.30

2018-06-22

- ▶ Allow initial usage of `PduLengthType` set to `uint32`
- ▶ ASCPDUR-2476 Fixed known issue: Duplicated `SessionQueueMemRef` entries
- ▶ ASCPDUR-2482 Fixed known issue: Misaligned `SessionQueueMemRef`

Module version 5.3.29

2018-04-20

- ▶ ASCPDUR-2420 Fixed known issue: Generation aborts without error message for incompletely configured routing path

Module version 5.3.28

2018-02-16

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 5.3.27

2017-09-22

- ▶ Added support for call of `Up_TxConfirmation` in a multicast transmission
- ▶ Switch from MISRA-C:2004 to MISRA-C:2012

Module version 5.3.26

2017-08-25

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 5.3.25

2017-07-28

- ▶ ASCPDUR-2283 Fixed known issue: `PduR` generates wrong buffer assignments for If-gateway relations

Module version 5.3.24

2017-06-30

- ▶ Abolished memory size limitation of 64 KiB

Module version 5.3.23

2017-05-05

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 5.3.22

2017-03-31

- ▶ Added support of N:1 PDU routing

Module version 5.3.21

2016-07-01

- ▶ ASCPDUR-2165 Fixed known issue: Wrong buffer size request for ASR 3.2 upper layer TP module

Module version 5.3.20

2016-05-25

- ▶ Avoid compiler warning similar to dead assignment to "AvailableBufferSize" eliminated

Module version 5.3.19

2016-04-29

- ▶ Added handling for AUTOSAR 3.2 upper layer TP module in dependence on existence of BUFREQ_E_-BUSY on Rx side

Module version 5.3.18

2016-04-01

- ▶ ASCPDUR-2154 Fixed known issue: Open section PDUR_START_SEC_CODE within an already opened section for ASR 3.2 wrapper

Module version 5.3.17

2015-11-06

- ▶ Create/extend recommended configurations for Ethernet

Module version 5.3.16

2015-06-19

- ▶ Internal module improvement. This module version update does not affect module functionality

Module version 5.3.15

2015-02-20

- ▶ Implemented non-functional code improvements to fix Misra violation

Module version 5.3.14

2015-01-07

- ▶ Added support for configurable mapping of `PduR_IsValidConfig` function to dedicate memory section
- ▶ Removed AUTOSAR 3.x compliant symbolic name value macros and updated the logic to only provide AUTOSAR 4.0.2 compliant macros if macro `PDUR_PROVIDE_LEGACY_SYMBOLIC_NAMES` is defined

Module version 5.3.13

2014-10-02

- ▶ ASCPDUR-2077 Fixed known issue: Missing memory section for external declared AUTOSAR 3.2 wrapper functions

Module version 5.3.12

2014-04-25

- ▶ Improved robustness of multicast transmission for `SduLength 0`
- ▶ ASCPDUR-2036 Fixed known issue: Handle ID wizard fails when `PduRZeroCostOperation` is enabled
- ▶ Replaced enumeration type of `PduR_StateType` in defining literals according to company guidelines
- ▶ ASCPDUR-2045 Fixed known issue: XDM checks to prevent incorrect configuration are missing
- ▶ Implemented call of `LoTp_Transmit` by successful `PduR_LoTpRxIndication` if TP threshold was not reached for routing *on the fly*
- ▶ ASCPDUR-2063 Fixed known issue: Build error due to missing file `PduR_PBcfg.c` if code generation for PduR is disabled and only post-build configuration is compiled
- ▶ Improved robustness of retry mechanism for TP multicast gateway
- ▶ ASCPDUR-2075 Fixed known issue: Nested `MemMap` section if `TS_MERGED_COMPILE` is activated

Module version 5.3.11

2013-10-11

- ▶ ASCPDUR-1984 Fixed known issue: Resources are not released if `PduR_LoTpRxIndication` follows `PduR_LoTpTxConfirmation` for routing *on the fly*
- ▶ Improved nonfunctional code of TP gateway regarding shared data handling
- ▶ ASCPDUR-2003 Fixed known issue: Retry fails for a TP gateway to multiple destinations
- ▶ ASCPDUR-2006 Fixed known issue: TP buffer smaller than `TpThreshold` is assigned by an unknown message length request for routing *on the fly*

Module version 5.3.10

2013-09-13

- ▶ Updated descriptions in release notes

Module version 5.3.9

2013-08-16

- ▶ ASCPDUR-1890 Fixed known issue: Invalid pointer in interface gateway causes incorrect runtime behavior even if development error detection is enabled
- ▶ ASCPDUR-1913 Fixed known issue: `PduR_LoTpTxConfirmation` and `PduR_LoTpCopyTxData` call wrong PDU-ID for multicast transmission
- ▶ Implemented `0xFFFF` as invalid return value of `PduR_GetConfigurationId()` in case of an error. Therefore this value has been excluded from the valid ranges of configuration IDs in the configuration
- ▶ Improved handling of disabled `TxPdIds` of upper layer modules
- ▶ Implemented generation of symbolic name values
- ▶ Improved nonfunctional code of AUTOSAR 4.0 to 3.2 TP API Wrapper
- ▶ Improved nonfunctional code of If gateway regarding shared data handling
- ▶ Added Debug & Trace code instrumentation
- ▶ Improved nonfunctional code of TP multicast regarding shared data handling
- ▶ ASCPDUR-1943 Fixed known issue: Compilation error for If gateway when all lower layer modules are optimized for static payload

Module version 5.3.8

2013-06-14

- ▶ ASCPDUR-1852 Fixed known issue: Calling `PduR_LoTriggerTransmit()` does not pass the initial default values to the lower layer module
- ▶ ASCPDUR-1858 Fixed known issue: `PduR_LoTriggerTransmit()` uses the wrong PDU for multicast transmission

Module version 5.3.7

2013-05-10

- ▶ Added checks of published information signature to prevent loading of incompatible post-build configuration
- ▶ Improved error message in case Handle IDs are not set properly

Module version 5.3.6

2013-04-12

- ▶ ASCPDUR-1785 Fixed known issue: Incorrect data will be copied when `PduR_LoTpCopyTxData()` is called via direct gateway
- ▶ ASCPDUR-1783 Fixed known issue: `PduR_LoTpCopyRxData()` returns an incorrect buffer size when requested from AUTOSAR 3.2 upper layer
- ▶ ASCPDUR-1805 Fixed known issue: Handle ID calculation fails if `PduRZeroCostOperation` is enabled

Module version 5.3.5

2013-02-08

- ▶ Updated the Basic Software Module Description for memory mapping macros
- ▶ Removed `BUFREQ_E_BUSY` from `CopyRxData`
- ▶ Added wrapper to AUTOSAR 3.2 upper layers with TP interface
- ▶ ASCPDUR-1754 Fixed known issue: A segmentation fault is reported for multicast transmission to lower layer modules with unsupported retry mechanism
- ▶ ASCPDUR-1753 Fixed known issue: Possible misalignment in PduR RAM may cause an exception on some CPUs

Module version 5.3.4

2013-01-11

- ▶ ASCPDUR-1714 Fixed known issue: The containers `PduRTpBuffer` and `PduRTxBuffer` can be changed at post-build time even though this is not supported

- ▶ ASCPDUR-1677 Fixed known issue: Invalid reference to SCHM_PDUR_EXCLUSIVE_AREA_0 in PduR_Bswmd.arxml
- ▶ ASCPDUR-1722 Fixed known issue: PduR_LoTpCopyTxData() returns wrong value in a multicast transmission if the available data size is requested with SduLength 0

Module version 5.3.3

2012-12-14

- ▶ ASCPDUR-1510 Fixed known issue: A compilation error may occur when zero cost optimization, i.e. the parameter PduRZeroCostOperation is enabled

Module version 5.3.2

2012-11-14

- ▶ ASCPDUR-1645 Fixed known issue: Upper layer APIs are called with wrong TxPduld
- ▶ Implemented AUTOSAR 4.0 TP gateway and multicast functionality

Module version 5.3.1

2012-10-12

- ▶ Changed the top-level structure of the software-component description in the arxml files from /AUTOSAR/PduR to /AUTOSAR_PduR

Module version 5.3.0

2012-09-14

- ▶ Updated to AUTOSAR 4.0 ComStack Handle ID policy
- ▶ Updated CancelReceive, CancelTransmit and ChangeParameter functionality to Rev 3

Module version 5.2.0

2012-08-17

- ▶ Implemented AUTOSAR 4.0 TP APIs for single cast
- ▶ Removed Dem handling according to AUTOSAR 4.0 Rev 3

Module version 5.1.2

2012-06-15

- ▶ ASCPDUR-1520 Fixed known issue: Generated Basic Software Module Description (BSWMD) of PduR is invalid
- ▶ Implemented support for the post-build configuration manager

Module version 5.1.1

2012-04-20

- ▶ ASCPDUR-1495 Fixed known issue: Double inclusion of memory section `PDUR_START_SEC_CODE/`
`PDUR_STOP_SEC_CODE` may lead to build or runtime errors

Module version 5.1.0

2012-03-16

- ▶ Added generation of BSWMD
- ▶ Modified `SchM_Enter/Exit()` calls to match AUTOSAR 4.0
- ▶ Updated naming scheme of symbolic name values to AUTOSAR 4.0 Rev 3
- ▶ Implemented DET call to uninitialized `PduR_GetConfigurationId`

Module version 5.0.4

2012-02-17

- ▶ ASCPDUR-1455 Fixed known issue: If multicast is configured to only one lower layer communication interface module, a compilation error is reported

Module version 5.0.3

2012-01-20

- ▶ ASCPDUR-1435 Fixed known issue: Corrupted I-PDU may be copied to destination by `TriggerTransmit`
- ▶ Implemented zero cost operation
- ▶ ASCPDUR-1443 Fixed known issue: The PduR will not compile if an upper layer module does not provide a `TxConfirmation` function
- ▶ Defined APIs for interaction with upper or lower layer modules are defined in single file (`PduR_AdjLayerApi.c`)
- ▶ Implemented separate enabling/disabling of Handle ID calculation for each adjacent module

Module version 5.0.2

2011-12-09

- ▶ ASCPDUR-1394 Fixed known issue: If multicast is configured with only one lower layer transport protocol module, a compilation error is reported
- ▶ ASCPDUR-1423 Fixed known issue: An error is reported if LinTp functions `CancelTransmit()`, `CancelReceive()` and `ChangeParameter()` are disabled

Module version 5.0.1

2011-09-30

- ▶ ASCPDUR-1351 Fixed known issue: When gatewaying non TP-PDUs, the `SduLength` is not updated in the `PduInfoPtr` parameter passed to `PduR_<Lo>TriggerTransmit()`
- ▶ ASCPDUR-1356 Fixed known issue: `ChangeParameter` functionality for Dcm is not supported but is enabled in recommended configuration for PduR
- ▶ ASCPDUR-1354 Fixed known issue: LoTp module does not compile due to missing type definition
- ▶ ASCPDUR-1357 Fixed known issue: Auxiliary function `PduR_DtctRxPathOvrAllLoTpConfigs` is not linkable
- ▶ ASCPDUR-1220 Fixed known issue: If `Com.h` and `PduR.h` are included in the same translation unit, a linker error may occur
- ▶ ASCPDUR-1359 Fixed known issue: PduR incorrectly routes I-PDUs that are not configured in PduR to upper layer modules

Module version 5.0.0

2011-09-02

- ▶ Initial AUTOSAR 4.0 version

2.3.3.2. New features

- ▶ Allow initial usage of `PduLengthType` set to `uint32`.

2.3.3.3. EB-specific enhancements

This chapter lists the enhancements provided by the module.

- ▶ Optimization Impact of pre-compile time switches in parameter description of XDM

Description:

The impact of enabling/disabling of any pre-compile time switch on resource consumption with respect to RAM Usage (configuration), ROM Usage (configuration), RAM Usage (static code), ROM Usage (static code) and Execution time is stated.

- ▶ Connection to AUTOSAR 3.2 upper layer modules with TP interface

Description:

Routing to an AUTOSAR 3.2 upper layer module with TP interface can be configured the same way as for an AUTOSAR 4.0 module. The configuration parameters `PduRMaxRxPduId` and `PduRMaxTxPduId` have to be set in the `PduRBswModules` container of the `PduR` module appropriately. To avoid wasting RAM, the Handle IDs within the upper layer module should be zero-based and dense.

- ▶ Optimization for gateway I-PDUs of static size

Description:

If all I-PDUs of a lower layer communication interface module that are routed via a Tx-buffer to a lower layer module are of fixed size, the `PduR` module allows optimization in enabling parameter `PduRStaticPduLengthSupport` of the associated source module of the `PduRBswModules` container.

2.3.3.4. Deviations

This chapter lists the deviations of the module from the AUTOSAR standard.

- ▶ Minimum routing capability

Description:

The minimum routing capability is not supported (reference to product description: ASCPD-14)

This comprises also the following:

- ▶ The undefined enumeration value `PDUR_REDUCED` for the `PduR_StateType` (PDUR0742).
- ▶ The unused production error code `PDUR_E_INIT_FAILED` (PDUR100).

Additionally, the restrictively specified configuration class `PreCompile` of parameter `PduRIsMinimumRouting` violates the post-build loadable concept by rule EB08 for the container list `PduRRoutingTables` and `PduRRoutingTable`.

Requirements:

PDUR285, PDUR286, PDUR324, PDUR327, PDUR329, PDUR100, PDUR221, PDUR0742, PDUR106, PDUR306_Conf

- ▶ Symbolic source module PDU IDs are generated to `PduR_SymbolicNames_PBcfg.h`

Description:

The requirement [ecuc_sws_2108] states that symbolic source module PDU IDs should be generated to the PduR module's header file. An example below the requirement shows that these IDs should be generated to `<Module>_cfg.h`. The EB module configuration generator does not generate the source module PDU IDs into the file `PduR_Cfg.h`. Instead of this, the source module PDU IDs are generated to the file `PduR_SymbolicNames_PBcfg.h`. Furthermore the file `PduR_SymbolicNames_PBcfg.h` is included by the header file `PduR_PBcfg.h`.

Rationale:

- ▶ The generated symbolic source module PDU IDs are not used within the EB AUTOSAR modules
- ▶ Changes in the symbolic source module PDU IDs cause other modules which include header files from the module PduR to recompile unnecessarily.

Requirements:

ecuc_sws_2108

- ▶ `Lo_Transmit` called by `PduR_LoTxConfirmation` for triggered data provision of non-empty FIFO

Description:

The `Lo_Transmit` is called by `PduR_LoTxConfirmation` instead of `PduR_LoTriggerTransmit` in case of a gateway to a communication interface module with triggered data provision when the FIFO buffer is non-empty.

Requirements:

PDUR640, PDUR0666 (second part of the requirement)

Rationale:

The gateway to a communication interface module is provided the way of AUTOSAR 3.x.

- ▶ Transmit cancellation from upper communication interface module not supported (reference to product description: ASCPD-24)

Description:

The PduR deviates from the description in AUTOSAR R4.0 Rev 2 SWS section 7.2. *Cancel transmission* in that it allows no transmit cancellation for communication interface modules.

Requirements:

PDUR0721, PDUR0723, PDUR0700, PDUR0701, PDUR424, PDUR0769

- ▶ Unsupported cancel transmit functionality for multicasted SF-TP PDUs

Description:

`PduR_<Up>CancelTransmit` function does not support multicast SF-TP PDUs.

Requirements:

PDUR0724, PDUR0701, PDUR0729, PDUR0730

- File structure differs to AUTOSAR specification

Description:

The code file structure differs by requirement PDUR226 since a different strategy is applied where `PduR_`
`Cfg.c` is not needed but `PduR_Lcfg.c`.

The header file structure differs by requirement PDUR132 in the following:

- `PduR.h` does not include `PduR_Lcfg.h`.
- `PduR.h` does not include directly `PduR_Types.h` and `PduR_Cfg.h`.
- `PduR_Cfg.h` does not include the `<module>_PduR.h` and `<module>_CbK.h`.
- `Det.h` is not included directly to the implementation.

Requirements:

PDUR226, PDUR132

- Usage of AUTOSAR 3.1 reentrancy

Description:

Instead of the new reentrancy concept for APIs introduced in AUTOSAR 4.0, the AUTOSAR 3.1 reentrancy concept is used.

Rationale:

The reentrancy concept of the PduR in AUTOSAR 4.0.2/4.0.3 shows several critical lacks and is removed in AUTOSAR 4.1.1. See http://www.autosar.org/bugzilla/show_bug.cgi?id=51758.

Requirements:

PDUR630, PDUR622, PDUR624

- Prioritization of multicast destination I-PDUs

Description:

The prioritization of multicast destination I-PDUs is not handled by the generator. This comprises the requirements:

- ▶ PDUR635: If there is a multicast (1:n, n>1) transmission, the PDU Router module shall call them in routing path ID order with the lowest ID first.
- ▶ PDUR618: In case an I-PDU is multicasted or gatewayed to more than one destination, the configuration parameter routing path's Handle ID denotes the order in which the destinations are served. The lowest HandleID is served first.

Rationale:

- ▶ PDUR618: The requirement is removed with Bugzilla entry http://www.autosar.org/bugzilla/show_bug.cgi?id=52882.

Requirements:

PDUR635, PDUR618

- ▶ Unsupported mode-dependent routing (reference to product description: ASCPD-101)

Description:

The enabling/disabling of routing path groups (e.g. functional requirement BSW06120 *mode-dependent control of PDU-gateway*) is not supported.

Requirements:

PDUR0726, PDUR0710, PDUR0714, PDUR0717, PDUR0715, PDUR646, PDUR663, PDUR100, PDUR654, PDUR0709, PDUR615 PDUR647, PDUR648, PDUR617, PDUR0716, PDUR649, PDUR329_Conf, PDUR309_Conf, PDUR301_Conf

- ▶ Unsupported error codes

Description:

The following error codes listed in requirement PDUR100 are not supported due to the reason given below:

- ▶ PDUR_E_PARAM_INVALID: No need to apply this error code. See Bugzilla entry http://www.autosar.org/bugzilla/show_bug.cgi?id=51765.
- ▶ PDUR_E_DUPLICATE_IPDU_ID: Incomplete, ambiguous and unclear definition of requirement PDUR622 and reentrancy at all. See Bugzilla entry http://www.autosar.org/bugzilla/show_bug.cgi?id=51758.
- ▶ PDUR_E_IPDU_TOO_LONG: No need to apply this error code. See Bugzilla entry http://www.autosar.org/bugzilla/show_bug.cgi?id=51765.
- ▶ PDUR_E_ROUTING_PATH_GROUP_INVALID: The feature mode-dependent routing is not supported.
- ▶ PDUR_E_INIT_FAILED: The feature minimum routing is not supported.

Requirements:

PDUR100, PDUR624

- Distinguish name of `PduR_<Up>Transmit` when `Up` allows both API archetypes

Description:

The function `PduR_<User:Up>Transmit` shall be defined with a different API name for modules that support communication interface as well as transport protocol I-PDUs. For example `PduR_ComTransmit` and `PduR_ComTpTransmit`.

Requirements:

PDUR406

- Imprecise description of requirement PDUR662

Description:

Requirement PDUR662: If the destination communication interface module is requesting the I-PDU buffer that uses `PduR_<DstLo>TriggerTransmit` and the buffer is not available the return value `E_NOT_OK` shall be used.

Rationale:

Discussions on that issue, especially what is meant by *buffer is not available*, are still ongoing. See Bugzilla entry http://www.autosar.org/bugzilla/show_bug.cgi?id=52286 or http://www.autosar.org/bugzilla/show_bug.cgi?id=59825.

EB-interpretation with respect to the considered RfC: For a destination communication interface module requesting buffer through a call of `PduR_<DstLo>TriggerTransmit` for an I-PDU which has `PduRTxBufferDepth > 1`, the first buffer initialized by the default values with `PduR_Init()` is returned with `E_OK`. The same applies to an I-PDU which has `PduRTxBufferDepth = 1` configured.

Requirements:

PDUR662

- Deviation of service IDs

Description:

The specification of service IDs has been changed several times. The specification according to AUTOSAR 3.1 is used instead.

Requirements:

PDUR0780, PDUR0781, PDUR0782

- Changed signature of `PduR_<User:Up>Transmit`

Description:

By requirement PDUR406, the function `PduR_<User:Up>Transmit` is defined as `Std_ReturnType PduR_<User:Up>Transmit(PduIdType id, PduInfoType* info)`. In contrast to this, the EB-implementation defines the function as `Std_ReturnType PduR_<User:Up>Transmit(PduIdType id, const PduInfoType* info)`.

Rationale:

Usage of the qualifier `const` for `PduInfoType`: All the adjacent modules apply this qualifier for `PduInfoType` and also the `GenericComServices_Transmit`. See http://www.autosar.org/bugzilla/show_bug.cgi?id=51374).

Requirements:

PDUR406

- Changed signature of `PduR_<User:LoTp>CopyRxData`

Description:

By requirement PDUR512, the function `PduR_<User:LoTp>CopyRxData` is defined as `BufReq_ReturnType PduR_<User:LoTp>CopyRxData(PduIdType id, PduInfoType* info, PduLengthType* bufferSizePtr)`. In contrast to this, the EB-implementation defines the function as `BufReq_ReturnType PduR_<User:LoTp>CopyRxData(PduIdType id, const PduInfoType* info, PduLengthType* bufferSizePtr)`.

Rationale:

Due to the generic usage of `GenericComServices_CopyRxData`, the `CopyRxData` APIs shall be unified as described by Bugzilla entry http://www.autosar.org/bugzilla/show_bug.cgi?id=56021 in providing the qualifier `const` to the type `PduInfoType*`.

Requirements:

PDUR512

- Unsupported TP (multicast) gateway requirements

Description:

The following requirements associated with the TP (multicast) gateway are not supported:

Rationale:

- PDUR624: It is assumed that the requirement is meant for single cast. The requirement is removed by AUTOSAR Bugzilla RFC: http://www.autosar.org/bugzilla/show_bug.cgi?id=55453.

PDUR0779: The requirement is removed with AUTOSAR 4.1 by AUTOSAR Bugzilla RfC [http://www.-autosar.org/bugzilla/show_bug.cgi?id=51765](http://www.autosar.org/bugzilla/show_bug.cgi?id=51765). The considered error could only happen if the TP is wrongly implemented.

Requirements:

PDUR624, PDUR0779

- ▶ No AUTOSAR Debugging support

Description:

The requirements associated with AUTOSAR Debugging are not supported. This comprises all requirements mentioned within the section *Debugging*.

Rationale:

EB tresos Debug & Trace is intended to be used.

Requirements:

PDUR487, PDUR488, PDUR489, PDUR490

- ▶ No consistency check between code files and header files

Description:

According to the PduR SWS, the PduR module shall perform inter-module version checks.

Rationale:

Module consistency check is not within the responsibility of the basic software but part of configuration management and delivery process.

Requirements:

PDUR0774

- ▶ Unsupported AUTOSAR configuration parameter `PduRRetransmission`

Description:

Optimizing the static BSW with respect to retransmission of transport protocol modules as part of a direct-gateway is not provided.

Requirement:

PDUR332_Conf

- ▶ Unsupported AUTOSAR configuration parameter `PduRUseTag`

Description:

The AUTOSAR configuration parameter `PduRUseTag` is not supported.

Rationale:

A different strategy is applied in using property files for all adjacent modules. This allows to overrule the generated API names without usage of `PduRUseTag`.

Requirement:

PDUR319_Conf

- ▶ Unsupported AUTOSAR configuration parameter `PduRBswModuleRef`

Description:

The AUTOSAR configuration parameter `PduRBswModuleRef` is not supported.

Rationale:

A different strategy is applied in using property files for all adjacent modules. The information necessary is provided therein, e.g. API names of a CDD.

Requirement:

PDUR294_Conf, PDUR504

- ▶ Unsupported AUTOSAR configuration parameter `PduRMaxTxBufferNumber`

Description:

The AUTOSAR configuration parameter `PduRMaxTxBufferNumber` is not supported.

Rationale:

There is no stringent need for the AUTOSAR parameter `PduRMaxTxBufferNumber`, see AUTOSAR Bugzilla RFC: http://www.autosar.org/bugzilla/show_bug.cgi?id=59190.

Requirement:

PDUR331_Conf

- ▶ Unsupported AUTOSAR configuration parameter `PduRMaxTpBufferNumber`

Description:

The AUTOSAR configuration parameter `PduRMaxTpBufferNumber` is not supported.

Rationale:

There is no stringent need for the AUTOSAR parameter `PduRMaxTpBufferNumber`, see AUTOSAR Bugzilla RfC: http://www.autosar.org/bugzilla/show_bug.cgi?id=59190.

Requirement:

PDUR330_Conf

- Support of configuration variant post-build (reference to product description: ASCPD-77)

Description:

The PduR module supports only configuration variant post-build. This is handled in restricting the range of configuration parameter `IMPLEMENTATION_CONFIG_VARIANT` to `VariantPostBuild` leading to violation of rule `EcucSws_6051`.

Requirements:

PDUR425, PDUR287, PDUR619

- Allow configuration class `PostBuild` to some AUTOSAR parameters of class `PreCompile`

Description:

Some AUTOSAR configuration parameters are restricted to configuration class `PreCompile` for all variants. The following parameters are provided with added configuration class `PostBuild` and thus violate rule `EcucSws_6051` due to the following reason:

- `PduRConfigurationId`: Allow post-build loadable concept as stated by AUTOSAR Bugzilla RfC 53197, see http://www.autosar.org/bugzilla/show_bug.cgi?id=53197.
- `PduRIIsEnabledAtInit`: Allow post-build loadable concept as stated by AUTOSAR Bugzilla RfC 53197, see http://www.autosar.org/bugzilla/show_bug.cgi?id=53197.
- `PduRTpThreshold`: Allow post-build loadable concept as stated by AUTOSAR Bugzilla RfC 53197, see http://www.autosar.org/bugzilla/show_bug.cgi?id=53197.
- `PduRTransmissionConfirmation`: Wrongly classified as stated by AUTOSAR Bugzilla RfC 58298, see http://www.autosar.org/bugzilla/show_bug.cgi?id=58298.

Requirements:

PDUR327_Conf, PDUR329_Conf, PDUR320_Conf

- Software and specification version number solely defined within `PduR_Version.h.m4`

Description:

Requirement PDUR0762 is specified in providing a software and specification version number to all PDU Router header files, not a single one.

Rationale:

A different strategy is applied defining the software and specification number within single header file `PduR_Version.h.m4`. This comprises EB coding guidelines BSWM_GEN_007 and BSWM_GEN_009.

Requirements:

PDUR0762

2.3.3.5. Limitations

This chapter lists the limitations of the module. Refer to the module references chapter *Integration notes*, subsection *Integration requirements* for requirements on integrating this module.

- Restricted number of destination modules for TP multicast transmission

Description:

The TP multicast transmission (1:n, n>1) of an I-PDU from a local module to *n* transport protocol modules is restricted to $n < 16$ destinations.

Rationale:

This reduces the consumption of RAM.

Requirements:

PDUR634

- Restricted multiplicity of container `PduRRoutingTable`

Description:

The upper multiplicity of container `PduRRoutingTable` is restricted to 1.

Rationale:

The EB tresos AutoCore does not support the post-build time selectable concept, stated within AutoCore_Generic_documentation.pdf in section 'Post-build time selectable concept'. Instead, the post-build time loadable concept is supported where multiple routing tables are not applied.

- Restricted support of PDUs with more than 64 KiB of user data

Description:

With EcuC parameter `PduLengthTypeEnum` configured to `UINT32`, the PduR module is allowed to handle PDUs with more than 64 KiB of user data.

For TP-PDUs, this is restricted to

- ▶ singlecast transmission and
- ▶ singlecast reception.

NonTp-PDUs are not allowed to handle more than 64 KiB of user data at all.

Rationale:

In order to support the SomelpTp module, which is designed to handle TP-PDUs greater than 64 KiB, only the use cases singlecast transmission and singlecast reception are required.

TP gateway as well as TP multicast is not a use case in combination with the SomelpTp module.

For nonTP-PDUs, the `PduRPduMaxLength` is limited to 255 by AUTOSAR.

3. ACG8 COM Services user's guide

3.1. Overview

The ACG8 COM Services user's guide provides information about the concepts of network-independent communication services in the AUTOSAR context. Before you read this user's guide, read the general concepts about communication stacks in AUTOSAR that are described in the EB tresos AutoCore Generic documentation.

- ▶ [Section 3.2, “Background information”](#) describes the concept of the network-independent communication in the AUTOSAR context.

3.2. Background information

This chapter provides general information about network-independent communication concepts in the AUTOSAR context. If you are not familiar with the general concepts of communication in AUTOSAR, read the general information provided in the EB tresos AutoCore Generic documentation first.

3.2.1. Network-independent and network-dependent communication in AUTOSAR

The AUTOSAR communication stack can be divided into a *network-independent* and a *network-dependent* part.

The following sections provide further details about network-independent communication:

- ▶ [Section 3.2.1.1, “Modules and dependencies of the network-independent communication stack”](#) describes the modules and dependencies of the network-independent communication stack.
- ▶ [Section 3.2.1.2, “Data transmission in the network-independent communication stack”](#) describes the data transmission in the network-independent communication stack.

You find information about the network-dependent communication in the user's guides in the network-dependent product documentation.

3.2.1.1. Modules and dependencies of the network-independent communication stack

This chapter describes the modules of the network-independent part of the communication stack together with their dependencies. Since the modules of the network-independent part of the communication stack do not depend on the communication protocols, only generic versions of these modules are present in EB tresos AutoCore. The following text gives you a short overview of these modules.

PDU Router (`PduR`):

The PDU Router module provides two major services:

1. It *dispatches PDUs received via the underlying modules* (i.e., Interface and Transport Layer modules) to the higher layers (`Com`, `Dcm`) and vice versa.
2. The PDU Router *performs gateway functionalities* between communication networks by forwarding PDUs from one Interface to another of either the same (e.g., FlexRay to FlexRay) or of a different type (e.g., CAN to FlexRay). Routing decisions in the PDU Router are based on a static routing table and on the identifiers of the PDUs.

PDU Multiplexer (`IpduM`)

The PDU Multiplexer module takes care of *multiplexing parts of a PDU*. Hereby, the value of a dedicated part of the PDU (the *multiplexer switch*) is used to define the semantic content of the remainder of the PDU. This works just like the tag element in a variant record or a union in programming languages. In the reception case, multiplexed PDUs are forwarded from the `PduR` to the `IpduM` for demultiplexing. Once demultiplexed, the `IpduM` hands the PDUs back to the `PduR`. In the sending case, the `PduR` obtains a PDU from `Com` and hands this PDU to the `IpduM` for multiplexing. The `IpduM` returns the multiplexed PDU to the `PduR`, which routes the multiplexed PDU to its final destination.

Note that the multiplexing features of the `IpduM` are limited to `Com` I-PDUs (i.e., I-PDUs sent or received by `Com`).

Communication (`Com`):

The `Com` module provides *signal-based inter-ECU communication* to the upper layer (`Rte`). On the sender side `Com` packs multiple signals into a PDU and forwards this PDU to the `PduR` in order to issue the PDU's transmission via the respective Interface module. On the receiver side, the `Com` module obtains a PDU from the PDU router, extracts the signals contained in the PDU, and forwards the extracted signals to the upper software layer (`Rte`).

Diagnostic Communication Manager (`Dcm`):

The Diagnostic Communication Manager module is a sub-module of the AUTOSAR diagnostic stack. The `Dcm` module provides *services which allow a tester device to control diagnostic functions* in an ECU via the communication network (i.e., CAN, LIN, FlexRay). Hereby the `Dcm` supports the diagnostic protocols OBD [3] and UDS [2].

3.2.1.2. Data transmission in the network-independent communication stack

This chapter enables you to understand the data transmission of the network-independent communication stack. Specifically the topics `signals` and `signal groups`, `transmission modes` and `I-PDU groups` are addressed.

3.2.1.2.1. Signals and signal groups

This chapter provides information about signals and signal groups exchanged by the `Com` module. At the API provided by the `Com` module, sender and receiver exchange signals as basic communication objects. Supported data types of these signals are *primitive* data types (e.g., `int`, `char`, etc.) as well as *opaque* data types (transmitted as an array of bytes). The `Com` API functions `Com_SendSignal()` and `Com_ReceiveSignal()` enable the transmission and reception of signals.

In order to simplify the *atomic* transmission and reception of structured data types (i.e., `structs` of the C programming language), `Com` provides facilities to group multiple signals into a *signal group*. The value of a signal, which is part of a signal group, can be updated via the API call `Com_UpdateShadowSignal()`. The atomic transfer of all updated signals of a group into the corresponding I-PDU can be issued via the API call `Com_SendSignalGroup()`. At the receiver side the `Com` module provides the API function `Com_ReceiveSignalGroup()` to atomically receive all signals contained in the group and the API call `Com_ReceiveShadowSignal()` to read the value of a single signal of this group.

For both signal and signal group transmission, the `Com` module takes care of the

- ▶ *endianness* conversion from the sending ECU's endianness to the endianness defined for the transmission (network endianness),
- ▶ the packing of multiple signals into a single I-PDU,
- ▶ and the transmission of this I-PDU depending on the configured transmission mode (see [Section 3.2.1.2.2, "Transmission modes and transfer properties"](#)).

At the receiving ECU, the `Com` module

- ▶ unpacks the signals from the received I-PDU,
- ▶ performs an endianness conversion from the network endianness into the endianness of the receiving ECU,
- ▶ and carries out a sign extension of the received value for signals of signed type.

3.2.1.2.2. Transmission modes and transfer properties

The way an I-PDU is sent out by the `Com` module is called the *I-PDU transmission mode*. `Com` supports up to two transmission modes per I-PDU. During run-time a switch between these two transmission modes is possible.

Depending on the transmission mode, the sending of an I-PDU may be triggered

- ▶ by the lower layer (transmission mode *none*¹)
- ▶ by `Com`-internal timers (transmission mode *periodic*),

¹In case of transmission mode *none*, `Com` does not initiate the transmission of the I-PDU by calling `PduR_ComTransmit()`, but the transmission is triggered by the temporal schedule of the interface module(s) (i.e., by the job list of the `FrIf` or the schedule table of the `LinIf`. These temporal schedules invoke the `PduR_<Net>IfTriggerTransmit()` at the appropriate point in time, which then calls `Com_TriggerTransmit()` in order to retrieve the data from `Com`'s I-PDU buffers.,

- ▶ or by signals with *triggered* transfer property (see below) (transmission mode *direct* or *n-times*²).

Even a combination of periodic and direct/n-times transmission mode is supported (transmission mode *mixed*).

In addition to the I-PDU transmission modes of `Com`, two different *transfer properties* for signals are supported, namely *triggered* transfer and *pending* transfer. In direct/n-times or mixed transmission mode, updating a signal with triggered transfer property via the API call `Com_SendSignal()` immediately transmits the corresponding I-PDU. As opposed to that, updating a signal with pending transfer property does not trigger an immediate transmission.

3.2.1.2.3. I-PDU groups

In the `Com` module multiple I-PDUs can be grouped into an *I-PDU group*. This grouping takes place in a recursive manner, which means that one I-PDU group consisting of multiple I-PDUs can itself be a member of some other I-PDU group. The transmission and reception of configured I-PDU groups can be started and stopped via the API call `Com_IpduGroupControl()`. You can configure the `BswM` module to use this grouping feature, for example, to stop the transmission of all I-PDUs when the `ComM` module reports the internal state silent communication by calling `BswM_ComM_CurrentMode()`. It does so by stopping an I-PDU group that contains all transmit I-PDUs of an ECU.

Stopping an I-PDU on the sender side yields the following behavior:

- ▶ The signal's values updated via `Com_SendSignal()` are still stored within the respective I-PDU buffer, but no transmission of the I-PDU is initiated.
- ▶ The transmission deadline monitoring for this I-PDU (and thus for the contained signals) is disabled.

Stopping an I-PDU on the receiver side leads to the following behavior:

- ▶ Upon an invocation of `Com_RxIndication()` no data is copied into the I-PDU buffer of `Com`.
- ▶ Therefore calls to `Com_ReceiveSignal()` yield the last signal value received prior to the call of `Com_IpduGroupControl()`.
- ▶ The reception deadline monitoring for this I-PDU (and thus for the contained signals) is disabled.

²The n-times transmission mode is a special case of the direct transmission mode. In this mode the first transmission is triggered by the update of a signal with triggered transfer property. After this first transmission `Com` issues *n* subsequent transmissions with a given retransmission period (*n* and the retransmission period are configurable on a per I-PDU basis).

4. ACG8 COM Services module references

4.1. Overview

This chapter provides module references for the ACG8 COM Services product modules. These include a detailed description of all configuration parameters. Furthermore this chapter lists the application programming interface with all data types, constants and functions.

The content of the sections is sorted alphabetically according to the EB tresos AutoCore Generic module names.

For further information on the functional behavior of these modules, refer to the chapter ACG8 COM Services user's guide.

4.1.1. Notation in EB module references

EB notation may differ from the AUTOSAR standard notation in the software specification documents (SWS). This section describes the notation of *default value* and *range* fields in the EB module references.

4.1.1.1. Default value of configuration parameters

If there is no default value specified for a parameter, the default value field is omitted to prevent ambiguity with parameters that have -- as default values.

Example: The parameter `BswMCompuConstText` of the `BswM` module of EB tresos AutoCore Generic 8 Mode Management has no default value field, therefore it is omitted.

4.1.1.2. Range information of configuration parameters

The range of a configuration parameter contains an upper and a lower boundary. However, in special cases the range of allowed values can be computed by means of an XPath function that is evaluated at configuration time. An XPath function can either be a standard `xpath:<function>()` or a custom `cxpath:<function>()` function. The range of a configuration parameter may be computed based on other configuration parameters that are referenced from the XPath function. For more information on custom XPath functions, see section *Custom XPath Functions API* of the EB tresos Studio developer's guide.

Example: The parameter `BswMCompuConstText` of the `BswM` module of EB tresos AutoCore Generic 8 Mode Management has the custom XPath function `cxpath:getCompuMethodsVT()` in the range field which provides the allowed values.

4.2. Com

4.2.1. Configuration parameters

Containers included		
Container name	Multiplicity	Description
ComDefensiveProgramming	1..1	Label: Defensive Programming Options Parameters for defensive programming
ComConfig	1..n	This container contains the configuration parameters and sub containers of the AUTOSAR COM module. This container is a <code>MultipleConfigurationContainer</code> , i.e. this container and its sub-containers exist once per configuration set.
ComGeneral	1..1	Contains the general configuration parameters of the module.
CommonPublishedInformation	1..1	Label: Common Published Information Common container, aggregated by all modules. It contains published information about vendor and versions.
PublishedInformation	1..1	Label: EB Published Information Additional published parameters not covered by <code>CommonPublishedInformation</code> container.

Parameters included	
Parameter name	Multiplicity
IMPLEMENTATION_CONFIG_VARIANT	1..1

Parameter Name	IMPLEMENTATION_CONFIG_VARIANT
Label	Config Variant
Multiplicity	1..1

Type	ENUMERATION
Default value	VariantPostBuild
Range	VariantPostBuild

4.2.1.1. ComDefensiveProgramming

Parameters included	
Parameter name	Multiplicity
ComDefProgEnabled	1..1
ComPrecondAssertEnabled	1..1
ComPostcondAssertEnabled	1..1
ComStaticAssertEnabled	1..1
ComUnreachAssertEnabled	1..1
ComInvariantAssertEnabled	1..1

Parameter Name	ComDefProgEnabled	
Label	Enable Defensive Programming	
Description	<p>Enables or disables the defensive programming feature for the module Com.</p> <p>Note: This feature is dependent on the use of the development error detection module. To use the defensive programming feature, proceed as follows:</p> <ol style="list-style-type: none"> 1. Enable development error detection 2. Enable defensive programming 3. Enable assertions as required 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComPrecondAssertEnabled
Label	Enable Precondition Assertions
Description	<p>Enables handling of precondition assertion checks reported from the module Com.</p> <p>Dependency on parameter(s):</p> <ul style="list-style-type: none"> ▶ Enable Development Error Detection (ComConfigurationUseDet): must be enabled ▶ Enable Defensive Programming (ComDefProgEnabled): must be enabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComPostcondAssertEnabled
Label	Enable Postcondition Assertions
Description	<p>Enables handling of postcondition assertion checks reported from the module Com.</p> <p>Dependency on parameter(s):</p> <ul style="list-style-type: none"> ▶ Enable Development Error Detection (ComConfigurationUseDet): must be enabled ▶ Enable Defensive Programming (ComDefProgEnabled): must be enabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComStaticAssertEnabled
Label	Enable Static Assertions
Description	<p>Enables handling of static assertion checks reported from the module Com.</p> <p>Dependency on parameter(s):</p> <ul style="list-style-type: none"> ▶ Enable Development Error Detection (ComConfigurationUseDet): must be enabled

	► Enable Defensive Programming (ComDefProgEnabled): must be enabled	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComUnreachAssertEnabled	
Label	Enable Unreachable Code Assertions	
Description	<p>Enables handling of unreachable code assertion checks reported from the module Com.</p> <p>Dependency on parameter(s):</p> <ul style="list-style-type: none"> ► Enable Development Error Detection (ComConfigurationUseDet): must be enabled ► Enable Defensive Programming (ComDefProgEnabled): must be enabled 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComInvariantAssertEnabled	
Label	Enable Invariant Assertions	
Description	<p>Enables handling of invariant assertion checks reported from functions of the module Com.</p> <p>Dependency on parameter(s):</p> <ul style="list-style-type: none"> ► Enable Development Error Detection (ComConfigurationUseDet): must be enabled ► Enable Defensive Programming (ComDefProgEnabled): must be enabled 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	Elektrobit Automotive GmbH
---------------	----------------------------

4.2.1.2. ComConfig

Containers included		
Container name	Multiplicity	Description
ComGwMapping	0..n	
ComIPdu	1..n	Contains the configuration parameters of the AUTOSAR COM module's I-PDUs.
ComIPduGroup	1..n	Contains the configuration parameters of the AUTOSAR COM module's I-PDU groups.
ComSignal	0..n	Contains the configuration parameters of the AUTOSAR COM module's signals.
ComSignalGroup	0..n	Contains the configuration parameters of the AUTOSAR COM module's signal groups.
ComTimeBase	0..1	Contains the timebase parameters for Tx, Rx and routing. If this parameter is omitted ComTxMainFunctions and ComRx-MainFunctions can be configured.
ComTxMainFunctions	1..1	Contains the transmission main functions of COM module.
ComRxMainFunctions	1..1	Contains the reception main functions of COM module.

Parameters included	
Parameter name	Multiplicity
ComConfigurationId	1..1

Parameter Name	ComConfigurationId	
Description	This ID is returned by a call to Com_GetConfigurationId.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=4294967295	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.3. ComGwMapping

Containers included		
Container name	Multiplicity	Description
ComGwDestination	1..n	Each instance of this choice container allows to define one routing destination either by reference to an already configured COM signal / signal group or by a destination description container.
ComGwSource	1..1	This choice container allows the definition of the gateway source signal either by reference to an already configured COM signal / signal group or by a source description container.

4.2.1.4. ComGwDestination

Containers included		
Container name	Multiplicity	Description
ComGwDestinationDescription	1..1	Description of a gateway destination. This container allows defining a gateway destination without the configuration of a complete COM signal. This allows adding / changing gateway relations post build without the configuration of new signals.
ComGwSignal	1..1	This container allows specifying a gateway source or destination respectively with a reference to a ComSignal, a ComGroupSignal or a ComSignalGroup.

4.2.1.5. ComGwDestinationDescription

Containers included		
Container name	Multiplicity	Description
ComFilter	0..1	This container contains the configuration parameters of the AUTOSAR COM module's Filters. Note: On sender side the container is used to specify the transmission mode conditions.

Parameters included	
Parameter name	Multiplicity
ComBitPosition	1..1
ComSignalEndianness	1..1

Parameters included	
ComSignalInitValue	0..1
ComTransferProperty	0..1
ComUpdateBitPosition	0..1
ComGwIPduRef	1..1

Parameter Name	ComBitPosition	
Description	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.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=2031	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalEndianness	
Description	Defines the endianness of the signal's network representation.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	BIG_ENDIAN	
	LITTLE_ENDIAN	
	OPAQUE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalInitValue
Description	<p>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:</p> <ul style="list-style-type: none"> ▶ <code>UINT8, UINT16, UINT32, UINT64, SINT8, SINT16, SINT32, SINT64</code>: the string shall be interpreted as defined in the chapter Integer Type in the AUTOSAR EcuC specification.

	<ul style="list-style-type: none"> ▶ Float32, Float64: the string shall be interpreted as defined in the chapter Float Type in the AUTOSAR EcuC specification. ▶ Boolean: the string shall be interpreted as defined in the chapter Boolean Type in the AUTOSAR EcuC specification. ▶ 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. 	
Multiplicity	0..1	
Type	STRING	
Default value	0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComTransferProperty	
Description	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.	
Multiplicity	0..1	
Type	ENUMERATION	
Default value	TRIGGERED	
Range	PENDING	
	TRIGGERED	
	TRIGGERED_ON_CHANGE	
	TRIGGERED_ON_CHANGE_WITHOUT_REPETITION	
	TRIGGERED_WITHOUT_REPETITION	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComUpdateBitPosition	
Description	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..2031 for FlexRay	
Multiplicity	0..1	

Type	INTEGER	
Range	<=2031	
	>=0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComGwIPduRef	
Description	Reference to an I-PDU of a Signal Gateway source or destination description.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.6. ComFilter

Parameters included	
Parameter name	Multiplicity
ComFilterAlgorithm	1..1
ComFilterMask	0..1
ComFilterMax	0..1
ComFilterMin	0..1
ComFilterOffset	0..1
ComFilterPeriod	0..1
ComFilterX	0..1

Parameter Name	ComFilterAlgorithm	
Description	The range of values is specified in the [17] specification, chapter 2.2.2, Reception Filtering.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	ALWAYS	
	MASKED_NEW_DIFFERS_MASKED_OLD	
	MASKED_NEW_DIFFERS_X	

	MASKED_NEW_EQUALS_X	
	NEVER	
	NEW_IS_OUTSIDE	
	NEW_IS_WITHIN	
	ONE_EVERY_N	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterMask	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	0..1	
Type	INTEGER	
Range	<=18446744073709551615	
	>=0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterMax	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	0..1	
Type	INTEGER	
Range	<=18446744073709551615	
	>=0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterMin	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	0..1	
Type	INTEGER	

Range	<=18446744073709551615	
	>=0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterOffset	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering. Range = 0..(ComFilterPeriod-1)	
Multiplicity	0..1	
Type	INTEGER	
Range	<=18446744073709551615	
	>=0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterPeriod	
Description	This parameter defines the period of the ComFilterAlgorithm ONE_EVERY_N.	
Multiplicity	0..1	
Type	INTEGER	
Range	<=18446744073709551615	
	>=0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterX	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	0..1	
Type	INTEGER	
Range	<=18446744073709551615	
	>=0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.7. ComGwSignal

Parameters included	
Parameter name	Multiplicity
ComGwSignalRef	1..1

Parameter Name	ComGwSignalRef
Description	Reference to an object of a gateway relation. Either to a ComSignal, Com-GroupSignal or to a SignalGroup.
Multiplicity	1..1
Type	CHOICE-REFERENCE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

4.2.1.8. ComGwSource

Containers included		
Container name	Multiplicity	Description
ComGwSignal	1..1	This container allows specifying a gateway source or destination respectively with a reference to a ComSignal, a Com-GroupSignal or a ComSignalGroup.
ComGwSourceDescription	1..1	Description of a gateway source. This container allows defining a gateway source without the configuration of a complete COM signal. This allows adding / changing gateway relations post build without the configuration of new signals.

4.2.1.9. ComGwSignal

Parameters included	
Parameter name	Multiplicity
ComGwSignalRef	1..1

Parameter Name	ComGwSignalRef
Description	Reference to an object of a gateway relation. Either to a ComSignal, Com-GroupSignal or to a SignalGroup.

Multiplicity	1..1
Type	CHOICE-REFERENCE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

4.2.1.10. ComGwSourceDescription

Parameters included	
Parameter name	Multiplicity
ComBitPosition	1..1
ComBitSize	0..1
ComSignalEndianness	1..1
ComSignalLength	0..1
ComSignalType	1..1
ComUpdateBitPosition	0..1
ComGwIPduRef	1..1

Parameter Name	ComBitPosition
Description	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.
Multiplicity	1..1
Type	INTEGER
Range	<=2031
	>=0
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComBitSize
Description	Size in bits, for non-array signal types. For ComSignalType UINT8_N and UINT8_DYN this size shall be configured by ComSignalLength.
Multiplicity	0..1

Type	INTEGER
Range	<=64
	>=0
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComSignalEndianness
Description	Defines the endianness of the signal's network representation.
Multiplicity	1..1
Type	ENUMERATION
Range	BIG_ENDIAN
	LITTLE_ENDIAN
	OPAQUE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComSignalLength
Description	Description:.. 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. Range: 0..8 for normal CAN/ LIN I-PDUs, 0..254 for normal FlexRay I-PDUs, and 0..4095 for I-PDUs with ComIPduType TP.
Multiplicity	0..1
Type	INTEGER
Range	<=4095
	>=0
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComSignalType
Description	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.
Multiplicity	1..1

Type	ENUMERATION	
Range	BOOLEAN	
	FLOAT32	
	FLOAT64	
	SINT16	
	SINT32	
	SINT64	
	SINT8	
	UINT16	
	UINT32	
	UINT64	
	UINT8	
	UINT8_DYN	
	UINT8_N	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComUpdateBitPosition	
Description	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..2031 for FlexRay	
Multiplicity	0..1	
Type	INTEGER	
Range	<=2031	
	>=0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComGwIPduRef	
Description	Reference to an I-PDU of a Signal Gateway source or destination description.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	AUTOSAR_ECUC
---------------	--------------

4.2.1.11. ComIPdu

Containers included		
Container name	Multiplicity	Description
ComIPduCounter	1..1	This optional container contains the configuration parameters of PDU Counter. NOTE: Data sequence control is not supported in this version.
ComIPduReplication	1..1	This optional container contains the information needed for each I-PDU replicated. NOTE: Data sequence control is not supported in this version.
ComTxIPdu	1..1	This container contains additional transmission related configuration parameters of the AUTOSAR COM module's I-PDUs.

Parameters included	
Parameter name	Multiplicity
ComIPduCallout	0..1
ComIPduCancellationSupport	0..1
ComIPduDirection	1..1
ComIPduHandleId	1..1
ComIPduSignalProcessing	1..1
ComIPduTriggerTransmitCallout	0..1
ComIPduType	1..1
ComIPduGroupRef	1..n
ComIPduSignalGroupRef	0..n
ComIPduSignalRef	0..n
ComPduIdRef	1..1
ComMainFunctionRef	1..1

Parameter Name	ComIPduCallout
-----------------------	-----------------------

Description	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.	
Multiplicity	0..1	
Type	FUNCTION-NAME	
Configuration class	Link:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComIPduCancellationSupport	
Description	Defines for I-PDUs with ComIPduType NORMAL:. If the underlying IF-module 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.	
Multiplicity	0..1	
Type	BOOLEAN	
Default value	false	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComIPduDirection	
Description	The direction defines if this I-PDU, and therefore the contributing signals and signal groups, shall be sent or received.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	RECEIVE	
	SEND	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComIPduHandleId	
Description	The numerical value used as the ID of this I-PDU. The ComIPduHandleId is required by the API calls to receive I-PDUs from the PduR (ComIP-duDirection: Receive). For Tx-I-PDUs (ComIPduDirection: Send) this handle Id is used by the PduR to confirm the transmission of the ComIPdu. In case no Tx-Confirmation is configured for a Tx-I-PDU, the ComIPduHandleId is not used. NOTE: Handle Ids for TxConfirmation is not supported.	

Multiplicity	1..1
Type	INTEGER
Range	<div><=65535</div> <div>>=0</div>
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComIPduSignalProcessing
Description	For the definition of the two modes Immediate and Deferred.
Multiplicity	1..1
Type	ENUMERATION
Default value	IMMEDIATE
Range	<div>DEFERRED</div> <div>IMMEDIATE</div>
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComIPduTriggerTransmitCallout
Description	If there is a trigger transmit callout defined for this I-PDU this parameter contains the name of the callout function.
Multiplicity	0..1
Type	FUNCTION-NAME
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComIPduType
Description	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. NOTE: This version supports only normal I-PDUs.
Multiplicity	1..1
Type	ENUMERATION
Default value	NORMAL
Range	NORMAL

	TP
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComIPduGroupRef
Description	Reference to the I-PDU groups this I-PDU belongs to.
Multiplicity	1..n
Type	REFERENCE
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComIPduSignalGroupRef
Description	References to all signal groups contained in this I-Pdu.
Multiplicity	0..n
Type	REFERENCE
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComIPduSignalRef
Description	References to all signals contained in this I-PDU.
Multiplicity	0..n
Type	REFERENCE
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComPduldRef
Description	Reference to the "global" Pdu structure to allow harmonization of handle IDs in the COM-Stack.
Multiplicity	1..1
Type	REFERENCE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComMainFunctionRef
-----------------------	---------------------------

Multiplicity	1..1
Type	CHOICE-REFERENCE
Configuration class	PostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

4.2.1.12. ComIPduCounter

Parameters included	
Parameter name	Multiplicity
ComIPduCounter-ErrorNotification	0..1
ComIPduCounterSize	1..1
ComIPduCounterStart-Position	1..1
ComIPduCounterThreshold	0..1

Parameter Name	ComIPduCounterErrorNotification
Description	Name of Com_CbkCounterErr callback function to be called. If this. parameter is omitted no I-PDU counter mismatch notification shall take place.
Multiplicity	0..1
Type	FUNCTION-NAME
Configuration class	Link: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComIPduCounterSize
Description	Size of PDU Counter expressed in bits.
Multiplicity	1..1
Type	INTEGER
Range	<=8 >=1
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComIPduCounterStartPosition	
Description	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.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=2031	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComIPduCounterThreshold	
Description	Threshold value of I-PDU counter algorithm, see COM590.	
Multiplicity	0..1	
Type	INTEGER	
Default value	0	
Range	<=255	
	>=0	
Configuration class	PreCompile:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.13. ComIPduReplication

Parameters included	
Parameter name	Multiplicity
ComIPduReplication-Quorum	1..1
ComIPduReplicaRef	1..2

Parameter Name	ComIPduReplicationQuorum
Description	The number of identical I-PDUs needed for successful voting.
Multiplicity	1..1

Type	INTEGER	
Range	<=3	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComIPduReplicaRef	
Description	Reference to replicas PduR PDUs of this IPDU.	
Multiplicity	1..2	
Type	REFERENCE	
Configuration class	PreCompile:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.14. ComTxIPdu

Containers included		
Container name	Multiplicity	Description
ComTxModeFalse	0..1	This container contains the configuration parameters of the AUTOSAR COM module's transmission modes in the case the ComFilter evaluates to false.
ComTxModeTrue	0..1	This container contains the configuration parameters of the AUTOSAR COM module's transmission modes in the case the ComFilter evaluates to true.

Parameters included	
Parameter name	Multiplicity
ComMinimumDelay-Time	0..1
ComTxIPduClearUpdateBit	1..1
ComTxIPduUnusedAreasDefault	1..1

Parameter Name	ComMinimumDelayTime
-----------------------	----------------------------

Description	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.	
Multiplicity	0..1	
Type	FLOAT	
Default value	0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC V3.0	

Parameter Name	ComTxIPduClearUpdateBit	
Description	Defines when the update-bits of signals or signal groups, contained in this I-PDU, will be cleared. Parameter is enabled automatically if at least on signal or signal group is referenced which has the update bit enabled.	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	Transmit	
Range	Confirmation	
	Transmit	
	TriggerTransmit	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComTxIPduUnusedAreasDefault	
Description	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.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=255	

	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.15. ComTxModeFalse

Containers included		
Container name	Multiplicity	Description
ComTxMode	1..1	This container contains the configuration parameters of the AUTOSAR COM module's transmission modes.

4.2.1.16. ComTxMode

Parameters included	
Parameter name	Multiplicity
ComTxModeMode	1..1
ComTxModeNumberOfRepetitions	1..1
ComTxModeRepetitionPeriod	1..1
ComTxModeTimeOffset	1..1
ComTxModeTimePeriod	1..1

Parameter Name	ComTxModeMode
Description	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.
Multiplicity	1..1
Type	ENUMERATION
Range	DIRECT
	MIXED
	NONE
	PERIODIC
Configuration class	VariantPostBuild: VariantPostBuild

Origin	AUTOSAR_ECUC
---------------	--------------

Parameter Name	ComTxModeNumberOfRepetitions	
Description	Defines the number of repetitions for the transmission mode DIRECT and the event driven part of transmission mode MIXED.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=255	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComTxModeRepetitionPeriod	
Description	Defines the repetition period in seconds of the multiple transmissions in case ComTxModeNumberOfRepetitions is configured greater than 1 and ComTxModeMode is configured to DIRECT or MIXED. In case of the mixed transmission mode only the event driven part is affected.	
Multiplicity	1..1	
Type	FLOAT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComTxModeTimeOffset	
Description	Defines the period in seconds between the start of the I-PDU by Com_IpduGroupControl 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.	
	<p>EB implementation:</p> <ul style="list-style-type: none"> ▶ For values > 0: number of Com_MainFunctionTx invocations between the first periodic transmission and invocation of Com_IpduGroupControl is ComTxModeTimeOffset / ComTxTimeBase. ▶ For value = 0 (or omitted): first transmission request is in the next invocation of Com_MainFunctionTx. 	

	EB extension: If the value is lower 0, the I-PDU is sent out immediately (in context of Com_IpduGroupControl)	
Multiplicity	1..1	
Type	FLOAT	
Default value	0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComTxModeTimePeriod	
Description	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.	
Multiplicity	1..1	
Type	FLOAT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.17. ComTxModeTrue

Containers included		
Container name	Multiplicity	Description
ComTxMode	1..1	This container contains the configuration parameters of the AUTOSAR COM module's transmission modes.

4.2.1.18. ComTxMode

Parameters included	
Parameter name	Multiplicity
ComTxModeMode	1..1
ComTxModeNumberOfRepetitions	1..1
ComTxModeRepetitionPeriod	1..1
ComTxModeTimeOffset	1..1

Parameters included	
ComTxModeTimePeriod	1..1

Parameter Name	ComTxModeMode	
Description	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.	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	DIRECT	
Range	DIRECT	
	MIXED	
	NONE	
	PERIODIC	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComTxModeNumberOfRepetitions	
Description	Defines the number of repetitions for the transmission mode DIRECT and the event driven part of transmission mode MIXED.	
Multiplicity	1..1	
Type	INTEGER	
Default value	1	
Range	<=255	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

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

Type	FLOAT
Default value	1
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComTxModeTimeOffset
Description	<p>Defines the period in seconds between the start of the I-PDU by Com_IpduGroupControl 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.</p> <p>In case ComTxModeTimeOffset is omitted or configured to 0, the first periodic transmission shall be transmitted within the next invocation of Com_MainFunctionTx.</p> <p>EB implementation:</p> <ul style="list-style-type: none"> ▶ For values > 0: number of Com_MainFunctionTx invocations between the first periodic transmission and invocation of Com_IpduGroupControl is ComTxModeTimeOffset / ComTxTimeBase. ▶ For value = 0 (or omitted): first transmission request is in the next invocation of Com_MainFunctionTx. <p>EB extension: If the value is lower 0, the I-PDU is sent out immediately (in context of Com_IpduGroupControl)</p>
Multiplicity	1..1
Type	FLOAT
Default value	0
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComTxModeTimePeriod
Description	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.
Multiplicity	1..1
Type	FLOAT
Configuration class	VariantPostBuild: VariantPostBuild

Origin	AUTOSAR_ECUC
--------	--------------

4.2.1.19. ComIPduGroup

Parameters included	
Parameter name	Multiplicity
ComIPduGroupHandleId	1..1
ComIPduGroupGroupRef	0..n

Parameter Name	ComIPduGroupHandleId	
Description	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)	
Multiplicity	1..1	
Type	INTEGER	
Range	<=65535	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComIPduGroupGroupRef	
Description	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.	
Multiplicity	0..n	
Type	REFERENCE	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.20. ComSignal

Containers included		
Container name	Multiplicity	Description

Containers included		
ComFilter	0..1	This container contains the configuration parameters of the AUTOSAR COM module's Filters. Note: On sender side the container is used to specify the transmission mode conditions.

Parameters included	
Parameter name	Multiplicity
ComBitPosition	1..1
ComBitSize	1..1
ComDataInvalidAction	0..1
ComErrorNotification	0..1
ComFirstTimeout	0..1
ComHandleId	1..1
ComInitialValueOnly	0..1
ComInvalidNotification	1..1
ComNotification	0..1
ComRxDataTimeoutAction	1..1
ComSignalDataInvalidValue	0..1
ComSignalEndianness	1..1
ComSignalInitValue	0..1
ComSignalLength	1..1
ComSignalType	1..1
ComTimeout	0..1
ComTimeoutNotification	0..1
ComTransferProperty	0..1
ComUpdateBitPosition	0..1
ComSystemTemplateSystemSignalRef	0..1

Parameter Name	ComBitPosition
Description	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.

Multiplicity	1..1
Type	INTEGER
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComBitSize
Description	Size in bits, for non-array signal types. For ComSignalType UINT8_N and UINT8_DYN this size shall be configured by ComSignalLength.
Multiplicity	1..1
Type	INTEGER
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComDataInvalidAction
Description	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 ComSignalInitValue will be used for the replacement. NOTE: Signal Invalidation is not supported by the AUTOSAR COM module. Please make sure, that the RTE supports signal invalidation.
Multiplicity	0..1
Type	ENUMERATION
Default value	NOTIFY
Range	NOTIFY REPLACE
Configuration class	Link: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComErrorNotification
Description	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.
Multiplicity	0..1
Type	FUNCTION-NAME
Configuration class	Link: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComFirstTimeout
Description	<p>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 COM263_Conf. According to AUTOSAR SWS COM requirement COM716 the AUTOSAR COM module shall not monitor the reception of this signal or signal group respectively from the start of the corresponding I-PDU until the first reception if the configuration parameter ComFirstTimeout for a signal or signal group is omitted or configured to 0. The implementation behaves follows regarding the configuration parameter ComFirstTimeout for a signal or signal group:</p> <ul style="list-style-type: none"> ▶ If configured to 0: as defined in COM716 ▶ If omitted: ComTimeout is used for ComFirstTimeout
Multiplicity	0..1
Type	FLOAT
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComHandleId
Description	<p>The numerical value used as the ID. For signals it is required by the API calls Com_UpdateShadowSignal, Com_ReceiveShadowSignal and Com_InvalidateShadowSignal. For signals groups it is required by the Com_SendSignalGroup and Com_ReceiveSignalGroup calls.</p>
Multiplicity	1..1
Type	INTEGER
Range	<p><=65535</p> <p>>=0</p>
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComInitialValueOnly
Description	<p>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).</p>
Multiplicity	0..1
Type	BOOLEAN

Default value	false
Configuration class	PostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComInvalidNotification
Description	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 an invalidated signal/ signal group. Only applicable if ComDataInvalidAction is configured to NOTIFY. NOTE: Signal Invalidation is not supported by the AUTOSAR COM module. Please make sure, that the RTE supports signal invalidation. NOTE: Parameter is enabled automatically when ComDataInvalidAction == NOTIFY.
Multiplicity	1..1
Type	FUNCTION-NAME
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComNotification
Description	On sender side: Name of Com_CbkTxAck callback function to be called. On receiver side: Name of Com_CbkRxAck callback function to be called. If this parameter is omitted no notification shall take place.
Multiplicity	0..1
Type	FUNCTION-NAME
Configuration class	Link: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComRxDataTimeoutAction
Description	This parameter defines the action performed upon expiration of the reception deadline monitoring timer.
Multiplicity	1..1
Type	ENUMERATION
Range	NONE REPLACE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComSignalDataInvalidValue
-----------------------	----------------------------------

Description	<p>Defines the data invalid value of the signal. In case the ComSignalType is:</p> <ul style="list-style-type: none"> ▶ <code>UINT8</code>, <code>UINT16</code>, <code>UINT32</code>, <code>SINT8</code>, <code>SINT16</code>, <code>SINT32</code>: the string shall be interpreted as defined in the chapter Integer Type in the AUTOSAR EcuC specification. ▶ <code>FLOAT32</code>, <code>FLOAT64</code>: the string shall be interpreted as defined in the chapter Float Type in the AUTOSAR EcuC specification. ▶ <code>BOOLEAN</code>: the string shall be interpreted as defined in the chapter Boolean Type in the AUTOSAR EcuC specification. ▶ <code>UINT8_N</code>, <code>UINT8_DYN</code>: 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 <code>UINT8_DYN</code> the dynamic length shall be set to the number of configured characters. An empty string "" shall be interpreted as 0-sized dynamic signal. 	
Multiplicity	0..1	
Type	STRING	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalEndianness	
Description	Defines the endianness of the signal's network representation.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	BIG_ENDIAN	
	LITTLE_ENDIAN	
	OPAQUE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalInitValue	
Description	<p>Initial value for this signal. In case of <code>UINT8_N</code> the default value is a string of length <code>ComSignalLength</code> with all bytes set to 0x00. In case of <code>UINT8_DYN</code> the initial size shall be 0. In case the ComSignalType is:</p> <ul style="list-style-type: none"> ▶ <code>UINT8</code>, <code>UINT16</code>, <code>UINT32</code>, <code>UINT64</code>, <code>SINT8</code>, <code>SINT16</code>, <code>SINT32</code>, <code>SINT64</code>: the string shall be interpreted as defined in the chapter Integer Type in the AUTOSAR EcuC specification. 	

	<ul style="list-style-type: none"> ▶ Float32, Float64: the string shall be interpreted as defined in the chapter Float Type in the AUTOSAR EcuC specification. ▶ Boolean: the string shall be interpreted as defined in the chapter Boolean Type in the AUTOSAR EcuC specification. ▶ 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. 	
Multiplicity	0..1	
Type	STRING	
Default value	0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalLength	
Description	Description: 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. Range: 0..8 for normal CAN/ LIN I-PDUs, 0..254 for normal FlexRay I-PDUs, and 0..4095 for I-PDUs with ComIPduType TP.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalType	
Description	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.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	BOOLEAN	
	FLOAT32	
	FLOAT64	

	SINT16
	SINT32
	SINT64
	SINT8
	UINT16
	UINT32
	UINT64
	UINT8
	UINT8_DYN
	UINT8_N
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComTimeout
Description	Defines the length of the deadline monitoring timeout period in seconds. The period for the first timeout period (reception deadline monitoring only) can be configured separately by COM183_Conf.
Multiplicity	0..1
Type	FLOAT
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComTimeoutNotification
Description	On sender side: Name of Com_CbkTxTOut callback function to be called. On receiver side: Name of Com_CbkRxTOut callback function to be called.
Multiplicity	0..1
Type	FUNCTION-NAME
Configuration class	Link: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComTransferProperty
Description	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.

	<ul style="list-style-type: none"> ▶ PENDING: A write access to this signal never triggers the transmission of the corresponding I-PDU. ▶ TRIGGERED: Depending on the transmission mode, a write access to this signal can trigger the transmission of the corresponding I-PDU. ▶ TRIGGERED_ON_CHANGE: Depending on the transmission mode, a write access to this signal can trigger the transmission of the corresponding I-PDU, but only in case the written value is different to the locally stored (last written or init) value. 	
Multiplicity	0..1	
Type	ENUMERATION	
Default value	TRIGGERED	
Range	PENDING	
	TRIGGERED	
	TRIGGERED_ON_CHANGE	
	TRIGGERED_ON_CHANGE_WITHOUT_REPETITION	
	TRIGGERED_WITHOUT_REPETITION	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComUpdateBitPosition	
Description	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..2031 for FlexRay	
Multiplicity	0..1	
Type	INTEGER	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSystemTemplateSystemSignalRef	
Description	Reference to the ISignalToIPduMapping that contains a reference to the ISignal (System Template) which this ComSignal (or ComGroupSignal) represents.	
Multiplicity	0..1	
Type	FOREIGN-REFERENCE	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.21. ComFilter

Parameters included	
Parameter name	Multiplicity
ComFilterAlgorithm	1..1
ComFilterMask	1..1
ComFilterMax	1..1
ComFilterMin	1..1
ComFilterOffset	1..1
ComFilterPeriod	1..1
ComFilterX	1..1

Parameter Name	ComFilterAlgorithm	
Description	The range of values is specified in the [17] specification, chapter 2.2.2, Reception Filtering.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	ALWAYS	
	MASKED_NEW_DIFFERS_MASKED_OLD	
	MASKED_NEW_DIFFERS_X	
	MASKED_NEW_EQUALS_X	
	NEVER	
	NEW_IS_OUTSIDE	
	NEW_IS_WITHIN	
	ONE_EVERY_N	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterMask	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	AUTOSAR_ECUC
---------------	--------------

Parameter Name	ComFilterMax	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterMin	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterOffset	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering. Range = 0..(ComFilterPeriod-1)	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterPeriod	
Description	This parameter defines the period of the ComFilterAlgorithm ONE_EVERY_N.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterX	
-----------------------	-------------------	--

Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.22. ComSignalGroup

Containers included		
Container name	Multiplicity	Description
ComGroupSignal	0..n	This container contains the configuration parameters of group signals. I.e. signals that are included within a signal group.

Parameters included	
Parameter name	Multiplicity
ComDataInvalidAction	0..1
ComErrorNotification	0..1
ComFirstTimeout	0..1
ComHandleId	1..1
ComInitialValueOnly	0..1
ComInvalidNotification	1..1
ComNotification	0..1
ComRxDataTimeoutAction	1..1
ComSignalGroupArrayAccess	1..1
ComTimeout	0..1
ComTimeoutNotification	0..1
ComTransferProperty	0..1
ComUpdateBitPosition	0..1
ComSystemTemplateSignalGroupRef	0..1

Parameter Name	ComDataInvalidAction
-----------------------	-----------------------------

Description	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 ComSignalInitValue will be used for the replacement. NOTE: Signal Invalidation is not supported by the AUTOSAR COM module. Please make sure, that the RTE supports signal invalidation.	
Multiplicity	0..1	
Type	ENUMERATION	
Default value	NOTIFY	
Range	NOTIFY	
	REPLACE	
Configuration class	Link:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComErrorNotification	
Description	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.	
Multiplicity	0..1	
Type	FUNCTION-NAME	
Configuration class	Link:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFirstTimeout	
Description	<p>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 COM263_ - Conf. According to AUTOSAR SWS COM requirement COM716 the AUTOSAR COM module shall not monitor the reception of this signal or signal group respectively from the start of the corresponding I-PDU until the first reception if the configuration parameter ComFirstTimeout for a signal or signal group is omitted or configured to 0. The implementation behaves follows regarding the configuration parameter ComFirstTimeout for a signal or signal group:</p> <ul style="list-style-type: none"> ▶ If configured to 0: as defined in COM716 ▶ If omitted: ComTimeout is used for ComFirstTimeout 	
Multiplicity	0..1	
Type	FLOAT	
Configuration class	PostBuild:	VariantPostBuild

Origin	AUTOSAR_ECUC
---------------	--------------

Parameter Name	ComHandleId
Description	The numerical value used as the ID. For signals it is required by the API calls Com_UpdateShadowSignal, Com_ReceiveShadowSignal and Com_InvalidateShadowSignal. For signals groups it is required by the Com_SendSignalGroup and Com_ReceiveSignalGroup calls.
Multiplicity	1..1
Type	INTEGER
Range	<div><=65535</div> <div>>=0</div>
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComInitialValueOnly
Description	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).
Multiplicity	0..1
Type	BOOLEAN
Default value	false
Configuration class	PostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComInvalidNotification
Description	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 an invalidated signal/ signal group. Only applicable if ComDataInvalidAction is configured to NOTIFY. NOTE: Signal Invalidation is not supported by the AUTOSAR COM module. Please make sure, that the RTE supports signal invalidation. NOTE: Parameter is enabled automatically when ComDataInvalidAction == NOTIFY.
Multiplicity	1..1
Type	FUNCTION-NAME
Configuration class	VariantPostBuild: VariantPostBuild

Origin	AUTOSAR_ECUC
---------------	--------------

Parameter Name	ComNotification	
Description	On sender side: Name of Com_CbkTxAck callback function to be called. On receiver side: Name of Com_CbkRxAck callback function to be called. If this parameter is omitted no notification shall take place.	
Multiplicity	0..1	
Type	FUNCTION-NAME	
Configuration class	Link:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComRxDataTimeoutAction	
Description	This parameter defines the action performed upon expiration of the reception deadline monitoring timer.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	NONE	
	REPLACE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalGroupArrayAccess	
Description	Defines whether the uint8-array based access shall be used for this ComSignal-Group. This parameter is only enabled if ComBasedTransformerSupportTx or ComBasedTransformerSupportRx is true.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	PostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTimeout	
Description	Defines the length of the deadline monitoring timeout period in seconds. The period for the first timeout period can be configured separately by COM183_Conf.	

Multiplicity	0..1
Type	FLOAT
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComTimeoutNotification
Description	On sender side: Name of Com_CbkTxTOut callback function to be called. On receiver side: Name of Com_CbkRxTOut callback function to be called.
Multiplicity	0..1
Type	FUNCTION-NAME
Configuration class	Link: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComTransferProperty
Description	<p>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.</p> <ul style="list-style-type: none"> ▶ PENDING: A write access to this signal never triggers the transmission of the corresponding I-PDU. ▶ TRIGGERED: Depending on the transmission mode, a write access to this signal can trigger the transmission of the corresponding I-PDU. ▶ TRIGGERED_ON_CHANGE: Depending on the transmission mode, a write access to this signal can trigger the transmission of the corresponding I-PDU, but only in case the written value is different to the locally stored (last written or init) value.
Multiplicity	0..1
Type	ENUMERATION
Default value	TRIGGERED
Range	PENDING TRIGGERED TRIGGERED_ON_CHANGE TRIGGERED_ON_CHANGE_WITHOUT_REPETITION TRIGGERED_WITHOUT_REPETITION
Configuration class	PostBuild: VariantPostBuild

Origin	AUTOSAR_ECUC	
Parameter Name	ComUpdateBitPosition	
Description	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..2031 for FlexRay	
Multiplicity	0..1	
Type	INTEGER	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSystemTemplateSignalGroupRef	
Description	Reference to the ISignalToIPduMapping that contains a reference to the ISignal-Group (SystemTemplate) which this ComSignalGroup represents.	
Multiplicity	0..1	
Type	FOREIGN-REFERENCE	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.23. ComGroupSignal

Containers included		
Container name	Multiplicity	Description
ComFilter	0..1	This container contains the configuration parameters of the AUTOSAR COM module's Filters. Note: On sender side the container is used to specify the transmission mode conditions.

Parameters included	
Parameter name	Multiplicity
ComBitPosition	1..1
ComBitSize	1..1
ComHandleId	1..1
ComSignalDataInvalid-Value	0..1
ComSignalEndianness	1..1

Parameters included	
ComSignalInitValue	0..1
ComSignalLength	1..1
ComSignalType	1..1
ComTransferProperty	1..1
ComSystemTemplateSystemSignalRef	0..1

Parameter Name	ComBitPosition
Description	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.
Multiplicity	1..1
Type	INTEGER
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComBitSize
Description	Size in bits, for non-array signal types. For ComSignalType UINT8_N and UINT8_DYN this size shall be configured by ComSignalLength.
Multiplicity	1..1
Type	INTEGER
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComHandleId
Description	The numerical value used as the ID. For signals it is required by the API calls Com_UpdateShadowSignal, Com_ReceiveShadowSignal and Com_InvalidateShadowSignal. For signals groups it is required by the Com_SendSignalGroup and Com_ReceiveSignalGroup calls.
Multiplicity	1..1
Type	INTEGER
Range	<div><=65535</div> <div>>=0</div>

Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalDataInvalidValue	
Description	<p>Defines the data invalid value of the signal. In case the ComSignalType is:</p> <ul style="list-style-type: none"> ▶ <code>UINT8</code>, <code>UINT16</code>, <code>UINT32</code>, <code>UINT64</code>, <code>SINT8</code>, <code>SINT16</code>, <code>SINT32</code>, <code>SINT64</code>: the string shall be interpreted as defined in the chapter Integer Type in the AUTOSAR EcuC specification. ▶ <code>FLOAT32</code>, <code>FLOAT64</code>: the string shall be interpreted as defined in the chapter Float Type in the AUTOSAR EcuC specification. ▶ <code>BOOLEAN</code>: the string shall be interpreted as defined in the chapter Boolean Type in the AUTOSAR EcuC specification. ▶ <code>UINT8_N</code>, <code>UINT8_DYN</code>: 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 <code>UINT8_DYN</code> the dynamic length shall be set to the number of configured characters. An empty string "" shall be interpreted as 0-sized dynamic 	
Multiplicity	0..1	
Type	STRING	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalEndianness	
Description	Defines the endianness of the signal's network representation.	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	LITTLE_ENDIAN	
Range	BIG_ENDIAN	
	LITTLE_ENDIAN	
	OPAQUE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalInitValue	
-----------------------	---------------------------	--

Description	<p>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:</p> <ul style="list-style-type: none"> ▶ <code>UINT8</code>, <code>UINT16</code>, <code>UINT32</code>, <code>UINT64</code>, <code>SINT8</code>, <code>SINT16</code>, <code>SINT32</code>, <code>SINT64</code>: the string shall be interpreted as defined in the chapter Integer Type in the AUTOSAR EcuC specification. ▶ <code>FLOAT32</code>, <code>FLOAT64</code>: the string shall be interpreted as defined in the chapter Float Type in the AUTOSAR EcuC specification. ▶ <code>BOOLEAN</code>: the string shall be interpreted as defined in the chapter Boolean Type in the AUTOSAR EcuC specification. ▶ <code>UINT8_N</code>, <code>UINT8_DYN</code>: 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 <code>UINT8_DYN</code> the dynamic length shall be set to the number of configured characters. An empty string "" shall be interpreted as 0-sized dynamic signal. 	
Multiplicity	0..1	
Type	STRING	
Default value	0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalLength	
Description	<p>Description:.. For ComSignalType <code>UINT8_N</code> this parameter specifies the length n in bytes. For ComSignalType <code>UINT8_DYN</code> it specifies the maximum length in bytes. For all other types this parameter shall be ignored. Range: 0..8 for normal CAN/ LIN I-PDUs, 0..254 for normal FlexRay I-PDUs, and 0..4095 for I-PDUs with ComIPduType TP.</p>	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSignalType	
Description	<p>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.</p>	

Multiplicity	1..1
Type	ENUMERATION
Range	BOOLEAN
	FLOAT32
	FLOAT64
	SINT16
	SINT32
	SINT64
	SINT8
	UINT16
	UINT32
	UINT64
	UINT8
	UINT8_DYN
	UINT8_N
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComTransferProperty
Description	Optionally defines whether this group signal shall contribute to the TRIGGERED_ON_CHANGE transfer property of the signal group. If at least one group signal of a signal group has the ComTransferProperty configured all other group signals of that signal group shall have the attribute configured as well. PENDING: a change of the value of this group signal shall not be considered in the evaluation of the signal groups ComTransferProperty. TRIGGERED_ON_CHANGE: a change of the value of this group signal shall be considered in the in the evaluation of the signal groups ComTransferProperty.
Multiplicity	1..1
Type	ENUMERATION
Default value	TRIGGERED_ON_CHANGE
Range	PENDING
	TRIGGERED_ON_CHANGE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComSystemTemplateSystemSignalRef	
Description	Reference to the ISignalToIPduMapping that contains a reference to the ISignal (System Template) which this ComSignal (or ComGroupSignal) represents.	
Multiplicity	0..1	
Type	FOREIGN-REFERENCE	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.24. ComFilter

Parameters included	
Parameter name	Multiplicity
ComFilterAlgorithm	1..1
ComFilterMask	1..1
ComFilterMax	1..1
ComFilterMin	1..1
ComFilterOffset	1..1
ComFilterPeriod	1..1
ComFilterX	1..1

Parameter Name	ComFilterAlgorithm
Description	The range of values is specified in the [17] specification, chapter 2.2.2, Reception Filtering.
Multiplicity	1..1
Type	ENUMERATION
Range	ALWAYS
	MASKED_NEW_DIFFERS_MASKED_OLD
	MASKED_NEW_DIFFERS_X
	MASKED_NEW_EQUALS_X
	NEVER
	NEW_IS_OUTSIDE
	NEW_IS_WITHIN
	ONE EVERY_N

Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterMask	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterMax	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterMin	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterOffset	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering. Range = 0..(ComFilterPeriod-1)	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	AUTOSAR_ECUC
---------------	--------------

Parameter Name	ComFilterPeriod	
Description	This parameter defines the period of the ComFilterAlgorithm ONE_EVERY_N.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComFilterX	
Description	The name of this attribute corresponds to the parameter name in the [17] specification of Reception Filtering.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.2.1.25. ComTimeBase

Parameters included	
Parameter name	Multiplicity
ComGwTimeBase	1..1
ComRxTimeBase	1..1
ComTxTimeBase	1..1

Parameter Name	ComGwTimeBase
Description	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. The COM module (generator) might rely on the fact that Com_MainFunctionRouteSignals is scheduled according to the value configured here.
Multiplicity	1..1
Type	FLOAT

Default value	0.005
Range	<=3600
	>=0
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComRxTimeBase
Description	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. The COM module (generator) may rely on the fact that Com_MainFunctionRx is scheduled according to the value configured here.
Multiplicity	1..1
Type	FLOAT
Default value	0.005
Range	<=3600
	>=0
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComTxTimeBase
Description	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. The COM module (generator) may rely on the fact that Com_MainFunctionTx is scheduled according to the value configured here.
Multiplicity	1..1
Type	FLOAT
Default value	0.005
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

4.2.1.26. ComTxMainFunctions

Containers included		
Container name	Multiplicity	Description
Com_MainFunctionTx	1..n	Contains the transmission main functions of COM module.

4.2.1.27. Com_MainFunctionTx

Parameters included	
Parameter name	Multiplicity
ComTxTimeBase	1..1
ComPartitionRef	1..1

Parameter Name	ComTxTimeBase	
Description	The period between successive calls to Com transmission main functions in seconds. This parameter is 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. The COM module (generator) relies on the fact that the appropriate Com main function is scheduled according to the value configured here.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.005	
Range	<=3600	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComPartitionRef	
Description	References a EcuC partition to allow grouping of Tx Com main functions according to EcucPartition elements. This grouping is used to implement dedicated BswImplementations.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	Elektrobit Automotive GmbH
---------------	----------------------------

4.2.1.28. ComRxMainFunctions

Containers included		
Container name	Multiplicity	Description
Com_MainFunctionRx	1..n	Contains the reception main functions of COM module.

4.2.1.29. Com_MainFunctionRx

Parameters included	
Parameter name	Multiplicity
ComRxTimeBase	1..1
ComPartitionRef	1..1

Parameter Name	ComRxTimeBase	
Description	The period between successive calls to Com reception main functions in seconds. This parameter is 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. The COM module (generator) relies on the fact that the appropriate Com main function is scheduled according to the value configured here.	
Multiplicity	1..1	
Type	FLOAT	
Default value	0.005	
Range	<=3600	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComPartitionRef
Description	References a EcuC partition to allow grouping of Rx Com main functions according to EcucPartition elements. This grouping is used to implement dedicated BswImplementations.

Multiplicity	1..1
Type	REFERENCE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

4.2.1.30. ComGeneral

Containers included		
Container name	Multiplicity	Description
VendorSpecific	1..1	Contains the vendor specific configuration parameters of the AUTOSAR COM module.

Parameters included	
Parameter name	Multiplicity
ComConfigura- tionUseDet	0..1
ComEnableMDTFor- CyclicTransmission	0..1
ComRetryFailedTrans- mitRequests	0..1
ComSupportedIP- duGroups	1..1
ComVersionInfoApi	1..1
ComEnableSignal- GroupArrayApi	1..1

Parameter Name	ComConfigurationUseDet
Description	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), see COM028.
Multiplicity	0..1
Type	BOOLEAN
Default value	true
Configuration class	PreCompile: VariantPostBuild

Origin	AUTOSAR_ECUC
---------------	--------------

Parameter Name	ComEnableMDTForCyclicTransmission	
Description	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).	
Multiplicity	0..1	
Type	BOOLEAN	
Default value	true	
Configuration class	PreCompile:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComRetryFailedTransmitRequests	
Description	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.	
Multiplicity	0..1	
Type	BOOLEAN	
Default value	false	
Configuration class	PreCompile:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComSupportedIPduGroups	
Description	Defines the maximum number of supported I-PDU groups.	
Multiplicity	1..1	
Type	INTEGER	
Default value	32	
Range	<=65535	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	ComVersionInfoApi	
Description	Activate/Deactivate the version information API (Com_GetVersionInfo). True: version information API activated False: version information API deactivated	

Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	ComEnableSignalGroupArrayApi
Description	Activate/Deactivate the signal group array access APIs (Com_SendSignalGroupArray, Com_ReceiveSignalGroupArray). true: signal group array access APIs activated; Please use the parameter ComBasedTransformerSupportTx and ComBasedTransformerSupportRx to enable and disable the APIs individually false: signal group array access APIs deactivated
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

4.2.1.31. VendorSpecific

Containers included		
Container name	Multiplicity	Description
ComGeneratedRxSignal	0..1	<p>This container contains the configuration parameters for the generated functions / macros for reading signal values. NOTE> if this container is enabled, the macros are generated.</p> <ul style="list-style-type: none"> ▶ ENABLED: Macros for signal extraction are generated. Generation of functions depends on ComGeneratedRcvSigEnable. ▶ DISABLED: Neither macros nor functions for signal are generated. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): Using these macros and/or functions reduces the execution time extraction of signals.

Containers included		
		► ROM increase (code): Using these macros and/or functions increases the ROM consumption of the module code.

Parameters included	
Parameter name	Multiplicity
ComDataMemSize	0..1
ComRamSizeMax	1..1
ComCbkTxTOutArraySizeMax	1..1
ComCbkRxTOutArraySizeMax	1..1
ComCbkRxAckPtrArraySizeMax	1..1
ComCbkTxAckPtrArraySizeMax	1..1
ComCallOutFuncPtrArraySizeMax	1..1
ComTriggerTxCallOutEnable	1..1
ComRxDataTimeoutAction	1..1
ComRxTimeoutFactorSize	1..1
ComRxFirstTimeoutFactorSize	1..1
ComTxTimeoutFactorSize	1..1
ComTxModeRepetitionPeriodFactorS	1..1
ComTxModeTimeOffsetFactorSize	1..1
ComTxModeTimePeriodFactorSize	1..1
ComTxIpduMDTFactorSize	1..1

Parameters included	
ComUpdateBitRxConfig	1..1
ComUpdateBitTxConfig	1..1
ComClearUpdateBitTx- TransmitEnable	1..1
ComClearUpdateBitTx- TriggerTransmitEnable	1..1
ComClearUpdateBitTx- TxConfirmationEnable	1..1
ComTmsEnable	1..1
ComFilterReceiverEn- able	1..1
ComTxDynLengthIP- duEnable	1..1
ComRxDynLengthIP- duEnable	1..1
ComFilterOneEveryN- PeriodOffSMax	1..1
ComFilterOneEv- eryNOccuranceMax	1..1
ComTxModeDirectEn- able	1..1
ComTxModeNTimesEn- able	1..1
ComTxModePeriodicEn- able	1..1
ComTxModeMixedDi- rectEnable	1..1
ComTxModeMixed- NTimesEnable	1..1
ComTxSigConfDeferre- dEnable	1..1
ComTxSigConfImmedi- ateEnable	1..1
ComRxSigConfDeferre- dEnable	1..1

Parameters included	
ComRxSigConflmmediateEnable	1..1
ComSignalGwEnable	1..1
ComCheckValueSizeEnable	1..1
ComConstCfgAddressEnable	1..1
ComConstCfgAddress	1..1
ComRelocatableCfgEnable	1..1
Com_TxF_MaskNewDiffersMaskOld_En	1..1
ComSigGwRxFilterEnable	1..1
ComTransfPropTriggeredEnable	1..1
ComTransfPropTriggeredOCEnable	1..1
ComTransfPropWithoutRepEnable	1..1
ComRxTpAPIEnable	1..1
ComTxTpAPIEnable	1..1
ComTxZeroSignalEnable	1..1
ComBasedTransformerSupportTx	1..1
ComBasedTransformerSupportRx	1..1
ComTxGroupSignalNoLock	1..1
ComDeferTx2MainFunc	1..1
Parameter Name	ComDataMemSize
Description	Size of internal Com data in units of bytes (static memory allocation) - Memory required by post-build configuration must be smaller than this constant. If parameter is disabled, the MCG calculates itself.

Multiplicity	0..1
Type	INTEGER
Configuration class	Link: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComRamSizeMax
Description	<p>This parameter defines the maximum number of values which can addressed in RAM.</p> <ul style="list-style-type: none"> ▶ INDEX_UINT8: uint8 is used as array index (RAM usage of the AUTOSAR COM module has to be smaller than 256 bytes) ▶ INDEX_UINT16: uint16 is used as array index (RAM usage of the AUTOSAR COM module has to be smaller than 65536 bytes) ▶ INDEX_UINT32: uint32 is used as array index (RAM usage of the AUTOSAR COM module has to be smaller than 4294967296 bytes) <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): The smaller the index the smaller the ROM consumption of the module configuration.
Multiplicity	1..1
Type	ENUMERATION
Default value	INDEX_UINT16
Range	INDEX_UINT8 INDEX_UINT16 INDEX_UINT32
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComCbktxTOutArraySizeMax
Description	<p>This parameter defines the maximum size of the array for Com_CbktxTOut callback functions (see also COM554). NOTE: if (ComTxTimeoutFactorSize == SIZE_0_BIT) this parameter has to be INDEX_NONE.</p> <ul style="list-style-type: none"> ▶ INDEX_NONE: the array is omitted and therefore Tx Deadline Monitoring is not supported (for all signals/signal groups). ▶ INDEX_UINT8: the reference to the array is 8 bit and the maximum number of callbacks is 0xFE.

	<p>► INDEX_UINT16: the reference to the array is 16 bit and the maximum number of callbacks is 0xFFFE.</p> <p>Optimization Effect:</p> <p>► ROM reduction (config): The smaller the index the smaller the ROM consumption of the module configuration.</p> <p>► ROM reduction (code): If INDEX_NONE is used the feature is disabled which reduces the ROM consumption of the module code.</p> <p>► Execution time reduction (code): If INDEX_NONE is used the feature is disabled which reduces the execution time of the module code.</p>	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	INDEX_UINT16	
Range	INDEX_NONE	
	INDEX_UINT8	
	INDEX_UINT16	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComCbkrxTOutArraySizeMax
Description	<p>This parameter defines the maximum size of the array for Com_CbkrxTOut callback functions (see also COM556). NOTE: if (ComRxTimeoutFactorSize == SIZE_0_BIT) this parameter has to be INDEX_NONE.</p> <p>► INDEX_NONE: the array is omitted and therefore Rx Deadline Monitoring is not supported (for all signals/signal groups).</p> <p>► INDEX_UINT8: the reference to the array is 8 bit and the maximum number of callbacks is 0xFE.</p> <p>► INDEX_UINT16: the reference to the array is 16 bit and the maximum number of callbacks is 0xFFFE.</p> <p>Optimization Effect:</p> <p>► ROM reduction (config): The smaller the index the smaller the ROM consumption of the module configuration.</p> <p>► ROM reduction (code): If INDEX_NONE is used the feature is disabled which reduces the ROM consumption of the module code.</p>

	<p>► Execution time reduction (code): If INDEX_NONE is used the feature is disabled which reduces the execution time of the module code.</p>	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	INDEX_UINT16	
Range	INDEX_NONE	
	INDEX_UINT8	
	INDEX_UINT16	
Configuration class	VariantPostBuild:	VariantPostBuild

Parameter Name	ComCbRxAckPtrArraySizeMax	
Description	<p>This parameter defines the maximum size of the array for Com_CbRxAck call-back functions (see also COM555).</p> <p>► INDEX_NONE: the array is omitted and therefore receive notification is not supported (for all signals/signal groups).</p> <p>► INDEX_UINT8: the reference to the array is 8 bit and the maximum number of callbacks is 0xFE.</p> <p>► INDEX_UINT16: the reference to the array is 16 bit and the maximum number of callbacks is 0xFFFE.</p> <p>Optimization Effect:</p> <p>► ROM reduction (config): The smaller the index the smaller the ROM consumption of the module configuration.</p> <p>► ROM reduction (code): If INDEX_NONE is used the feature is disabled which reduces the ROM consumption of the module code.</p> <p>► Execution time reduction (code): If INDEX_NONE is used the feature is disabled which reduces the execution time of the module code.</p>	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	INDEX_UINT16	
Range	INDEX_NONE	
	INDEX_UINT8	
	INDEX_UINT16	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	Elektrobit Automotive GmbH
---------------	----------------------------

Parameter Name	ComCbktxAckPtrArraySizeMax
Description	<p>This parameter defines the maximum size of the array for Com_CbktxAck call-back functions (see also COM468).</p> <ul style="list-style-type: none"> ▶ INDEX_NONE: the array is omitted and therefore transmit notification is not supported (for all signals/signal groups). ▶ INDEX_UINT8: the reference to the array is 8 bit and the maximum number of callbacks is 0xFE. ▶ INDEX_UINT16: the reference to the array is 16 bit and the maximum number of callbacks is 0xFFFE. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): The smaller the index the smaller the ROM consumption of the module configuration. ▶ ROM reduction (code): If INDEX_NONE is used the feature is disabled which reduces the ROM consumption of the module code. ▶ Execution time reduction (code): If INDEX_NONE is used the feature is disabled which reduces the execution time of the module code.
Multiplicity	1..1
Type	ENUMERATION
Default value	INDEX_UINT16
Range	INDEX_NONE INDEX_UINT8 INDEX_UINT16
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComCallOutFuncPtrArraySizeMax
Description	<p>This parameter defines the maximum number of Call-out function pointers in Com_RxCallouts and Com_TxCallouts.</p> <ul style="list-style-type: none"> ▶ INDEX_NONE: the array is omitted and therefore Pdu callouts are not supported. ▶ INDEX_UINT8: the reference to the array is 8 bit and the maximum number of Rx-Pdu callouts is 0xFE and Tx-Pdu callouts is 0xFE.

	<p>► INDEX_UINT16: the reference to the array is 16 bit and the maximum number of Rx-Pdu callouts is 0xFFFE and Tx-Pdu callouts is 0xFFFE.</p> <p>Optimization Effect:</p> <p>► ROM reduction (config): The smaller the index the smaller the ROM consumption of the module configuration.</p> <p>► ROM reduction (code): If INDEX_NONE is used the feature is disabled which reduces the ROM consumption of the module code.</p> <p>► Execution time reduction (code): If INDEX_NONE is used the feature is disabled which reduces the execution time of the module code.</p>	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	INDEX_UINT16	
Range	INDEX_NONE	
	INDEX_UINT8	
	INDEX_UINT16	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTriggerTxCallOutEnable	
Description	<p>Enables the configuration of callout for Com_TriggerTransmit() API (configuration parameter ComIPduTriggerTransmitCallout).</p> <p>► TRUE: The configuration parameter ComIPduTriggerTransmitCallout is available.</p> <p>► FALSE: The configuration parameter ComIPduTriggerTransmitCallout is not available.</p> <p>Optimization Effect:</p> <p>► ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration.</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComRxDataTimeoutAction
Description	<p>This parameter defines the action performed upon a reception of a timeout violation.</p> <ul style="list-style-type: none"> ▶ <code>RX_DATA_TIMEOUT_ACTION_NONE</code>: for all signals no replacement takes place. ▶ <code>RX_DATA_TIMEOUT_ACTION_REPLACE</code>: for all signals a replacement with <code>ComInitValue</code> takes place. ▶ <code>RX_DATA_TIMEOUT_ACTION_CONFIG</code>: for each signal the action can be defined. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): <code>RX_DATA_TIMEOUT_ACTION_NONE</code> and <code>RX_DATA_TIMEOUT_ACTION_REPLACE</code> may reduce the ROM consumption of the module configuration (depends on other features if a reduction can be achieved). ▶ ROM reduction (code): <code>RX_DATA_TIMEOUT_ACTION_NONE</code> removes code for feature; <code>RX_DATA_TIMEOUT_ACTION_REPLACE</code> disables feature partly, therefore these values of the parameter reduce the ROM consumption of the module code. ▶ Execution time reduction (code): <code>RX_DATA_TIMEOUT_ACTION_NONE</code> removes code for feature; <code>RX_DATA_TIMEOUT_ACTION_REPLACE</code> disables feature partly, therefore these values of the parameter reduce the execution time of the module code.
Multiplicity	1..1
Type	ENUMERATION
Default value	<code>RX_DATA_TIMEOUT_ACTION_CONFIG</code>
Range	<code>RX_DATA_TIMEOUT_ACTION_NONE</code> <code>RX_DATA_TIMEOUT_ACTION_REPLACE</code> <code>RX_DATA_TIMEOUT_ACTION_CONFIG</code>
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComRxTimeoutFactorSize
Description	<p>This parameter defines the size of <code>ComRxTimeoutFactor</code> for all Rx signals / group signals.</p> <ul style="list-style-type: none"> ▶ <code>SIZE_0_BIT</code>: the parameter <code>ComRxTimeoutFactor</code> is not available.

	<ul style="list-style-type: none"> ▶ SIZE_8_BIT: the parameter ComRxTimeoutFactor is a 8 bit value. ▶ SIZE_16_BIT: the parameter ComRxTimeoutFactor is a 16 bit value. ▶ SIZE_32_BIT: the parameter ComRxTimeoutFactor is a 32 bit value. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): The smaller the size the smaller the ROM consumption of the module configuration. ▶ ROM reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the ROM consumption of the module code. ▶ Execution time reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	SIZE_16_BIT	
Range	SIZE_0_BIT	
	SIZE_8_BIT	
	SIZE_16_BIT	
	SIZE_32_BIT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComRxFirstTimeoutFactorSize
Description	<p>This parameter defines the size of ComRxFirstTimeoutFactor for all Rx signals / group signals.</p> <ul style="list-style-type: none"> ▶ SIZE_0_BIT: the parameter ComRxFirstTimeoutFactor is not available. ▶ SIZE_8_BIT: the parameter ComRxFirstTimeoutFactor is a 8 bit value. ▶ SIZE_16_BIT: the parameter ComRxFirstTimeoutFactor is a 16 bit value. ▶ SIZE_32_BIT: the parameter ComRxFirstTimeoutFactor is a 32 bit value. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): The smaller the size the smaller the ROM consumption of the module configuration. ▶ ROM reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the ROM consumption of the module code.

	<p>► Execution time reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the execution time of the module code.</p>	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	SIZE_16_BIT	
Range	SIZE_0_BIT	
	SIZE_8_BIT	
	SIZE_16_BIT	
	SIZE_32_BIT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTxTimeoutFactorSize	
Description	<p>This parameter defines the size of ComTxTimeoutFactor for all Tx signals / group signals.</p> <ul style="list-style-type: none"> ► SIZE_0_BIT: the parameter ComTxTimeoutFactor is not available. ► SIZE_8_BIT: the parameter ComTxTimeoutFactor is a 8 bit value. ► SIZE_16_BIT: the parameter ComTxTimeoutFactor is a 16 bit value. ► SIZE_32_BIT: the parameter ComTxTimeoutFactor is a 32 bit value. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ► ROM reduction (config): The smaller the size the smaller the ROM consumption of the module configuration. ► ROM reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the ROM consumption of the module code. ► Execution time reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	SIZE_16_BIT	
Range	SIZE_0_BIT	
	SIZE_8_BIT	
	SIZE_16_BIT	
	SIZE_32_BIT	

Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTxModeRepetitionPeriodFactorS	
Description	<p>This parameter defines the size of ComTxModeRepetitionPeriodFactor.</p> <ul style="list-style-type: none"> ▶ SIZE_0_BIT: the parameter ComTxModeRepetitionPeriodFactor is not available and therefore the transmission modes "Direct/NTimes" and "Mixed". ▶ SIZE_8_BIT: the parameter ComTxModeRepetitionPeriodFactor is a 8 bit value. ▶ SIZE_16_BIT: the parameter ComTxModeRepetitionPeriodFactor is a 16 bit value. ▶ SIZE_32_BIT: the parameter ComTxModeRepetitionPeriodFactor is a 32 bit value. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): The smaller the size the smaller the ROM consumption of the module configuration. ▶ ROM reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the ROM consumption of the module code. ▶ Execution time reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	SIZE_16_BIT	
Range	SIZE_0_BIT	
	SIZE_8_BIT	
	SIZE_16_BIT	
	SIZE_32_BIT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTxModeTimeOffsetFactorSize	
Description	<p>This parameter defines the size of ComTxModeTimeOffsetFactor.</p> <ul style="list-style-type: none"> ▶ SIZE_0_BIT: the parameter ComTxModeTimeOffsetFactor is not available and equal to 0. 	

	<ul style="list-style-type: none"> ▶ SIZE_8_BIT: the parameter ComTxModeTimeOffsetFactor is a 8 bit value. ▶ SIZE_16_BIT: the parameter ComTxModeTimeOffsetFactor is a 16 bit value. ▶ SIZE_32_BIT: the parameter ComTxModeTimeOffsetFactor is a 32 bit value. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): The smaller the size the smaller the ROM consumption of the module configuration. ▶ ROM reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the ROM consumption of the module code. ▶ Execution time reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	SIZE_16_BIT	
Range	SIZE_0_BIT	
	SIZE_8_BIT	
	SIZE_16_BIT	
	SIZE_32_BIT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTxModeTimePeriodFactorSize
Description	<p>This parameter defines the size of ComTxModeTimePeriodFactor.</p> <ul style="list-style-type: none"> ▶ SIZE_0_BIT: the parameter ComTxModeTimePeriodFactor is not available and therefore the transmission modes "Periodic" and "Mixed". ▶ SIZE_8_BIT: the parameter ComTxModeTimePeriodFactor is a 8 bit value. ▶ SIZE_16_BIT: the parameter ComTxModeTimePeriodFactor is a 16 bit value. ▶ SIZE_32_BIT: the parameter ComTxModeTimePeriodFactor is a 32 bit value. <p>Optimization Effect:</p>

	<ul style="list-style-type: none"> ▶ ROM reduction (config): The smaller the size the smaller the ROM consumption of the module configuration. ▶ ROM reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the ROM consumption of the module code. ▶ Execution time reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	SIZE_16_BIT	
Range	SIZE_0_BIT	
	SIZE_8_BIT	
	SIZE_16_BIT	
	SIZE_32_BIT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTxIPduMDTFactorSize
Description	<p>This parameter defines the size of ComTxIPduMinimumDelayTimeFactor.</p> <ul style="list-style-type: none"> ▶ SIZE_0_BIT: the parameter ComTxIPduMinimumDelayTimeFactor is not available. ▶ SIZE_8_BIT: the parameter ComTxIPduMinimumDelayTimeFactor is a 8 bit value. ▶ SIZE_16_BIT: the parameter ComTxIPduMinimumDelayTimeFactor is a 16 bit value. ▶ SIZE_32_BIT: the parameter ComTxIPduMinimumDelayTimeFactor is a 32 bit value. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): The smaller the size the smaller the ROM consumption of the module configuration. ▶ ROM reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the ROM consumption of the module code. ▶ Execution time reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the execution time of the module code.
Multiplicity	1..1

Type	ENUMERATION	
Default value	SIZE_16_BIT	
Range	SIZE_0_BIT	
	SIZE_8_BIT	
	SIZE_16_BIT	
	SIZE_32_BIT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComUpdateBitRxConfig	
Description	<p>This parameter defines the update bit behavior on receiver side.</p> <ul style="list-style-type: none"> ▶ UPDATE_BIT_ABSENT_FOR_ALL: Optimization is switched on, update bits are not supported on Rx side. ▶ UPDATE_BIT_PRESENT_FOR_ALL: Optimization is switched on, update bits are configured for all signals / signal groups on Rx side. ▶ UPDATE_BIT_INDIVIDUAL: Optimization is switched off, presents of update bits can be configured individually on Rx side. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): UPDATE_BIT_ABSENT_FOR_ALL reduces the ROM consumption of the module configuration, UPDATE_BIT_PRESENT_FOR_ALL may reduce the ROM consumption of the module configuration. ▶ ROM reduction (code): UPDATE_BIT_ABSENT_FOR_ALL reduces the ROM consumption of the module code, UPDATE_BIT_PRESENT_FOR_ALL reduces the ROM consumption of the module code slightly. ▶ Execution time reduction (code): UPDATE_BIT_ABSENT_FOR_ALL reduces the execution time of the module code, UPDATE_BIT_PRESENT_FOR_ALL reduces the execution time of the module code slightly. 	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	UPDATE_BIT_INDIVIDUAL	
Range	UPDATE_BIT_ABSENT_FOR_ALL	
	UPDATE_BIT_PRESENT_FOR_ALL	
	UPDATE_BIT_INDIVIDUAL	

Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComUpdateBitTxConfig	
Description	<p>This parameter defines the update bit behavior on sender side.</p> <ul style="list-style-type: none"> ▶ UPDATE_BIT_ABSENT_FOR_ALL: Optimization is switched on, update bits are not supported on Tx side. ▶ UPDATE_BIT_PRESENT_FOR_ALL: Optimization is switched on, update bits are configured for all signals / signal groups on Tx side. ▶ UPDATE_BIT_INDIVIDUAL: Optimization is switched off, presents of update bits can be configured individually on Tx side. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): UPDATE_BIT_ABSENT_FOR_ALL reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): UPDATE_BIT_ABSENT_FOR_ALL reduces the ROM consumption of the module code, UPDATE_BIT_PRESENT_FOR_ALL reduces the ROM consumption of the module code slightly. ▶ Execution time reduction (code): UPDATE_BIT_ABSENT_FOR_ALL reduces the execution time of the module code, UPDATE_BIT_PRESENT_FOR_ALL reduces the execution time of the module code slightly. 	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	UPDATE_BIT_INDIVIDUAL	
Range	UPDATE_BIT_ABSENT_FOR_ALL	
	UPDATE_BIT_PRESENT_FOR_ALL	
	UPDATE_BIT_INDIVIDUAL	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComClearUpdateBitTxTransmitEnable	
Description	<p>This parameter enables / disables clearing of update bits after a call to PduR_ComTransmit.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, clearing of update bits after a call to PduR_ComTransmit is enabled. 	

	<p>▶ FALSE: Optimization is switched on, update bits are never cleared after a call to PduR_ComTransmit.</p> <p>Optimization Effect:</p> <p>▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code.</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComClearUpdateBitTxTriggerTransmitEnable	
Description	<p>This parameter enables / disables clearing of update bits during a call to Com_TriggerTransmit.</p> <p>▶ TRUE: Optimization is switched off, clearing of update bits during a call to Com_TriggerTransmit is enabled.</p> <p>▶ FALSE: Optimization is switched on, update bits are never cleared during a call to Com_TriggerTransmit.</p> <p>Optimization Effect:</p> <p>▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code.</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComClearUpdateBitTxTxConfirmationEnable	
Description	<p>This parameter enables / disables clearing of update bits during a call to Com_TxConfirmation.</p> <p>▶ TRUE: Optimization is switched off, clearing of update bits during a call to Com_TxConfirmation is enabled.</p> <p>▶ FALSE: Optimization is switched on, update bits are never cleared during a call to Com_TxConfirmation.</p>	

	Optimization Effect:	
	<ul style="list-style-type: none"> ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTmsEnable	
Description	<p>This parameter enables / disables the Transmission Mode Selection (TMS).</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, TMS is enabled. ▶ FALSE: Optimization is switched on, TMS is disabled (change between TM is not supported). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Enabling this optimization reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComFilterReceiverEnable	
Description	<p>This parameter enables / disables filtering on receiver side.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, filtering on receiver side is enabled. ▶ FALSE: Optimization is switched on, filtering on receiver side is disabled for all signals. 	

	Optimization Effect: <ul style="list-style-type: none"> ▶ ROM reduction (config): Enabling this optimization reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTxDynLengthIPduEnable	
Description	<p>This parameter enables / disables features related to the variable length of an Tx-IPdu. This is required for dynamic length signal support.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, length of Tx-IPdu can vary. ▶ FALSE: Optimization is switched on, length of Rx-IPdu is fix as configured in EcuC. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ RAM reduction (code): Enabling this optimization reduces the RAM consumption of the module configuration. ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComRxDynLengthIPduEnable	
Description	<p>This parameter enables / disables features related to the variable length of an Rx-IPdu. This is required for dynamic length signal support.</p>	

	<ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, length of Rx-IPdu can vary as provided by the lower layer. ▶ FALSE: Optimization is switched on, length of Rx-IPdu is fix as configured in EcuC. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ RAM reduction (code): Enabling this optimization reduces the RAM consumption of the module configuration. ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComFilterOneEveryNPeriodOffSMax
Description	<p>This parameter defines the size of the parameter ComFilterOffset and ComFilterPeriodFactor of the filter OneEveryN. Note, the value of the parameter ComFilterOneEveryNPeriodOffSMax and ComFilterOneEveryNOccuranceMax have to be equal.</p> <ul style="list-style-type: none"> ▶ SIZE_0_BIT: the filter OneEveryN is not supported. ▶ SIZE_8_BIT: the maximum value of the parameter ComFilterOffset and ComFilterPeriodFactor is a 8 bit value. ▶ SIZE_16_BIT: the maximum value of the parameter ComFilterOffset and ComFilterPeriodFactor is a 16 bit value. ▶ SIZE_32_BIT: the maximum value of the parameter ComFilterOffset and ComFilterPeriodFactor is a 32 bit value. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): The smaller the size the smaller the ROM consumption of the module configuration. ▶ ROM reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the ROM consumption of the module code. ▶ Execution time reduction (code): If SIZE_0_BIT is used the feature is disabled which reduces the execution time of the module code.

Multiplicity	1..1
Type	ENUMERATION
Default value	SIZE_16_BIT
Range	SIZE_0_BIT
	SIZE_8_BIT
	SIZE_16_BIT
	SIZE_32_BIT
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComFilterOneEveryNOccuranceMax
Description	<p>This parameter defines the size of internal parameter 'occurance' (stored in RAM) of the filter OneEveryN. Note, the value of the parameter ComFilterOneEveryNPeriodOffSMax and ComFilterOneEveryNOccuranceMax have to be equal.</p> <ul style="list-style-type: none"> ▶ SIZE_0_BIT: the filter OneEveryN is not supported. ▶ SIZE_8_BIT: the maximum value of the parameter occurance is a 8 bit value. ▶ SIZE_16_BIT: the maximum value of the parameter occurance is a 16 bit value. ▶ SIZE_32_BIT: the maximum value of the parameter occurance is a 32 bit value. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ RAM reduction (config): The smaller the size the smaller the RAM consumption of the module configuration.
Multiplicity	1..1
Type	ENUMERATION
Default value	SIZE_16_BIT
Range	SIZE_0_BIT
	SIZE_8_BIT
	SIZE_16_BIT
	SIZE_32_BIT
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComTxModeDirectEnable
Description	<p>This parameter enables / disables the transmission mode Direct (parameter ComTxModeMode = DIRECT, ComTxModeNumberOfRepetitions = 0).</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, the transmission mode is supported. ▶ FALSE: Optimization is switched on, the transmission mode is not supported. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComTxModeNTimesEnable
Description	<p>This parameter enables / disables the transmission mode N-Times (parameter ComTxModeMode = DIRECT, ComTxModeNumberOfRepetitions > 0).</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, the transmission mode is supported. ▶ FALSE: Optimization is switched on, the transmission mode is not supported. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComTxModePeriodicEnable
Description	<p>This parameter enables / disables the transmission mode Periodic.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, the transmission mode is supported. ▶ FALSE: Optimization is switched on, the transmission mode is not supported. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComTxModeMixedDirectEnable
Description	<p>This parameter enables / disables the transmission mode Mixed/Direct (parameter ComTxModeMode = MIXED, ComTxModeNumberOfRepetitions = 0).</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, the transmission mode is supported. ▶ FALSE: Optimization is switched on, the transmission mode is not supported. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComTxModeMixedNTimesEnable
Description	<p>This parameter enables / disables the transmission mode Mixed/N-Times (parameter ComTxModeMode = MIXED, ComTxModeNumberOfRepetitions > 0).</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, the transmission mode is supported. ▶ FALSE: Optimization is switched on, the transmission mode is not supported. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComTxSigConfDeferredEnable
Description	<p>This parameter enables / disables the deferred Tx confirmation.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, deferred confirmation is supported. ▶ FALSE: Optimization is switched on, deferred confirmation is not supported. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComTxSigConfImmediateEnable
Description	<p>This parameter enables / disables the immediate Tx confirmation.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, immediate confirmation is supported. ▶ FALSE: Optimization is switched on, immediate confirmation is not supported. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComRxSigConfDeferredEnable
Description	<p>This parameter enables / disables the deferred Rx confirmation.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, deferred confirmation is supported. ▶ FALSE: Optimization is switched on, deferred confirmation is not supported. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComRxSigConfImmediateEnable
----------------	-----------------------------

Description	<p>This parameter enables / disables the immediate Rx confirmation.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, immediate confirmation is supported. ▶ FALSE: Optimization is switched on, immediate confirmation is not supported. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code; , if both ComTmsEnable and ComFilterReceiverEnable are disabled, additionally no code for filtering is included. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComSignalGwEnable	
Description	<p>This parameter enables / disables the signal gateway.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, signal gateway is enabled. ▶ FALSE: Optimization is switched on, signal gateway is disabled. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Enabling this optimization reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code. ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComCheckValueSizeEnable	
-----------------------	--------------------------------	--

Description	This is an EB extension to the AUTOSAR specification. It is an additional check in the functions Com_SendSignal() / Com_UpdateShadowSignal(). The check verifies if the value provided by an application fits into the configured size of the signal / group signal. If the value does not fit into the signal / group signal it is reported to DET. If ComCheckValueSizeEnable is set to TRUE the check is enabled, otherwise disabled. In case ComReportToDetEnable == FALSE, the configuration of ComCheckValueSizeEnable is ignored. The number of bits of the value of a signal / group signal copied into the I-Pdu is the number of bits, which are configured for the signal / group signal, independent from the configuration of ComCheckValueSizeEnable.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComConstCfgAddressEnable	
Description	<p>This parameter defines if a constant starting address for the configuration of the module is used. The fix address has to be configured with ComConstCfgAddress.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched on, configuration is placed on the configured address. ▶ FALSE: Optimization is switched off, the starting address of the configuration has to be provided for the function Com_Init(). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): Enabling this optimization reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComConstCfgAddress	
Description	Only valid if ComConstCfgAddressEnable == TRUE. Defines the fix address where the configuration starts.	
Multiplicity	1..1	

Type	INTEGER	
Default value	0	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComRelocatableCfgEnable	
Description	<p>Enables/disable support for relocatable postbuild configuration.</p> <ul style="list-style-type: none"> ▶ True: Postbuild configuration relocatable in memory. ▶ False: Postbuild configuration not relocatable in memory. <p>Note: If PbcfgM support is enabled for Com, this feature is managed by by the parameter PbcfgMRelocatableCfgEnable.</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	Com_TxF_MaskNewDiffersMaskOld_En	
Description	<p>This parameter defines if the filter MaskedNewDiffersMaskOld is available for Tx (group) signals. Only valid if ComTmsEnable == TRUE.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, filter for is Tx (group) signals is supported. ▶ FALSE: Optimization is switched on, filter for is Tx (group) signals is not supported. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this optimization reduces the ROM consumption of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComSigGwRxFilterEnable	
----------------	-------------------------------	--

Description	<p>This parameter defines if the Rx filter is applied to decide if the signal is gated. NOTE: Parameter is only valid/enabled if signal gateway (ComSignalGwEnable) and Rx Filter is enabled (ComFilterReceiverEnable).</p> <ul style="list-style-type: none"> ▶ TRUE: Filtering of gated signals is switched on. ▶ FALSE: Filtering of gated signals is switched off. (as defined in the SWS AUTOSAR COM 3.x). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ RAM reduction (config): Disabling this feature reduces RAM consumption of the module configuration. ▶ ROM reduction (config): Disabling this feature reduces ROM consumption of the module configuration. ▶ ROM reduction (code): Disabling this feature reduces the ROM consumption of the module code. ▶ Execution time reduction (code): Disabling this feature reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTransfPropTriggeredEnable
Description	<p>This parameter defines if the transfer property TRIGGERED of Tx signals is available.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, signal with transfer property TRIGGERED can be configured. ▶ FALSE: Optimization is switched on, no signal with transfer property TRIGGERED can be configured. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): If set to FALSE the feature is disabled which reduces the execution time of the module code. ▶ ROM reduction (code): If set to FALSE the feature is disabled which reduces the ROM consumption of the module code.
Multiplicity	1..1

Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTransfPropTriggeredOCEnable	
Description	<p>This parameter defines if the transfer property TRIGGERED_ON_CHANGE of Tx signals is available.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, signal with transfer property TRIGGERED_ON_CHANGE can be configured. ▶ FALSE: Optimization is switched on, no signal with transfer property TRIGGERED_ON_CHANGE can be configured. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): If set to FALSE the feature is disabled which reduces the execution time of the module code. ▶ ROM reduction (code): If set to FALSE the feature is disabled which reduces the ROM consumption of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTransfPropWithoutRepEnable	
Description	<p>This parameter defines if the transfer properties _WITHOUT_REPETITION [TRIGGERED_ON_CHANGE_WITHOUT_REPETITION, TRIGGERED_WITHOUT_REPETITION] of Tx signals are available.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched off, signal with transfer properties _WITHOUT_REPETITION can be configured. ▶ FALSE: Optimization is switched on, no signal with transfer properties _WITHOUT_REPETITION can be configured. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): If set to FALSE the feature is disabled which reduces the execution time of the module code. 	

	► ROM reduction (code): If set to FALSE the feature is disabled which reduces the ROM consumption of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComRxTpAPIEnable
Description	<p>This parameter enables / disables Com Rx Tp support.</p> <ul style="list-style-type: none"> ► TRUE: Optimization is switched off, no Rx Pdu with ComIPduType TP can be configured. ► FALSE: Optimization is switched on, Rx Pdu with ComIPduType TP can be configured. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ► Execution time reduction (code): If set to FALSE the feature is disabled which reduces the execution time of the module code. ► RAM reduction (code): If set to FALSE the feature is disabled which reduces the RAM consumption of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComTxTpAPIEnable
Description	<p>This parameter enables / disables Com Tx Tp support.</p> <ul style="list-style-type: none"> ► TRUE: Optimization is switched off, no Tx Pdu with ComIPduType TP can be configured. ► FALSE: Optimization is switched on, Tx Pdu with ComIPduType TP can be configured. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ► Execution time reduction (code): If set to FALSE the feature is disabled which reduces the execution time of the module code.

	► RAM reduction (code): If set to FALSE the feature is disabled which reduces the RAM consumption of the module code.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTxZeroSignalEnable	
Description	<p>This parameter defines if the zero size Tx signals can be configured.</p> <ul style="list-style-type: none"> ► TRUE: Optimization is switched off, Tx signal with size zero can be configured. ► FALSE: Optimization is switched on, no Tx signal with size zero can be configured. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ► Execution time reduction (code): If set to FALSE the feature is disabled which reduces the execution time of the module code. ► ROM reduction (code): If set to FALSE the feature is disabled which reduces the ROM consumption of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComBasedTransformerSupportTx	
Description	<p>This parameter defines if support for the Tx Com based transformer is available. Note: this parameter is only enabled when ComEnableSignalGroupArrayApi == true.</p> <ul style="list-style-type: none"> ► TRUE: Optimization is switched on, a Tx signal group can also be accessed with Com based transformer concept. ► FALSE: Optimization is switched off, signal group update and sending only possible using the APIs Com_UpdateShadowSignal and Com_SendSignalGroup. 	

	Optimization Effect:	
	<ul style="list-style-type: none"> ▶ Execution time reduction (code): If set to TRUE the feature is enabled which reduces the execution time of the module code. (only if Rte supports feature) 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComBasedTransformerSupportRx	
Description	<p>This parameter defines if support for the Rx Com based transformer is available. Note: this parameter is only enabled when ComEnableSignalGroupArrayApi == true.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched on, a Rx signal group can also be accessed with Com based transformer concept. ▶ FALSE: Optimization is switched off, access to a signal group is only possible via Com_ReceiveSignalGroup and Com_ReceiveShadowSignal. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): If set to TRUE the feature is enabled which reduces the execution time of the module code (only if Rte supports feature) 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	ComTxGroupSignalNoLock	
Description	<p>This parameter defines if the shadow buffer of a Tx signal group shall be locked during the update of a group signal. With the typical AUTOSAR use case (Rte updates all group signals sequentially and calls afterwards Com_SendSignalGroup) the locking can be disabled.</p> <ul style="list-style-type: none"> ▶ TRUE: Optimization is switched on, shadow buffers of Tx signal groups are not locked during the update of a group signal. 	

	<p>► FALSE: Optimization is switched off, shadow buffers of Tx signal groups are locked during the update of a group signal.</p> <p>Optimization Effect:</p> <p>► Execution time reduction (code): If set to TRUE the feature is enabled which reduces the execution time of the module code.</p>
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComDeferTx2MainFunc
Description	<p>This parameter defines if transmission request from Com module are only issued from Com transmission main function or also from other Com APIs.</p> <p>► TRUE: Transmission request from Com module are only issued from Com transmission main function.</p> <p>► FALSE: Transmission request from Com module are also issued from other APIs than Com transmission main function.</p>
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

4.2.1.32. ComGeneratedRxSignal

Parameters included	
Parameter name	Multiplicity
ComRcvRxSigLockGenerated	1..1
ComGeneratedRcvSigEnable	1..1
ComMapReceiveSignal	1..1

Parameters included	
ComRcvSig-MacroExtPrefix	0..1

Parameter Name	ComRcvRxSigLockGenerated
Description	If set to TRUE the reading of Rx-signal with the generated Com_ReceiveSignal() API is protected with the critical section (SCHM_COM_EXCLUSIVE_AREA_0) as configured in SchM. NOTE: this configuration applies for the generated macros and functions.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComGeneratedRcvSigEnable
Description	If set to STD_ON a function is generated which extracts the values of the signals. The name of the function is either Com_ReceiveSignalGenerated() or Com_ReceiveSignal() depending on the configuration of ComMapReceiveSignal.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComMapReceiveSignal
Description	Defines which function is used when Com_ReceiveSignal is called.
Multiplicity	1..1
Type	ENUMERATION
Default value	Com_ReceiveSignalGenerated
Range	Com_ReceiveSignalGenerated Com_ReceiveSignalGeneric
Configuration class	PreCompile: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	ComRcvSigMacroExtPrefix
----------------	-------------------------

Description	<p>The macro COM_RECEIVE_SIGNAL_<signalId> will be mapped to the macro defined here. The following macros will be generated:</p> <ul style="list-style-type: none"> ▶ <ComRcvSigMacroExtPrefix><signalShortName> ▶ <ComRcvSigMacroExtPrefix><signalId> ▶ <ComRcvSigMacroExtPrefix><signalIdU> ▶ <ComRcvSigMacroExtPrefix><signalIdu> 	
Multiplicity	0..1	
Type	FUNCTION-NAME	
Configuration class	PreCompile:	VariantPostBuild
Origin	AUTOSAR_ECUC V1.0.0	

4.2.1.33. CommonPublishedInformation

Parameters included	
Parameter name	Multiplicity
ArMajorVersion	1..1
ArMinorVersion	1..1
ArPatchVersion	1..1
SwMajorVersion	1..1
SwMinorVersion	1..1
SwPatchVersion	1..1
ModuleId	1..1
VendorId	1..1
Release	1..1

Parameter Name	ArMajorVersion
Label	AUTOSAR Major Version
Description	Major version number of AUTOSAR specification on which the appropriate implementation is based on.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	4
Configuration class	PublishedInformation:

Origin	Elektrobit Automotive GmbH
---------------	----------------------------

Parameter Name	ArMinorVersion	
Label	AUTOSAR Minor Version	
Description	Minor version number of AUTOSAR specification on which the appropriate implementation is based on.	
Multiplicity	1..1	
Type	INTEGER_LABEL	
Default value	2	
Configuration class	PublishedInformation:	
Origin	Elektrobit Automotive GmbH	

Parameter Name	ArPatchVersion	
Label	AUTOSAR Patch Version	
Description	Patch level version number of AUTOSAR specification on which the appropriate implementation is based on.	
Multiplicity	1..1	
Type	INTEGER_LABEL	
Default value	0	
Configuration class	PublishedInformation:	
Origin	Elektrobit Automotive GmbH	

Parameter Name	SwMajorVersion	
Label	Software Major Version	
Description	Major version number of the vendor specific implementation of the module.	
Multiplicity	1..1	
Type	INTEGER_LABEL	
Default value	6	
Configuration class	PublishedInformation:	
Origin	Elektrobit Automotive GmbH	

Parameter Name	SwMinorVersion	
Label	Software Minor Version	
Description	Minor version number of the vendor specific implementation of the module. The numbering is vendor specific.	

Multiplicity	1..1
Type	INTEGER_LABEL
Default value	3
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	SwPatchVersion
Label	Software Patch Version
Description	Patch level version number of the vendor specific implementation of the module. The numbering is vendor specific.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	37
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	ModuleId
Label	Numeric Module ID
Description	Module ID of this module from Module List
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	50
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	VendorId
Label	Vendor ID
Description	Vendor ID of the dedicated implementation of this module according to the AUTOSAR vendor list
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	1
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	Release
Label	Release Information
Multiplicity	1..1
Type	STRING_LABEL
Default value	
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

4.2.1.34. PublishedInformation

Parameters included	
Parameter name	Multiplicity
PbcfgMSupport	1..1

Parameter Name	PbcfgMSupport
Label	PbcfgM support
Description	Specifies whether or not the Com can use the PbcfgM module for post-build support.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

4.2.2. Recommended configurations

4.2.2.1. ComRecConfigurationStandard

Containers included	
Container name	Container definition

Containers included	
ComGeneral	ComGeneral
Parameters included	
Parameter name	Value

4.2.2.1.1. ComGeneral

Containers included	
Container name	Container definition
VendorSpecific	ComGeneral VendorSpecific
Parameters included	
Parameter name	Value
ComConfigurationUseDet	true
ComVersionInfoApi	true

4.2.2.1.2. VendorSpecific

Parameters included	
Parameter name	Value
ComDataMemSize	(DISABLED)
ComRamSizeMax	INDEX_UINT16
ComCbktxOutArraySizeMax	INDEX_UINT16
ComCbkrxOutArraySizeMax	INDEX_UINT16
ComCbkrxAckPtrArraySizeMax	INDEX_UINT16
ComCbktxAckPtrArraySizeMax	INDEX_UINT16
ComCallOutFuncPtrArraySizeMax	INDEX_UINT16
ComTriggerTxCallOutEnable	true
ComRxDataTimeoutAction	RX_DATA_TIMEOUT_ACTION_CONFIG
ComRxTimeoutFactorSize	SIZE_16_BIT
ComRxFirstTimeoutFactorSize	SIZE_16_BIT
ComTxTimeoutFactorSize	SIZE_16_BIT
ComTxModeRepetitionPeriodFactorS	SIZE_16_BIT

Parameters included	
ComTxModeTimeOffsetFactorSize	SIZE_16_BIT
ComTxModeTimePeriodFactorSize	SIZE_16_BIT
ComTxlpduMDTFactorSize	SIZE_16_BIT
ComUpdateBitRxConfig	UPDATE_BIT_INDIVIDUAL
ComUpdateBitTxConfig	UPDATE_BIT_INDIVIDUAL
ComTmsEnable	true
ComFilterReceiverEnable	true
ComTxDynLengthIPduEnable	true
ComRxDynLengthIPduEnable	true
ComFilterOneEveryNPeriodOffSMax	SIZE_16_BIT
ComFilterOneEveryNOccuranceMax	SIZE_16_BIT
ComTxModeDirectEnable	true
ComTxModeNTimesEnable	true
ComTxModePeriodicEnable	true
ComTxModeMixedDirectEnable	true
ComTxModeMixedNTimesEnable	true
ComTxSigConfDeferredEnable	true
ComTxSigConfImmediateEnable	true
ComRxSigConfDeferredEnable	true
ComRxSigConfImmediateEnable	true
ComSignalGwEnable	true
ComCheckValueSizeEnable	false
ComConstCfgAddressEnable	false
ComConstCfgAddress	0
ComRelocatableCfgEnable	true
Com_TxF_MaskNewDiffersMaskOld_En	true

4.2.2.2. ComRecConfigurationSmall

Containers included	
Container name	Container definition

Containers included	
ComGeneral	ComGeneral
Parameters included	
Parameter name	Value

4.2.2.2.1. ComGeneral

Containers included	
Container name	Container definition
VendorSpecific	ComGeneral VendorSpecific
Parameters included	
Parameter name	Value
ComConfigurationUseDet	false
ComVersionInfoApi	false

4.2.2.2.2. VendorSpecific

Parameters included	
Parameter name	Value
ComDataMemSize	(DISABLED)
ComRamSizeMax	INDEX_UINT8
ComCbktxTOutArraySizeMax	INDEX_NONE
ComCbkrxTOutArraySizeMax	INDEX_NONE
ComCbkrxAckPtrArraySizeMax	INDEX_UINT8
ComCbktxAckPtrArraySizeMax	INDEX_NONE
ComCallOutFuncPtrArraySizeMax	INDEX_NONE
ComTriggerTxCallOutEnable	false
ComRxDataTimeoutAction	RX_DATA_TIMEOUT_ACTION_NONE
ComRxTimeoutFactorSize	SIZE_0_BIT
ComRxFirstTimeoutFactorSize	SIZE_0_BIT
ComTxTimeoutFactorSize	SIZE_0_BIT
ComTxModeRepetitionPeriodFactorS	SIZE_8_BIT

Parameters included	
ComTxModeTimeOffsetFactorSize	SIZE_8_BIT
ComTxModeTimePeriodFactorSize	SIZE_8_BIT
ComTxIpduMDTFactorSize	SIZE_0_BIT
ComUpdateBitRxConfig	UPDATE_BIT_ABSENT_FOR_ALL
ComUpdateBitTxConfig	UPDATE_BIT_ABSENT_FOR_ALL
ComTmsEnable	false
ComFilterReceiverEnable	false
ComTxDynLengthIpduEnable	false
ComRxDynLengthIpduEnable	false
ComFilterOneEveryNPeriodOffSMax	SIZE_0_BIT
ComFilterOneEveryNOccuranceMax	SIZE_0_BIT
ComTxModeDirectEnable	true
ComTxModeNTimesEnable	true
ComTxModePeriodicEnable	true
ComTxModeMixedDirectEnable	false
ComTxModeMixedNTimesEnable	false
ComTxSigConfDeferredEnable	false
ComTxSigConfImmediateEnable	true
ComRxSigConfDeferredEnable	false
ComRxSigConfImmediateEnable	true
ComSignalGwEnable	false
ComCheckValueSizeEnable	false
ComConstCfgAddressEnable	false
ComConstCfgAddress	0
ComRelocatableCfgEnable	true
Com_TxF_MaskNewDiffersMaskOld_En	true

4.2.2.3. ComRecConfigurationMedium

Containers included	
Container name	Container definition

Containers included	
ComGeneral	ComGeneral

Parameters included	
Parameter name	Value

4.2.2.3.1. ComGeneral

Containers included	
Container name	Container definition
VendorSpecific	ComGeneral VendorSpecific

Parameters included	
Parameter name	Value
ComConfigurationUseDet	true
ComVersionInfoApi	true

4.2.2.3.2. VendorSpecific

Parameters included	
Parameter name	Value
ComDataMemSize	(DISABLED)
ComRamSizeMax	INDEX_UINT16
ComCbktxTOutArraySizeMax	INDEX_UINT16
ComCbkrxTOutArraySizeMax	INDEX_UINT16
ComCbkrxAckPtrArraySizeMax	INDEX_UINT16
ComCbktxAckPtrArraySizeMax	INDEX_UINT16
ComCallOutFuncPtrArraySizeMax	INDEX_UINT16
ComTriggerTxCallOutEnable	true
ComRxDataTimeoutAction	RX_DATA_TIMEOUT_ACTION_CONFIG
ComRxTimeoutFactorSize	SIZE_16_BIT
ComRxFirstTimeoutFactorSize	SIZE_16_BIT
ComTxTimeoutFactorSize	SIZE_16_BIT

Parameters included	
ComTxModeRepetitionPeriodFactorS	SIZE_16_BIT
ComTxModeTimeOffsetFactorSize	SIZE_16_BIT
ComTxModeTimePeriodFactorSize	SIZE_16_BIT
ComTxIpduMDTFactorSize	SIZE_16_BIT
ComUpdateBitRxConfig	UPDATE_BIT_ABSENT_FOR_ALL
ComUpdateBitTxConfig	UPDATE_BIT_ABSENT_FOR_ALL
ComTmsEnable	true
ComFilterReceiverEnable	true
ComTxDynLengthIPduEnable	true
ComRxDynLengthIPduEnable	true
ComFilterOneEveryNPeriodOffSMax	SIZE_16_BIT
ComFilterOneEveryNOccuranceMax	SIZE_16_BIT
ComTxModeDirectEnable	true
ComTxModeNTimesEnable	true
ComTxModePeriodicEnable	true
ComTxModeMixedDirectEnable	true
ComTxModeMixedNTimesEnable	true
ComTxSigConfDeferredEnable	true
ComTxSigConfImmediateEnable	true
ComRxSigConfDeferredEnable	true
ComRxSigConfImmediateEnable	true
ComSignalGwEnable	false
ComCheckValueSizeEnable	false
ComConstCfgAddressEnable	false
ComConstCfgAddress	0
ComRelocatableCfgEnable	true

4.2.2.4. ComRecConfigurationMax

Containers included	
Container name	Container definition

Containers included	
ComGeneral	ComGeneral

Parameters included	
Parameter name	Value

4.2.2.4.1. ComGeneral

Containers included	
Container name	Container definition
VendorSpecific	ComGeneral VendorSpecific

Parameters included	
Parameter name	Value
ComConfigurationUseDet	true
ComVersionInfoApi	true

4.2.2.4.2. VendorSpecific

Parameters included	
Parameter name	Value
ComDataMemSize	(DISABLED)
ComRamSizeMax	INDEX_UINT16
ComCbktxTOutArraySizeMax	INDEX_UINT16
ComCbkrxTOutArraySizeMax	INDEX_UINT16
ComCbkrxAckPtrArraySizeMax	INDEX_UINT16
ComCbktxAckPtrArraySizeMax	INDEX_UINT16
ComCallOutFuncPtrArraySizeMax	INDEX_UINT16
ComTriggerTxCallOutEnable	true
ComRxDataTimeoutAction	RX_DATA_TIMEOUT_ACTION_CONFIG
ComRxTimeoutFactorSize	SIZE_16_BIT
ComRxFirstTimeoutFactorSize	SIZE_16_BIT

Parameters included	
ComTxTimeoutFactorSize	SIZE_16_BIT
ComTxModeRepetitionPeriodFactorS	SIZE_16_BIT
ComTxModeTimeOffsetFactorSize	SIZE_16_BIT
ComTxModeTimePeriodFactorSize	SIZE_16_BIT
ComTxIpduMDTFactorSize	SIZE_16_BIT
ComUpdateBitRxConfig	UPDATE_BIT_INDIVIDUAL
ComUpdateBitTxConfig	UPDATE_BIT_INDIVIDUAL
ComTmsEnable	true
ComFilterReceiverEnable	true
ComTxDynLengthIPduEnable	true
ComRxDynLengthIPduEnable	true
ComFilterOneEveryNPeriodOffSMax	SIZE_16_BIT
ComFilterOneEveryNOccuranceMax	SIZE_16_BIT
ComTxModeDirectEnable	true
ComTxModeNTimesEnable	true
ComTxModePeriodicEnable	true
ComTxModeMixedDirectEnable	true
ComTxModeMixedNTimesEnable	true
ComTxSigConfDeferredEnable	true
ComTxSigConfImmediateEnable	true
ComRxSigConfDeferredEnable	true
ComRxSigConfImmediateEnable	true
ComSignalGwEnable	true
ComCheckValueSizeEnable	true
ComConstCfgAddressEnable	false
ComConstCfgAddress	0
ComRelocatableCfgEnable	true
Com_TxF_MaskNewDiffersMaskOld_En	true

4.2.3. Application programming interface (API)

4.2.3.1. Type definitions

4.2.3.1.1. Com_IpduGroupIdType

Purpose	definition of the Com_IpduGroupIdType
Type	uint16

4.2.3.1.2. Com_IpduGroupVector

Purpose	definition of the Com_IpduGroupVector
Type	uint8[COM_IPDU_GROUP_VECTOR_NUM_BYTES]

4.2.3.1.3. Com_PduGroupIdType

Purpose	definition of the Com_PduGroupIdType
Type	uint8

4.2.3.1.4. Com_RxCalloutType

Purpose	Define Com_RxCalloutType.
Type	boolean() (PduIdType ID, const PduInfoType *PduInfoPtr)

4.2.3.1.5. Com_ServiceIdType

Purpose	definition of the Com_ServiceIdType
Type	uint8

4.2.3.1.6. Com_SignalGroupIdType

Purpose	definition of the Com_SignalGroupIdType
----------------	---

Type	uint16
-------------	--------

4.2.3.1.7. Com_SignalIdType

Purpose	definition of the Com_SignalIdType
Type	uint16

4.2.3.1.8. Com_StatusType

Purpose	definition of the Com_StatusType	
Type	enum	
Constants	COM_UNINIT	
	COM_INIT	

4.2.3.1.9. Com_TxCalloutType

Purpose	Define Com_TxCalloutType.
Type	boolean() (PduIdType ID, PduInfoType *PduInfoPtr)

4.2.3.2. Macro constants

4.2.3.2.1. COMServiceId_ClearIpduGroupVector

Purpose	Definition of constant COMServiceId_ClearIpduGroupVector.
Value	0x1CU
Description	Define COMServiceId_ClearIpduGroupVector

4.2.3.2.2. COMServiceId_CopyRxData

Purpose	Definition of constant COMServiceId_CopyRxData.
----------------	---

Value	0x23
Description	Define COMServiceId_CopyRxData

4.2.3.2.3. COMServiceId_CopyTxData

Purpose	Definition of constant COMServiceId_CopyRxData.
Value	0x24
Description	Define COMServiceId_CopyTxData

4.2.3.2.4. COMServiceId_DeInit

Purpose	Definition of constant COMServiceId_DeInit.
Value	0x02U
Description	Define COMServiceId_DeInit

4.2.3.2.5. COMServiceId_GetConfigurationId

Purpose	Definition of constant COMServiceId_GetConfigurationId.
Value	0x08U
Description	Define COMServiceId_GetConfigurationId

4.2.3.2.6. COMServiceId_GetRxIPduBuffer

Purpose	Definition of constant COMServiceId_GetRxIPduBuffer.
Value	0xFEU
Description	Define COMServiceId_GetRxIPduBuffer

4.2.3.2.7. COMServiceId_GetStatus

Purpose	Definition of constant COMServiceId_GetStatus.
----------------	--

Value	0x07U
Description	Define COMServiceId_GetStatus

4.2.3.2.8. COMServiceId_GetVersionInfo

Purpose	Definition of constant COMServiceId_GetVersionInfo.
Value	0x09U
Description	Define COMServiceId_GetVersionInfo

4.2.3.2.9. COMServiceId_Init

Purpose	Definition of constant COMServiceId_Init.
Value	0x01U
Description	Define COMServiceId_Init

4.2.3.2.10. COMServiceId_InternalAPI

Purpose	Definition of constant COMServiceId_InternalAPI.
Value	0xFFU
Description	Define COMServiceId_InternalAPI

4.2.3.2.11. COMServiceId_InvalidateShadowSignal

Purpose	Definition of constant COMServiceId_InvalidateShadowSignal.
Value	0x16U
Description	Define COMServiceId_InvalidateShadowSignal

4.2.3.2.12. COMServiceId_InvalidateSignal

Purpose	Definition of constant COMServiceId_InvalidateSignal.
----------------	---

Value	0x10U
Description	Define COMServiceId_InvalidSignal

4.2.3.2.13. COMServiceId_InvalidSignalGroup

Purpose	Definition of constant COMServiceId_InvalidSignalGroup.
Value	0x1BU
Description	Define COMServiceId_InvalidSignalGroup

4.2.3.2.14. COMServiceId_IpduGroupControl

Purpose	Definition of constant COMServiceId_IpduGroupControl.
Value	0x03U
Description	Define COMServiceId_IpduGroupControl

4.2.3.2.15. COMServiceId_MainFunctionRouteSignals

Purpose	Definition of constant COMServiceId_MainFunctionRouteSignals.
Value	0x1AU
Description	Define COMServiceId_MainFunctionRouteSignals

4.2.3.2.16. COMServiceId_MainFunctionRx

Purpose	Definition of constant COMServiceId_MainFunctionRx.
Value	0x18U
Description	Define COMServiceId_MainFunctionRx

4.2.3.2.17. COMServiceId_MainFunctionTx

Purpose	Definition of constant COMServiceId_MainFunctionTx.
----------------	---

Value	0x19U
Description	Define COMServiceId_MainFunctionTx

4.2.3.2.18. COMServiceId_ReceiveDynSignal

Purpose	Definition of constant COMServiceId_ReceiveDynSignal.
Value	0x22
Description	Define COMServiceId_ReceiveDynSignal

4.2.3.2.19. COMServiceId_ReceiveShadowSignal

Purpose	Definition of constantDefinition of constant COMServiceId_ReceiveShadowSignal.
Value	0x0FU
Description	Define COMServiceId_ReceiveShadowSignal

4.2.3.2.20. COMServiceId_ReceiveSignal

Purpose	Definition of constant COMServiceId_ReceiveSignal.
Value	0x0BU
Description	Define COMServiceId_ReceiveSignal

4.2.3.2.21. COMServiceId_ReceiveSignalGroup

Purpose	Definition of constant COMServiceId_ReceiveSignalGroup.
Value	0x0EU
Description	Define COMServiceId_ReceiveSignalGroup

4.2.3.2.22. COMServiceId_ReceiveSignalGroupArray

Purpose	Definition of constant COMServiceId_ReceiveSignalGroupArray.
----------------	--

Value	0x44U
Description	Define COMServiceId_ReceiveSignalGroupArray

4.2.3.2.23. COMServiceId_ReceptionDMControl

Purpose	Definition of constant COMServiceId_ReceptionDMControl.
Value	0x06U
Description	Define COMServiceId_EnableReceptionDM

4.2.3.2.24. COMServiceId_RxIndication

Purpose	Definition of constant COMServiceId_RxIndication.
Value	0x42U
Description	Define COMServiceId_RxIndication

4.2.3.2.25. COMServiceId_SendDynSignal

Purpose	Definition of constant COMServiceId_SendDynSignal.
Value	0x21
Description	Define COMServiceId_SendDynSignal

4.2.3.2.26. COMServiceId_SendSignal

Purpose	Definition of constant COMServiceId_SendSignal.
Value	0x0AU
Description	Define COMServiceId_SendSignal

4.2.3.2.27. COMServiceId_SendSignalGroup

Purpose	Definition of constant COMServiceId_SendSignalGroup.
----------------	--

Value	0x0DU
Description	Define COMServiceId_SendSignalGroup

4.2.3.2.28. COMServiceId_SendSignalGroupArray

Purpose	Definition of constant COMServiceId_SendSignalGroupArray.
Value	0x43U
Description	Define COMServiceId_SendSignalGroupArray

4.2.3.2.29. COMServiceId_SetIpduGroup

Purpose	Definition of constant COMServiceId_SetIpduGroup.
Value	0x1DU
Description	Define COMServiceId_SetIpduGroup

4.2.3.2.30. COMServiceId_StartOfReception

Purpose	Definition of constant COMServiceId_StartOfReception.
Value	0x25
Description	Define COMServiceId_StartOfReception

4.2.3.2.31. COMServiceId_SwitchIpduTxMode

Purpose	Definition of constant COMServiceId_SwitchIpduTxMode.
Value	0x27U
Description	Define COMServiceId_SwitchIpduTxMode

4.2.3.2.32. COMServiceId_TpRxIndication

Purpose	Definition of constant COMServiceId_TpRxIndication.
----------------	---

Value	0x1EU
Description	Define COMServiceId_TpRxIndication

4.2.3.2.33. COMServiceId_TpTxConfirmation

Purpose	Definition of constant COMServiceId_TpTxConfirmation.
Value	0x26
Description	Define COMServiceId_TpTxConfirmation

4.2.3.2.34. COMServiceId_TriggerIPDUSend

Purpose	Definition of constant COMServiceId_TriggerIPDUSend.
Value	0x17U
Description	Define COMServiceId_TriggerIPDUSend

4.2.3.2.35. COMServiceId_TriggerTransmit

Purpose	Definition of constant COMServiceId_TriggerTransmit.
Value	0x41U
Description	Define COMServiceId_TriggerTransmit

4.2.3.2.36. COMServiceId_TxConfirmation

Purpose	Definition of constant COMServiceId_TxConfirmation.
Value	0x40U
Description	Define COMServiceId_TxConfirmation

4.2.3.2.37. COMServiceId_UpdateShadowSignal

Purpose	Definition of constant COMServiceId_UpdateShadowSignal.
----------------	---

Value	0x0CU
Description	Define COMServiceId_UpdateShadowSignal

4.2.3.2.38. COM_E_PARAM

Purpose	Definition of error code COM_E_PARAM.
Value	1U
Description	Define COM_E_PARAM

4.2.3.2.39. COM_E_PARAM_POINTER

Purpose	Definition of error code COM_E_PARAM_POINTER.
Value	3U
Description	Define COM_E_PARAM_POINTER

4.2.3.2.40. COM_E_SIGNAL_TOO_WIDE

Purpose	Define COM_E_SIGNAL_TOO_WIDE.
Value	0x21U

4.2.3.2.41. COM_E_UNINIT

Purpose	Definition of error code COM_E_UNINIT.
Value	2U
Description	Define COM_E_UNINIT

4.2.3.2.42. COM_INSTANCE_ID

Purpose	Com instance ID.
----------------	------------------

Value	0U
--------------	----

4.2.3.3. Functions

4.2.3.3.1. Com_ClearIpduGroupVector

Purpose	Com_ClearIpduGroupVector - sets all bits of the given Com_IpduGroupVector to 0.	
Synopsis	<pre>void Com_ClearIpduGroupVector (Com_- IpduGroupVector ipduGroupVector);</pre>	
Service ID	0x1c	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	ipduGroupVector	I-PDU group vector to be cleared

4.2.3.3.2. Com_CopyRxData

Purpose	<p>Com_CopyRxData Called once upon reception of each segment. Within this call, the received data is copied to the receive TP buffer. The function must only be called if the connection has been accepted by an initial call to Com_StartOfReception. Preconditions:</p> <ul style="list-style-type: none"> ▶ prior to this call, the COM module shall be initialized ▶ the Pdu must be started before. 	
Synopsis	<pre>BufReq_ReturnType Com_CopyRxData (PduId- Type PduId , const PduInfoType * PduInfo- Pointer , PduLengthType * RxBufferSizePtr);</pre>	
Parameters (in)	PduId	- ID of Tp I-PDU to be transmitted
	PduInfoPointer	- Pointer to a PduInfoType, which indicates the number of bytes to be copied (SduLength) and the location of the source data (SduDataPtr). An SduLength of 0 is possible in order to poll the available receive buffer size. In this case no data are to be copied and PduInfoPtr might be invalid.
Parameters (out)	RxBufferSizePtr	- Remaining receive buffer after successful completion of this call (Com_CopyRx-

		Data returns BUFREQ_OK otherwise output parameter RxBufferSizePtr does not change).
Return Value	BUFREQ_OK	- Data has been copied to the receive buffer completely as requested.
	BUFREQ_E_BUSY	- The receive buffer is actually not available (implementation specific).
BUFREQ_E_NOT_OK	- Data has not been copied. Request failed.	

4.2.3.3.3. Com_CopyTxData

Purpose	Com_CopyTxData function which copy the requested transmit data of the large IPDU Preconditions: <ul style="list-style-type: none"> ▶ prior to this call, the COM module shall be initialized ▶ the Pdu must be started before. 	
Synopsis	<pre>BufReq_ReturnType Com_CopyTxData (PduIdType PduId , PduInfoType * PduInfoPtr , RetryInfoType * RetryInfoPtr , PduLengthType * TxDataCntPtr);</pre>	
Parameters (in)	PduId	- ID of Tp I-PDU to be transmitted
	PduInfoPtr	- Pointer to a PduInfoType, which indicates the number of bytes to be copied (SduLength) and the location where the data have to be copied to (SduDataPtr). An SduLength of 0 is possible in order to poll the available transmit data count. In this case no data are to be copied and SduDataPtr might be invalid.
	RetryInfoPtr	- The COM module ignores the value of this pointer, since it always keeps the complete buffer until the transmission of a large I-PDU is either confirmed or aborted.
Parameters (out)	TxDataCntPtr	- Remaining Tx data after successful completion of this call (Com_CopyTxData returns BUFREQ_OK otherwise output parameter TxDataCntPtr does not change).
Return Value		

	BUFREQ_OK	- Data has been copied to the transmit buffer completely as requested.
	BUFREQ_E_BUSY	- The transmission buffer is actually not available (implementation specific).
BUFREQ_E_NOT_OK	- Data has not been copied. Request failed, in case the corresponding I-PDU was stopped.	

4.2.3.3.4. Com_Delnit

Purpose	Com_Delnit - sets COM to de-initialized state.
Synopsis	<pre>void Com_DeInit (void);</pre>
Service ID	0x02
Sync/Async	Synchronous
Reentrancy	Non re-entrant
Description	Simply sets the global variable Com_Status to COM_UNINIT.

4.2.3.3.5. Com_GetConfigurationId

Purpose	Com_GetConfiguratinold - get post-build-time configuration ID.
Synopsis	<pre>uint32 Com_GetConfigurationId (void);</pre>
Service ID	0x08
Sync/Async	Synchronous
Reentrancy	Re-entrant
Return Value	post-build-time configuration ID

4.2.3.3.6. Com_GetRxIPduBuffer

Purpose	Com_GetRxIPduBuffer - returns Rx IPdu buffer reference The service Com_GetRxIPduBuffer returns the buffer object identified by ComRxPduld with the buffer referenced by the PdulInfoPtr parameter. Preconditions: COM must be initialized.
----------------	--

Synopsis	<pre>uint8 Com_GetRxIPduBuffer (PduIdType Com- RxPduId , PduInfoType * PduInfoPtr);</pre>	
Parameters (in)	ComRxPduId	- ID of the Rx ComIPdu
Parameters (out)	PduInfoPtr	- Rx IPdu buffer reference
Return Value	Function execution success status	
	E_OK	- success
	E_NOT_OK	- failure (Com not initialized or service failed due to development error)

4.2.3.3.7. Com_GetStatus

Purpose	returns status of Com	
Synopsis	<pre>Com_StatusType Com_GetStatus (void);</pre>	
Service ID	0x07	
Sync/Async	Synchronous	
Reentrancy	Re-entrant	
Return Value	Result of init status	
	COM_INIT	the module is initialized
	COM_UNINIT	the module is not initialized
Description	This function returns whether the module is initialized	

4.2.3.3.8. Com_GetVersionInfo

Purpose	Returns the module version information.	
Synopsis	<pre>void Com_GetVersionInfo (Std_Ver- sionInfoType * versionInfoPtr);</pre>	
Service ID	0x09	
Sync/Async	Synchronous	
Reentrancy	Re-entrant	
Parameters (out)	versionInfoPtr	Address the version information should be written to.
Description	This service returns the version information of this module.	

4.2.3.3.9. Com_Init

Purpose	Com_Init - Initializes the Com module.	
Synopsis	<pre>void Com_Init (const Com_ConfigType * Com_ConfigPtr);</pre>	
Service ID	0x01	
Sync/Async	Synchronous	
Reentrancy	Non re-entrant	
Parameters (in)	Com_ConfigPtr	Pointer to configuration structure that holds the Com module post-build-time configuration data.
Description	Function to initialize the Com module. First function to be called of Com. The module calling the function Com_Init has to include Com_PBcfg.h. The invocation of the Com_Init function is without usage of PbcfgM Com_Init(&<short name of Com configuration>); or Com_Init(&Com_ConfigLayout.Com_RootConfig); The invocation of the Com_Init function for usage of PbcfgM Com_Init(NULL_PTR);	

4.2.3.3.10. Com_IpduGroupControl

Purpose	Com_IpduGroupControl - starts/stops I-PDU.	
Synopsis	<pre>void Com_IpduGroupControl (Com_IpduGroupVector ipduGroupVector , boolean Initialize);</pre>	
Service ID	0x03	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	ipduGroupVector	I-PDU group vector containing the activation state (stopped = 0/ started = 1) for all IPdus.
	Initialize	flag to request initialization of the I-PDUs which are newly started
Description	Function to start/stop every Rx-Ipdu and Tx-Ipdu according to the passed states of the ComIpduGroups in the parameter ipduGroupVector. Preconditions: ► COM must be initialized	

4.2.3.3.11. Com_IsValidConfig

Purpose	Com_IsValidConfig - Checks validity of the post-build configuration.	
Synopsis	<code>Std_ReturnType Com_IsValidConfig (const void * ConfigPtr);</code>	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	ConfigPtr	Pointer to configuration structure that holds the Com module post-build-time configuration data.
Return Value	Function execution success status	
	E_OK	the provided module configuration is valid
	E_NOT_OK	the provided module configuration is invalid
Description	<p>Checks if the post build configuration is valid. A configuration is invalid if</p> <ul style="list-style-type: none"> ▶ the platform signature does not match. ▶ the published information signature does not match. ▶ the link time signature does not match. ▶ the compile time signature does not match. ▶ the function is called with a null pointer. 	

4.2.3.3.12. Com_MainFunctionRouteSignals

Purpose	Com_MainFunctionRouteSignals - handle cyclic Signal Gateway tasks.	
Synopsis	<code>void Com_MainFunctionRouteSignals (void);</code>	
Service ID	0x1A	
Sync/Async	Synchronous	
Reentrancy	Non re-entrant	
Description	<p>This function handles cyclic receiving and sending (group)signals(group) for the Signal Gateway functionality (SigGW). Preconditions:</p> <ul style="list-style-type: none"> ▶ COM must be initialized 	

4.2.3.3.13. Com_MainFunctionRx

Purpose	Com_MainFunctionRx - handle cyclic receiving-related tasks.
Synopsis	<pre>void Com_MainFunctionRx (void);</pre>
Service ID	0x18
Sync/Async	Synchronous
Reentrancy	Non re-entrant
Description	<p>This function handles cyclic receiving-related tasks like reception deadline monitoring. Preconditions:</p> <ul style="list-style-type: none">▶ COM must be initialized

4.2.3.3.14. Com_MainFunctionTx

Purpose	Com_MainFunctionTx - handle cyclic sending-related tasks.
Synopsis	<pre>void Com_MainFunctionTx (void);</pre>
Service ID	0x19
Sync/Async	Synchronous
Reentrancy	Non re-entrant
Description	<p>This function handles cyclic sending-related tasks such as minimum delay time and cyclic sending. Preconditions:</p> <ul style="list-style-type: none">▶ COM must be initialized

4.2.3.3.15. Com_ReceiveDynSignal

Purpose	Com_ReceiveDynSignal - get a dynamic length signal's actual value from COM.
Synopsis	<pre>uint8 Com_ReceiveDynSignal (Com_SignalIdType SignalId , void * SignalDataPtr , uint16 * LengthPtr);</pre>
Service ID	0x22

Sync/Async	Synchronous	
Reentrancy	Re-entrant for different values of SignalId only.	
Parameters (in)	SignalId	ID of signal to receive
	SignalDataPtr	piece of memory to copy data to
Parameters (in,out)	LengthPtr	in: maximum length that could be received out: length of the dynamic length signal
Return Value	Result of operation	
	E_OK	success
	E_NOT_OK	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)
Description	<p>Com_ReceiveDynSignal copies the data of the signal identified by SignalId to the location specified by SignalDataPtr and stores the length of the dynamic length signal at the position given by the Length parameter.</p> <p>Preconditions:</p> <ul style="list-style-type: none"> ► COM must be initialized 	

4.2.3.3.16. Com_ReceiveShadowSignal

Purpose	Com_ReceiveShadowSignal - get a group signal's value from shadow buffer.	
Synopsis	<pre>void Com_ReceiveShadowSignal (Com_SignalId- Type SignalId , void * SignalDataPtr);</pre>	
Service ID	0x0F	
Sync/Async	Synchronous	
Reentrancy	Re-entrant for different values of SignalId only.	
Parameters (in)	SignalId	ID of group signal to receive
	SignalDataPtr	piece of memory to copy data to
Description	<p>This function returns the value of a group signal from its shadow buffer of the signal group</p> <p>Preconditions:</p> <ul style="list-style-type: none"> ► COM must be initialized 	

4.2.3.3.17. Com_ReceiveSignal

Purpose	Com_ReceiveSignal - get a signal's actual value from COM.	
Synopsis	<pre>uint8 Com_ReceiveSignal (Com_SignalId- Type SignalId , void * SignalDataPtr);</pre>	
Service ID	0x0B	
Sync/Async	Synchronous	
Reentrancy	Re-entrant for different values of SignalId only.	
Parameters (in)	SignalId	ID of signal to receive
Parameters (out)	SignalDataPtr	piece of memory to copy data to
Return Value	Result of operation	
	E_OK	success
	COM_SERVICE_NOT_AVAILABLE	Com is disabled, no valid value passed over
Description	<p>This function returns the actual value of a signal. This function is always available. If the configuration parameter ComGeneratedRcvSigEnable is disabled, this function maps to the generic implementation of the function. If the configuration parameter ComGeneratedRcvSigEnable is enabled and ComMapReceiveSignal == Com_ReceiveSignalGeneric, this function maps to the generic implementation of the function. If the configuration parameter ComGeneratedRcvSigEnable is enabled and ComMapReceiveSignal == Com_ReceiveSignalGenerated, this function maps to the generated implementation of the function. Preconditions:</p> <ul style="list-style-type: none"> ► COM must be initialized 	

4.2.3.3.18. Com_ReceiveSignalGeneric

Purpose	<p>Com_ReceiveSignalGeneric - get a signal's actual value from COM This function returns the actual value of a signal. This function is only available if the configuration parameter ComGeneratedRcvSigEnable is enabled and ComMapReceiveSignal == Com_ReceiveSignalGenerated. If available, this function maps to the generic implementation of the function. Preconditions:</p> <ul style="list-style-type: none"> ► COM must be initialized.
Synopsis	<pre>uint8 Com_ReceiveSignalGeneric (Com_Sig- nalIdType SignalId , void * SignalDataPtr);</pre>

Parameters (in)	SignalId	ID of signal to receive
Parameters (out)	SignalDataPtr	piece of memory to copy data to
Return Value	Result of operation	
	E_OK	success
	COM_SERVICE_NOT_AVAILABLE	Com is disabled, no valid value passed over

4.2.3.3.19. Com_ReceiveSignalGroup

Purpose	Com_ReceiveSignalGroup - copies the actual value of a signal group into the shadow buffer.	
Synopsis	<pre>uint8 Com_ReceiveSignalGroup (Com_ SignalGroupIdType SignalGroupId);</pre>	
Service ID	0x0E	
Sync/Async	Synchronous	
Reentrancy	Re-entrant for different values of SignalGroupId only.	
Parameters (in)	SignalGroupId	ID of signal group
Return Value	Result of operation	
	E_OK	success
	COM_SERVICE_NOT_AVAILABLE	Com is disabled, no valid value passed over
Description	<p>This function copies the value of a Rx signal group from its Rx-IPdu into the its shadow buffer Preconditions:</p> <ul style="list-style-type: none"> ► COM must be initialized 	

4.2.3.3.20. Com_ReceiveSignalGroupArray

Purpose	Com_ReceiveSignalGroupArray - access signal group array.	
Synopsis	<pre>uint8 Com_ReceiveSignalGroupArray (Com_Signal- GroupIdType SignalGroupId , uint8 * SignalGroupAr- rayPtr , uint16 * SignalGroupArrayLengthPtr);</pre>	
Service ID	0x44	
Sync/Async	Synchronous	

Reentrancy	Re-entrant for different values of SignalGroupId only.	
Parameters (in)	SignalGroupId	ID of signal group to be received.
	SignalGroupArrayLength	reference to length of the signal group array
Parameters (out)	SignalGroupArrayPtr	reference to the location where the received signal group array shall be stored
Return Value	Result of operation	
	E_OK	service has been accepted
	COM_SERVICE_NOT_AVAILABLE	corresponding I-PDU group was stopped (or service failed due to development error)
Description	<p>The service Com_ReceiveSignalGroupArray copies the received signal group array representation from the I-PDU to the SignalGroupArrayPtr. Preconditions:</p> <ul style="list-style-type: none"> ► COM must be initialized 	

4.2.3.3.21. Com_ReceptionDMControl

Purpose	Com_ReceptionDMControl - enables or disables Rx I-PDU Deadline Monitoring.	
Synopsis	<pre>void Com_ReceptionDMControl (Com_ IpduGroupVector ipduGroupVector);</pre>	
Service ID	0x06	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	ipduGroupVector	I-PDU group vector containing the activation of RxDM (disable = 0/ enable = 1) for all Rx-Pdus.
Description	<p>Function to enables/disables every RxDM of every Rx-IPdu according to the passed states of the ComIpduGroups in the parameter ipduGroupVector. Preconditions:</p> <ul style="list-style-type: none"> ► COM must be initialized 	

4.2.3.3.22. Com_RxIndication

Purpose	Com_RxIndication - Signal the COM a PDU has arrived.
----------------	--

Synopsis	<pre>void Com_RxIndication (PduIdType Com- RxPduId , PduInfoType * PduInfoPtr);</pre>	
Service ID	0x42	
Sync/Async	Synchronous	
Reentrancy	Re-entrant for different values of ComRxPduId only.	
Parameters (in)	ComRxPduId	ID of the received I-PDU.
	PduInfoPtr	Contains the length (SduLength) of the received I-PDU and a pointer to a buffer (SduDataPtr) containing the I-PDU.
Description	<p>This functions signals the COM that a PDU has arrived Preconditions:</p> <ul style="list-style-type: none"> ► COM should be initialized 	

4.2.3.3.23. Com_SendDynSignal

Purpose	Com_SendDynSignal - send a dynamic length signal.	
Synopsis	<pre>uint8 Com_SendDynSignal (Com_SignalIdType Sig- nalId , const void * SignalDataPtr , uint16 Length);</pre>	
Service ID	0x21	
Sync/Async	Asynchronous	
Reentrancy	Re-entrant for different values of SignalId only.	
Parameters (in)	SignalId	ID of the signal to be sent
	SignalDataPtr	place in memory to copy the data from
	Length	Length of the dynamic length signal to be send
Return Value	Result of operation	
	E_OK	success
	E_NOT_OK	in case the Length is greater than the configured ComSignalLength of this sent signal
	COM_SERVICE_NOT_AVAILABLE	failure (Com not initialized, corresponding I-PDU group was stopped or service failed due to development error)
	BUFREQ_E_BUSY	- The receive Tp buffer is actually not available

Description	The service Com_SendSignal updates the signal object identified by SignalId and from signal type UINT8_DYN with the value referenced by the SignalDataPtr parameter. Preconditions: COM must be initialized.
--------------------	--

4.2.3.3.24. Com_SendSignal

Purpose	Com_SendSignal - send a signal see COM197.	
Synopsis	<pre>uint8 Com_SendSignal (Com_SignalIdType SignalId , const void * SignalDataPtr);</pre>	
Service ID	0x0A	
Sync/Async	Asynchronous	
Reentrancy	Re-entrant for different values of SignalId only.	
Parameters (in)	SignalId	ID of the signal to be sent
	SignalDataPtr	place in memory to copy the data from
Return Value	Result of operation	
	E_OK	success
	COM_SERVICE_NOT_AVAILABLE	failure (Com not initialized, corresponding I-PDU group was stopped or service failed due to development error)
	BUFREQ_E_BUSY	- The receive Tp buffer is actually not available
Description	The service Com_SendSignal updates the signal object identified by SignalId with the value referenced by the SignalDataPtr parameter. Preconditions: COM must be initialized.	

4.2.3.3.25. Com_SendSignalGroup

Purpose	Com_SendSignalGroup - send a signal group.	
Synopsis	<pre>uint8 Com_SendSignalGroup (Com_ SignalGroupIdType SignalGroupId);</pre>	
Service ID	0x0D	
Sync/Async	Asynchronous	
Reentrancy	Re-entrant for different values of SignalGroupId only.	

Parameters (in)	SignalGroupId	ID of the signal group to be sent
Return Value	Result of operation	
	E_OK	success
	COM_SERVICE_NOT_AVAILABLE	failure (Com not initialized, corresponding I-PDU group was stopped or service failed due to development error)
	BUFREQ_E_BUSY	- The receive Tp buffer is actually not available
Description	The service Com_SendSignalGroup updates the signal group object identified by SignalGroupId Preconditions: COM must be initialized.	

4.2.3.3.26. Com_SendSignalGroupArray

Purpose	Com_SendSignalGroupArray - update and send a signal group.	
Synopsis	<pre>uint8 Com_SendSignalGroupArray (Com_Signal- GroupIdType SignalGroupId , const uint8 * Signal- GroupArrayPtr , uint16 SignalGroupArrayLength);</pre>	
Service ID	0x43	
Sync/Async	Asynchronous	
Reentrancy	Re-entrant for different values of SignalGroupId only.	
Parameters (in)	SignalGroupId	ID of the signal group to be sent
	SignalGroupArrayPtr	Reference to the signal group array to be transmitted
	SignalGroupArrayLength	Length of the signal group array
Return Value	Result of operation	
	E_OK	success
	COM_SERVICE_NOT_AVAILABLE	failure (Com not initialized, corresponding I-PDU group was stopped or service failed due to development error)
	BUFREQ_E_BUSY	- The receive Tp buffer is actually not available
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. Preconditions: COM must be initialized.	

4.2.3.3.27. Com_SetIpduGroup

Purpose	Com_SetIpduGroup - sets the value of a bit in an I-PDU group vector.	
Synopsis	<pre>void Com_SetIpduGroup (Com_IpduGroupVector ipduGroupVec- tor , Com_IpduGroupIdType ipduGroupId , boolean bitval);</pre>	
Service ID	0x1d	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	ipduGroupVector	I-PDU group vector to be modified
	ipduGroupId	identifies the corresponding bit in the I-PDU group vector
	bitval	new value of the corresponding bit
Description	<p>Preconditions:</p> <ul style="list-style-type: none"> ► COM must be initialized 	

4.2.3.3.28. Com_StartOfReception

Purpose	<p>Com_StartOfReception returns the pointer to the size of the AUTOSAR COM module's internal receive buffer for the I-PDU with ID ComRxPduId. Preconditions:</p> <ul style="list-style-type: none"> ► prior to this call, the COM module shall be initialized ► the Pdu must be started before. 	
Synopsis	<pre>BufReq_ReturnType Com_StartOfReception (PduIdType ComRxPduId , PduLengthType TpSduLength , PduLengthType * RxBufferSizePtr);</pre>	
Parameters (in)	ComRxPduId	- ID of Tp I-PDU to be received.
	TpSduLength	- complete length of the TP I-PDU to be received.
Parameters (out)	RxBufferSizePtr	- Pointer to the size of internal TP-receive buffer
Return Value	BUFREQ_OK	- Connection has been accepted. RxBufferSizePtr indicates the available receive buffer.
	BUFREQ_E_NOT_OK	- Connection has been rejected. RxBufferSizePtr remains unchanged.

	BUFREQ_E_OVFL	- In case the configured buffer size as specified via ComPduldRef.PduLength is smaller than TpSduLength.
BUFREQ_E_BUSY	- In case the reception buffer is actually not available for a new reception (implementation specific).	

4.2.3.3.29. Com_SwitchIpduTxMode

Purpose	Switch to a Transmission Mode.	
Synopsis	<pre>void Com_SwitchIpduTxMode (PduIdType PduId , boolean Mode);</pre>	
Service ID	0x27	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different Pdulds. Non reentrant for the same PduId	
Parameters (in)	PduId	ID of the PDU to be sent
	Mode	the transmission mode that shall be set
Description	<p>The function sets the transmission mode of the I-PDU referenced by PduId to Mode</p> <p>Preconditions:</p> <ul style="list-style-type: none"> ▶ The COM must be initialized ▶ Should not be mixed with signal based TMS 	

4.2.3.3.30. Com_TpRxIndication

Purpose	Com_TpRxIndication - indicating the correct, or incorrect, end of the reception process.	
Synopsis	<pre>void Com_TpRxIndication (PduId- Type PduId , NotifResultType Result);</pre>	
Service ID	0x1E	
Sync/Async	Synchronous	
Reentrancy	Re-entrant for different values of ComRxPduld only.	
Parameters (in)	PduId	- ID of the AUTOSAR COM module's I-PDU that has been received. Identifies the data that has been received.

	Result	- NTFRSLT_OK: the complete I-PDU has been received and is stored in the receive buffer. <ANY other="" value>="">: the I-PDU has not been received; the receive buffer can be unlocked by the AUTOSAR COM
Description	<p>This functions signals the COM the correct, or incorrect, end of the reception process. Preconditions:</p> <ul style="list-style-type: none"> ► COM should be initialized 	

4.2.3.3.31. Com_TpTxConfirmation

Purpose	<p>TpTxConfirmation Function to signal the COM that an large IPDU has been transmitted Preconditions:</p> <ul style="list-style-type: none"> ► COM should be initialized. 	
Synopsis	<pre>void Com_TpTxConfirmation (PduId- Type PduId , NotifResultType Result);</pre>	
Parameters (in)	ComTxPduId	- ID of the large PDU which was transmitted successfully
	Result	- Result of the transmission of the I-PDU

4.2.3.3.32. Com_TriggerIPDUSend

Purpose	Send an IPDU.	
Synopsis	<pre>void Com_TriggerIPDUSend (PduIdType ComTxPduId);</pre>	
Service ID	0x17	
Sync/Async	Synchronous	
Reentrancy	Non re-entrant	
Parameters (in)	ComTxPduId	ID of the PDU to be sent
Description	<p>The function triggers sending of an IPDU Preconditions:</p> <ul style="list-style-type: none"> ► The COM must be initialized ► The function must only be called from callouts 	

4.2.3.3.33. Com_TriggerTransmit

Purpose	Com_TriggerTransmit - copy data to PDU-router memory.	
Synopsis	<pre>Std_ReturnType Com_TriggerTransmit (PduId- Type ComTxPduId , PduInfoType * PduInfoPtr);</pre>	
Service ID	0x41	
Sync/Async	Synchronous	
Reentrancy	Re-entrant for different values of ComTxPduId only.	
Parameters (in)	ComTxPduId	ID of the PDU which's data shall be copied
	PduInfoPtr	Contains a pointer to a buffer (SduDataPtr) to where the SDU shall be copied to. On return, the service will indicate the length of the copied SDU data in SduLength.
Return Value	E_OK:	SDU has been copied and SduLength indicates the number of copied bytes.
	E_NOT_OK:	No SDU has been copied. PduInfoPtr must not be used since it may contain a NULL pointer or point to invalid data.
Description	a function to be used to fetch data from the COM immediately Preconditions: <ul style="list-style-type: none"> ► COM should be initialized 	

4.2.3.3.34. Com_TxConfirmation

Purpose	Com_TxConfirmation.	
Synopsis	<pre>void Com_TxConfirmation (PduIdType ComTxPduId);</pre>	
Service ID	0x40	
Sync/Async	Synchronous	
Reentrancy	Re-entrant for different values of ComTxPduId only.	
Parameters (in)	ComTxPduId	ID of the PDU which was transmitted successfully
Description	Function to signal the COM that an IPDU has been transmitted Preconditions: <ul style="list-style-type: none"> ► COM should be initialized 	

4.2.3.3.35. Com_UpdateShadowSignal

Purpose	Com_UpdateShadowSignal - updates the data in the signal group.	
Synopsis	<pre>void Com_UpdateShadowSignal (Com_SignalId- Type SignalId , const void * SignalDataPtr);</pre>	
Service ID	0x0C	
Sync/Async	Synchronous	
Reentrancy	Re-entrant for different values of SignalId only.	
Parameters (in)	SignalId	ID of the group signal to be updated
	SignalDataPtr	place in memory to copy the data from
Description	The service Com_UpdateShadowSignal updates the group signal object identified by SignalId with the value referenced by the SignalDataPtr parameter. Preconditions: COM must be initialized.	

4.2.4. Integration notes

4.2.4.1. Exclusive areas

This section describes the exclusive areas used by the Com module.

4.2.4.1.1. COM_EXCLUSIVE_AREA_0

Protected data structures	All shared data that shall be protected from mutual access.
Recommended locking mechanism	<p>The locking mechanism for this exclusive area can be disabled if:</p> <ul style="list-style-type: none">▶ all Tx-related functions do not interrupt each other, and▶ all Rx-related functions do not interrupt each other, and▶ if signal gateway is used: Tx and Rx related functions do not interrupt each other <p>If the conditions listed above do not apply, the exclusive area shall be protected by a locking mechanism. The options for</p>



	locking are described in the EB tresos AutoCore Generic documentation. Refer to the section Mapping exclusive areas in the basic software modules in the Integration notes section for details.
--	---

4.2.4.1.2. COM_EXCLUSIVE_AREA_1

Protected data structures	All shared data that shall be protected from mutual access when they are accessed via Com_RxIndication() or Com_Tx-Confirmation().
Recommended locking mechanism	<p>The locking mechanism for this exclusive area can be disabled if:</p> <ul style="list-style-type: none">▶ no Com module related function can interrupt Com_-RxIndicaiton, and▶ no Com module related function can interrupt Com_Tx-Confirmation <p>If the conditions listed above do not apply, the exclusive area shall be protected by a locking mechanism. The options for locking are described in the EB tresos AutoCore Generic documentation. Refer to the section Mapping exclusive areas in the basic software modules in the Integration notes section for details.</p>

4.2.4.2. Production errors

Production errors are not reported by the Com module.

4.2.4.3. Memory mapping

General information about memory mapping is provided in the EB tresos AutoCore Generic documentation. Refer to the section Memory mapping and compiler abstraction in the Integration notes section for details.

The following table provides the list of sections that may be mapped for this module:

Memory section

CODE
APPL_CODE
CONST_32
CONST_UNSPECIFIED
VAR_FAST_NO_INIT_UNSPECIFIED
VAR_FAST_INIT_16BIT
VAR_FAST_NO_INIT_16BIT
VAR_FAST_INIT_32BIT
VAR_FAST_NO_INIT_32BIT
VAR_FAST_INIT_8BIT
VAR_FAST_NO_INIT_8BIT
VAR_NOINIT_UNSPECIFIED
CONFIG_DATA_UNSPECIFIED
CODE_CC_BLOCK

4.2.4.4. Integration requirements

WARNING



Integration requirements list is not exhaustive

The following list of integration requirements helps you to integrate your product. However, this list is not exhaustive. You also require information from the user's guide, release notes, and EB tresos AutoCore known issues to successfully integrate your product.

4.2.4.4.1. Com.EB.IntReq.Preemption01

Description	The Com_IpduGroupControl() or Com_Init() API shall not be interrupted or interrupt any function of the Com Module which accesses the RAM.
Rationale	Race conditions when Com_IpduGroupControl() or Com_Init() is preempted or pre-empts other Com services. Com_IpduGroupControl() with Initialize = TRUE or Com_Init() must not be interrupted or interrupt any function of the Com Module which accesses the RAM. Since these functions do not use protected memory access, the result might be undefined behavior of the Com module. For example, a mess-up of the number of transmission in case of n-time transmission. In order to prevent this race condition, all interrupts have to be disabled during the call to Com_Init() and Com_IpduGroupControl() with Initialize = TRUE.

4.2.4.4.2. Com.EB.IntReq.Preemption02

Description	<p>Restrictions to prevent race conditions in Com's Tx-path. The Com module exhibits several race conditions in its transmission path that can cause inconsistent and/or mutilated data to be transmitted. The transmission of an I-PDU can be triggered by a Tx-signal API if the I-PDU has a direct part (transmission mode is DIRECT or MIXED). The Tx-signal APIs are Com_SendSignal(), Com_SendDynSignal(), Com_SendSignalGroup(), and Com_SendSignalGroupArray(). The Tx-signal APIs have write access to the Com-internal I-PDU buffer. Note that (the internal implementations of) these APIs are also used in context of Com_MainFunctionRouteSignals(). Additionally the transmission of an I-PDU can be triggered in context of Com_MainFunctionTx(), Com_TriggerIPDUSend(), or Com_IpduGroupControl(), or Com_SwitchIpduTxMode. Triggering of a transmission in general requires the read access to the Com-internal I-PDU buffer by the Com lower layers. Depending on the implementation of a Tx-callout (ComIPduCallout and ComIPduTriggerTransmitCallout), it requires read and/or write access to the Com-internal I-PDU buffer. The callouts are invoked when a transmission is triggered. Depending on the underlying bus system, the API Com_TriggerTransmit() is invoked, which requires read access to the Com-internal I-PDU buffer. A race occurs when an ongoing transmission (access to the Com-internal I-PDU buffer by Com lower layer and Com callout) is interrupted by an invocation of a Tx-signal API. A race occurs when an ongoing transmission is interrupted by an API which triggers another transmission for the same I-PDU and a configured Com callout changes data. This behavior leads to the following cases:</p> <ul style="list-style-type: none"> - An I-PDU has a direct part. It also has a call to a Tx-signal API to a signal/signal group, in which one of the following transfer properties is interrupted by another Tx-signal API call of a signal of the very same I-PDU: TRIGGERED, TRIGGERED_ON_CHANGE, TRIGGERED_ON_CHANGE_WITHOUT_REPETITION, or TRIGGERED_WITHOUT_REPETITION. - A call to a Tx-signal API for a signal/signal group that belongs to the I-PDU interrupts a call to one of the following APIs of the very same I-PDU: Com_TriggerIPDUSend(), Com_IpduGroupControl(), Com_SwitchIpduTxMode(), or Com_TriggerTransmit(). - A call to a Tx-signal API interrupts a call to Com_MainFunctionTx(). - A callout uses the data of the I-PDU for a calculation (e.g. to calculate a CRC) and a call to Tx-signal API interrupts the sending of the I-PDU.
--------------------	---

	<p>With a call to Com_SendDynSignal() not only the content of an I-PDU may change, but also the length of the I-PDU. Work-around To prevent inconsistencies in the I-PDU, ensure the following:</p> <ul style="list-style-type: none"> - A call to a Tx-signal API that triggers a transmission does not interrupt a call to a Tx-signal API for signals which belong to the same I-PDU. - A call to a Tx-signal API does not interrupt one of the following APIs: Com_TriggerIPDUSend(), Com_SwitchIpduTxMode(), or Com_TriggerTransmit(). - A call to a Tx-signal API does not interrupt Com_MainFunctionTx(). - Additionally, if a callout is configured that modifies I-PDU data: Ensure that the APIs: Com_TriggerIPDUSend() and Com_SwitchIpduTxMode() and Com_TriggerTransmit() and Com_MainFunctionTx() do not interrupt each other for the very same I-PDU.
Rationale	<p>This issue could be avoided if you lock the PDU buffer or use expensive double buffers. However if you lock the PDU buffer while the callout function or the PduR_-ComTransmit function is called, it leads to an undefined locking time. It is not acceptable to disable interrupts for too long. Therefore a usage restriction has been defined in the work-around section to avoid race conditions.</p>

4.2.4.4.3. Com.EB.IntReq.Preemption02.TP

Description	<p>Restrictions to prevent race conditions and a undefined transmission behaviour in Com's Tx-path for large I-PDUs. The Com module exhibits several race conditions in its transmission path that can cause inconsistent and/or mutilated data to be transmitted as well as undefined transmission requests. The transmission of an large I-PDU can be initiated by a Tx-signal API (due to Com module Tp limitation only transmission mode DIRECT can be enabled) and is deferred to the next invocation of the Com_MainFunctionTx() (due to Com module Tp limitation all large Tx I-PDU transmission requests are deferred to the next execution of the Com transmission main function). The Tx-signal APIs are Com_SendSignal(), Com_SendDynSignal(), Com_SendSignalGroup(), and Com_SendSignalGroupArray(). The Tx-signal APIs have write access to the Com-internal I-PDU buffer. Additionally the transmission of an I-PDU can be initiated in context of Com_TriggerIPDUSend(). Depending on the underlying bus system, the API Com_TriggerTransmit() is invoked, which requires read access to the</p>
--------------------	---

Com-internal I-PDU buffer. A race occurs when an ongoing transmission (access to the Com-internal I-PDU buffer by Com lower layer and Com callout) is interrupted by an invocation of a Tx-signal API. A undefined transmission behaviour occurs when an ongoing transmission (execution of the Com_MainFunctionTx()) is interrupted by Com_TriggerIPDUSend() which possible triggers a transmission during the current execution of the Com_MainFunctionTx() and not the next invocation of the Com_MainFunctionTx(). This behaviour leads to the following cases:

- An large I-PDU has a direct part. It also has a call to a Tx-signal API to a signal/signal group, in which one of the following transfer properties is interrupted by another Tx-signal API call of a signal of the very same large I-PDU: TRIGGERED, TRIGGERED_ON_CHANGE, TRIGGERED_ON_CHANGE_WITHOUT_REPETITION, or TRIGGERED_WITHOUT_REPETITION.
- A call to a Tx-signal API interrupts a call to Com_MainFunctionTx().
- A call to Com_TriggerIPDUSend() interrupts a call to Com_MainFunctionTx().
- A callout uses the data of the large I-PDU for a calculation (e.g. to calculate a CRC) and a call to Tx-signal API interrupts the sending of the large I-PDU.

With a call to Com_SendDynSignal() not only the content of an large I-PDU may change, but also the length of the large I-PDU. Work-around To prevent inconsistencies in the large I-PDU, ensure the following:

- A call to a Tx-signal API that initiates a transmission does not interrupt a call to a Tx-signal API for signals which belong to the same large I-PDU.
- A call to a Tx-signal API does not interrupt one of the following APIs: Com_TriggerIPDUSend(), or Com_TriggerTransmit().
- A call to a Tx-signal API does not interrupt Com_MainFunctionTx().
- Additionally, if a callout is configured that modifies data of the large I-PDU: Ensure that the APIs: Com_TriggerIPDUSend()and Com_TriggerTransmit() and Com_MainFunctionTx() do not interrupt each other for the very same large I-PDU.

Rationale	This issue could be avoided if you lock the PDU buffer or use expensive double buffers. However if you lock the PDU buffer while the callout function or the PduR_-ComTpTransmit function is called, it leads to an undefined locking time. It is not acceptable to disable interrupts for too long. Therefore a usage restriction has been defined in the work-around section to avoid race conditions.
------------------	--

4.2.4.4.4. Com.EB.IntReq.Preemption03

Description	<p>The access to the shadow buffer of a signal group is not protected. Therefore restrictions apply to the mutually possible preemptions.</p> <ul style="list-style-type: none"> - On the Tx-side: A call to Com_UpdateShadowSignal() shall not get interrupted by Com_SendSignalGroup() for the signal group to which the group signal belongs to. - On the Rx side: A call to Com_ReceiveShadowSignal() shall not get interrupted by Com_ReceiveSignalGroup() for the signal group to which the group signal belongs to.
Rationale	<p>Restriction on allowed mutual preemptions Work-around:</p> <ul style="list-style-type: none"> - Ensure that Com_SendSignalGroup() does not interrupt Com_UpdateShadowSignal() for the signal group to which the group signal belongs to. - Ensure that Com_ReceiveSignalGroup() does not interrupt Com_ReceiveShadowSignal() for the signal group to which the group signal belongs to.

4.2.4.4.5. Com.EB.IntReq.MainRxSchedule04

Description	The Com_MainFunctionRx() shall be scheduled even if no Rx-I-PDU is configured on a certain ECU.
Rationale	Scheduling the Com main functions is restricted You always need to schedule the Com_MainFunctionRx() even if no Rx-I-PDU is configured on a certain ECU. The Com_MainFunctionRx() maintains the internal timer of the Com module. The internal timer is used as a time base for reception deadline monitoring but also for features of the Tx-path like transmission deadline monitoring, minimum delay timer and sending of Tx-I-PDUs (cyclic and n-times).

4.2.4.4.6. Com.EB.IntReq.UpdateBit05

Description	<p>Limitation on Com signals/signal groups with update-bits. AUTOSAR COM SWS specifies that signals/signal groups with update-bits which have not been updated shall be discarded. However, if after an update of an I-PDU the value of a signal changes from e.g. x to y without the update bit is set, a call to Com_ReceiveSignal()/Com_ReceiveSignalGroup()-Com_ReceiveGroupSignal() returns the changed value (i.e. y) and not the last received value (i.e. x). Note: It is very unlikely that the receiver receives an updated value without the update-bit set. Because at sender side, the sender always sets the update-bit in case a new value is transmitted. The value of a signal/signal group only changes when the Com_SendSignal()/Com_SendSignalGroup() is invoked which sets the update-bit. An impact may only occur if the value on the sender is changed while the update-bit is not set. If this conditions occur this has no impact on the following use-cases:</p> <ul style="list-style-type: none"> - For applications (SWCs), at least if the EB-optimization DirectReadFromCom in Rte is not used. Since the Rte reads the value from the Com module only if it is notified by the Com module. This does not happen when the update-bit is not set. Also it writes the received value into a buffer and reads requests from the application and uses the value of the buffer. - For applications which only use Com APIs when ComNotification is received. <p>However, this conditions may have an impact on the following use-case: Applications, which directly use the Com APIs, usually get the correct value, since the value of a signal usually does not change without setting the update-bit. If you use the Com APIs without ComNotification, changed values may be read that have no update-bit set. The following work-around is only applicable in this case. Work-around for signals of type U/SINT8/16/32 Configure a filter (ComFilterAlgorithm) NEW_IS_WITHIN, with the parameters [ComFilterMin, ComFilterMax] = maximum possible value range.</p>
Rationale	<p>This limitation allows a more efficient implementation and for the application usually the behavior does not change. Requirements: - COM324</p>

4.2.4.4.7. Com.EB.IntReq.Preemption06

Description	<p>Regarding the multiple main function support with gateway use cases, the generated Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>() and Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>_Dest_<MainFunctionName_Dest>() APIs for a related source main function shall be scheduled after (and thus non-concurrent to) the related source main function.</p>
--------------------	--

Rationale	Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>() and Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>_Dest_<MainFunctionName_Dest>() APIs perform a read access to signal buffers (filter flags) and ComIPdu buffers (which are written by Com_RxIndication() and the related source main function). However, by scheduling the Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>() and Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>_Dest_<MainFunctionName_Dest>() APIs after (and thus non-concurrent to) the related source main function, the concurrent access to data shared between the related source main function and the Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>() / Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>_Dest_<MainFunctionName_Dest>() APIs does not occur.
------------------	--

4.2.4.4.8. Com.EB.IntReq.Preemption07

Description	Regarding the multiple main function support with gateway use cases, the generated Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>() and Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>_Dest_<MainFunctionName_Dest>() APIs for a related source main function shall not be interrupted by similar generated APIs from different source main functions. Furthermore, the Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>() shall be called prior to every call to Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>_Dest_<MainFunctionName_Dest>() of a source main function.
Rationale	Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>() performs a write access to the ComIPdu buffers of the TX ComIPdus. Since Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>_Dest_<MainFunctionName_Dest>() (which triggers the TX ComIPdus) are potentially executed on different cores and thus theoretically multi-core capable mutual exclusion primitives are required. However, by scheduling the Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>() calls prior to every call of Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>_Dest_<MainFunctionName_Dest>() and avoiding interruption by similar generated APIs for different source main functions, the concurrent write access to the ComIPdu buffers of the TX ComIPdus does not occur. Due to restricted scheduling concurrent write access is avoided because any send action initiated in Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>() will be deferred till the execution of Com_MainFunctionRouteSignals_Src_<MainFunctionName_Src>_Dest_<MainFunctionName_Dest>().

4.3. IpduM

4.3.1. Configuration parameters

Containers included		
Container name	Multiplicity	Description
CommonPublishedInformation	1..1	Label: Common Published Information Common container, aggregated by all modules. It contains published information about vendor and versions.
IpduMDefensiveProgramming	1..1	Label: Defensive Programming Options Parameters for defensive programming
IpduMConfig	1..n	This container contains the sub containers of the IpduM module. The IpduMTxPathway sub container includes information about sent I-PDUs. The IpduMRxPathway includes information about received I-PDUs. This container is a MultipleConfigurationContainer, i.e. this container and its sub containers exist once per configuration set.
IpduMGeneral	1..1	Contains the general configuration parameters of IpduM.
IpduMPublishedInformation	1..1	Additional published parameters not covered by. CommonPublishedInformation container. Note that these parameters do not have any configuration class setting, since they are published information.
IpduMRequestMessageConfiguration	0..1	This is used to specify the configuration for multiplexed requesting messages.
PublishedInformation	1..1	Label: EB Published Information Additional published parameters not covered by CommonPublishedInformation container.

Parameters included	
Parameter name	Multiplicity
IMPLEMENTATION_CONFIG_VARIANT	1..1

Parameter Name	IMPLEMENTATION_CONFIG_VARIANT
Label	Config Variant
Multiplicity	1..1
Type	ENUMERATION

Default value	VariantPostBuild
Range	VariantPostBuild

4.3.1.1. CommonPublishedInformation

Parameters included	
Parameter name	Multiplicity
ArMajorVersion	1..1
ArMinorVersion	1..1
ArPatchVersion	1..1
SwMajorVersion	1..1
SwMinorVersion	1..1
SwPatchVersion	1..1
ModuleId	1..1
VendorId	1..1
Release	1..1

Parameter Name	ArMajorVersion
Label	AUTOSAR Major Version
Description	Major version number of AUTOSAR specification on which the appropriate implementation is based on.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	2
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	ArMinorVersion
Label	AUTOSAR Minor Version
Description	Minor version number of AUTOSAR specification on which the appropriate implementation is based on.
Multiplicity	1..1
Type	INTEGER_LABEL

Default value	2
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	ArPatchVersion
Label	AUTOSAR Patch Version
Description	Patch level version number of AUTOSAR specification on which the appropriate implementation is based on.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	0
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	SwMajorVersion
Label	Software Major Version
Description	Major version number of the vendor specific implementation of the module.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	3
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	SwMinorVersion
Label	Software Minor Version
Description	Minor version number of the vendor specific implementation of the module. The numbering is vendor specific.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	3
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	SwPatchVersion
-----------------------	-----------------------

Label	Software Patch Version
Description	Patch level version number of the vendor specific implementation of the module. The numbering is vendor specific.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	11
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	ModuleId
Label	Numeric Module ID
Description	Module ID of this module from Module List
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	52
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	VendorId
Label	Vendor ID
Description	Vendor ID of the dedicated implementation of this module according to the AUTOSAR vendor list
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	1
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	Release
Label	Release Information
Multiplicity	1..1
Type	STRING_LABEL
Default value	
Configuration class	PublishedInformation:

Origin	Elektrobit Automotive GmbH
---------------	----------------------------

4.3.1.2. IpduMDefensiveProgramming

Parameters included	
Parameter name	Multiplicity
IpduMDefProgEnabled	1..1
IpduMPrecondAssertEnabled	1..1
IpduMPostcondAssertEnabled	1..1
IpduMStaticAssertEnabled	1..1
IpduMUnreachAssertEnabled	1..1
IpduMInvariantAssertEnabled	1..1

Parameter Name	IpduMDefProgEnabled	
Label	Enable Defensive Programming	
Description	<p>Enables or disables the defensive programming feature for the module IpduM.</p> <p>Note: This feature is dependent on the use of the development error detection module. To use the defensive programming feature, proceed as follows:</p> <ol style="list-style-type: none"> 1. Enable development error detection 2. Enable defensive programming 3. Enable assertions as required 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMPrecondAssertEnabled
----------------	---------------------------

Label	Enable Precondition Assertions
Description	<p>Enables handling of precondition assertion checks reported from the module IpduM.</p> <p>Dependency on parameter(s):</p> <ul style="list-style-type: none"> ▶ Enable Development Error Detection (IpduMDevErrorDetect): must be enabled ▶ Enable Defensive Programming (IpduMDefProgEnabled): must be enabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	IpduMPostcondAssertEnabled
Label	Enable Postcondition Assertions
Description	<p>Enables handling of postcondition assertion checks reported from the module IpduM.</p> <p>Dependency on parameter(s):</p> <ul style="list-style-type: none"> ▶ Enable Development Error Detection (IpduMDevErrorDetect): must be enabled ▶ Enable Defensive Programming (IpduMDefProgEnabled): must be enabled
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	IpduMStaticAssertEnabled
Label	Enable Static Assertions
Description	<p>Enables handling of static assertion checks reported from the module IpduM.</p> <p>Dependency on parameter(s):</p> <ul style="list-style-type: none"> ▶ Enable Development Error Detection (IpduMDevErrorDetect): must be enabled

	► Enable Defensive Programming (IpduMDefProgEnabled): must be enabled	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMUnreachAssertEnabled	
Label	Enable Unreachable Code Assertions	
Description	<p>Enables handling of unreachable code assertion checks reported from the module IpduM.</p> <p>Dependency on parameter(s):</p> <ul style="list-style-type: none"> ► Enable Development Error Detection (IpduMDevErrorDetect): must be enabled ► Enable Defensive Programming (IpduMDefProgEnabled): must be enabled 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMInvariantAssertEnabled	
Label	Enable Invariant Assertions	
Description	<p>Enables handling of invariant assertion checks reported from functions of the module IpduM.</p> <p>Dependency on parameter(s):</p> <ul style="list-style-type: none"> ► Enable Development Error Detection (IpduMDevErrorDetect): must be enabled ► Enable Defensive Programming (IpduMDefProgEnabled): must be enabled 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	Elektrobit Automotive GmbH
---------------	----------------------------

4.3.1.3. IpduMConfig

Containers included		
Container name	Multiplicity	Description
IpduMContainedRxPdu	0..n	Configuration of a received contained Pdu.
IpduMContainedTxPdu	0..n	Configuration of a sender ContainedPdu.
IpduMContainerRxPdu	0..n	EN: Configuration of a receiver ContainerPdu which may collect several ContainedPdus.
IpduMContainerTxPdu	0..n	Configuration of a transmitted container Pdu.
IpduMRxPathway	0..65535	Contains the configuration parameters received I-PDUs by the IpduM module.
IpduMTxPathway	0..65535	Contains the configuration parameters transmitted I-PDUs by the IpduM module.

4.3.1.4. IpduMContainedRxPdu

Parameters included	
Parameter name	Multiplicity
IpduMContainedPdu-HeaderId	1..1
IpduMContainedRxIn-ContainerPduRef	0..1
IpduMContainedRxPduRef	1..1

Parameter Name	IpduMContainedPduHeaderId	
Description	Header Id which is part of the ContainerPdu when this ContainedPdu is inside.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainedRxInContainerPduRef	
Description	Optional reference to a container Pdu this contained Pdu may be transported in.	
Multiplicity	0..1	
Type	REFERENCE	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainedRxPduRef	
Description	Reference to the Pdu which represents this ContainedPdu and is used for reception indication.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.3.1.5. IpduMContainedTxPdu

Parameters included	
Parameter name	Multiplicity
IpduMContainedPdu-HeaderId	1..1
IpduMContainedTxPduCollectionSemantics	1..1
IpduMContainedTxPduConfirmation	0..1
IpduMContainedTxPduHandleId	1..1
IpduMContainedTxPduSendTimeout	0..1
IpduMContainedTxPduTrigger	1..1
IpduMContainedTxInContainerPduRef	1..1
IpduMContainedTxPduRef	1..1

Parameter Name	IpduMContainedPduHeaderId	
Description	Header Id which is part of the ContainerPdu when this ContainedPdu is inside.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainedTxPduCollectionSemantics	
Description	Defines whether this IpduMContainedTxPdu shall be collected using a last-is-best or queued semantics.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	IPDUM_COLLECT_LAST_IS_BEST IPDUM_COLLECT_QUEUED	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainedTxPduConfirmation	
Description	This Parameter determines whether for this contained I-PDU a TxConfirmation shall be provided. If set to TRUE a TxConfirmation is issued. It is not used when an I-PDU is requested using the trigger transmit API.	
Multiplicity	0..1	
Type	BOOLEAN	
Default value	false	
Configuration class	PreCompile:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainedTxPduHandleId	
Description	Handle Id of the ContainedPdu.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=65535 >=0	

Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainedTxPduSendTimeout	
Description	Defines a ContainedPdu specific sender timeout which can reduce the Container-Pdu timer when this ContainedPdu is put inside the ContainerPdu.	
Multiplicity	0..1	
Type	FLOAT	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainedTxPduTrigger	
Description	Defines whether this Pdu triggers the sending of the ContainerPdu.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	IPDUM_TRIGGER_ALWAYS	
	IPDUM_TRIGGER_NEVER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainedTxInContainerPduRef	
Description	Reference to the container Pdu which this contained Pdu shall be collected in.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainedTxPduRef	
Description	Reference to the Pdu which represents this ContainedPdu and is used for transmission.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	AUTOSAR_ECUC
--------	--------------

4.3.1.6. IpduMContainerRxPdu

Parameters included	
Parameter name	Multiplicity
IpduMContainerHeaderSize	1..1
IpduMContainerPduProcessing	1..1
IpduMContainerQueueSize	0..1
IpduMContainerRxAcceptContainedPdu	1..1
IpduMContainerRxHandleId	1..1
IpduMContainerRxPduRef	1..1

Parameter Name	IpduMContainerHeaderSize	
Description	Defines the layout of the header information (header id and length).	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	IPDUM_HEADERTYPE_SHORT	
Range	IPDUM_HEADERTYPE_LONG	
	IPDUM_HEADERTYPE_SHORT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainerPduProcessing
Description	Defines whether the handling of this ContainerPdu shall be done in the context of the caller (IMMEDIATE) or in the next IpduM main function (DEFERRED).
Multiplicity	1..1
Type	ENUMERATION

Default value	IPDUM_PROCESSING_IMMEDIATE	
Range	IPDUM_PROCESSING_DEFERRED	
	IPDUM_PROCESSING_IMMEDIATE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainerQueueSize	
Description	Defines a local queue for handling of each ContainerPdu.	
Multiplicity	0..1	
Type	INTEGER	
Default value	1	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainerRxAcceptContainedPdu	
Description	Defines for the received IpduMContainerRxPdu whether the list of referencing IpduMContainedRxPdus (via the reference IpduMContainedRxPduContainerRef) is a closed set.	
Multiplicity	1..1	
Type	ENUMERATION	
Range	IPDUM_ACCEPT_ALL	
	IPDUM_ACCEPT_CONFIGURED	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainerRxHandleId	
Description	EN: Handle Id used by the PduR for RxIndication.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=65535	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainerRxPduRef	
Description	Reference to the Pdu which represents the container and is used for reception.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.3.1.7. IpduMContainerTxPdu

Parameters included	
Parameter name	Multiplicity
IpduMContainerHeaderSize	1..1
IpduMContainerQueueSize	0..1
IpduMContainerTxConfirmationTimeout	0..1
IpduMContainerTxFirstContainedPduTrigger	1..1
IpduMContainerTxHandleId	0..1
IpduMContainerTxSendTimeout	0..1
IpduMContainerTxSizeThreshold	0..1
IpduMContainerTxTriggerMode	1..1
IpduMContainerTxPduRef	1..1

Parameter Name	IpduMContainerHeaderSize
Description	Defines the layout of the header information (header id and length).
Multiplicity	1..1
Type	ENUMERATION
Range	IPDUM_HEADERTYPE_LONG

	IPDUM_HEADERTYPE_SHORT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainerQueueSize	
Description	Defines a local queue for handling of each ContainerPdu.	
Multiplicity	0..1	
Type	INTEGER	
Default value	1	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainerTxConfirmationTimeout	
Description	This timeout (in seconds) defines the timeout period for monitoring the reception of the TxConfirmation. It is not used when an I-PDU is requested using the trigger transmit API.	
Multiplicity	0..1	
Type	FLOAT	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainerTxFirstContainedPduTrigger	
Description	Defines if the transmission of this IpduMContainerTxPdu shall be requested right after the first IpduMContainedTxPdu was put into it.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMContainerTxHandleId	
Description	Handle Id used by the PduR for TxConfirmation and for TriggerTransmit of the ContainerPdu.	
Multiplicity	0..1	

Type	INTEGER
Range	<=65535
	>=0
Configuration class	PreCompile: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMContainerTxSendTimeout
Description	When this timeout expires the ContainerPdu is triggered for sending. The respective timer is started when the first Pdu is put into the ContainerPdu.
Multiplicity	0..1
Type	FLOAT
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMContainerTxSizeThreshold
Description	Defines the size threshold in bytes which, when exceeded, triggers the sending of the ContainerPdu although the maximum Pdu size (PduLength parameter of Pdu object) has not been reached yet.
Multiplicity	0..1
Type	INTEGER
Range	<=4294967295
	>=0
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMContainerTxTriggerMode
Description	Defines whether this ContainerPdu is fetched via trigger transmit.
Multiplicity	1..1
Type	ENUMERATION
Range	IPDUM_DIRECT
	IPDUM_TRIGGERTRANSMIT
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMContainerTxPduRef	
Description	Reference to the Pdu which represents the container and is used for transmission.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.3.1.8. IpduMRxPathway

Containers included		
Container name	Multiplicity	Description
IpduMRxIndication	1..1	Contains the configuration for incoming RxIndication calls.

4.3.1.9. IpduMRxIndication

Containers included		
Container name	Multiplicity	Description
IpduMRxDynamicPart	1..256	This container contains the configuration for the dynamic part of incoming RxIndication calls. When an incoming received I-PDU's selector field matches the IpduM_Selector_Value, the new outgoing I-PDU for the dynamic part is constructed as defined by the segments of this container and sent out with the I-PDU ID referenced by IpduMOutgoingDynamicPduRef.
IpduMRxStaticPart	0..1	This container contains the configuration for the static part of incoming RxIndication calls. On reception, the new outgoing I-PDU for the static part is constructed as defined by the segments of this container and sent out with the I-PDU ID referenced by IpduMOutgoingStaticPduRef.
IpduMSelectorFieldPosition	1..1	This contains the location and the length of the selector field.

Parameters included	
Parameter name	Multiplicity
IpduMByteOrder	1..1

Parameters included	
IpduMRxHandleId	1..1
IpduMRxIndication-PduRef	1..1

Parameter Name	IpduMByteOrder	
Description	This parameter defines the ByteOrder for all IpduMSegments (static and dynamic part) and for the selectorField within the MultiplexedPdu. The absolute position of a segment in the MultiplexedIPdu is determined by the definition of the ByteOrder parameter: If BIG_ENDIAN is specified, the SegmentPosition indicates the bit position of the most significant bit in an IPDU. If LITTLE_ENDIAN is specified, the SegmentPosition indicates the bit position of the least significant bit in an IPDU.	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	LITTLE_ENDIAN	
Range	BIG_ENDIAN	
	LITTLE_ENDIAN	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMRxHandleId	
Description	This is the I-PDU ID of the incoming I-PDU. If an incoming RxIndication's I-PDU ID matches this value than it is unpacked according to the specification in this container.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=65535	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMRxIndicationPduRef	
Description	Reference to the received PDU representation in the ECU Configuration Description exchange file.	
Multiplicity	1..1	

Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.3.1.10. IpduMRxDynamicPart

Containers included		
Container name	Multiplicity	Description
IpduMSegment	1..n	This contains the location and the length of a segment. A segment must fit inside the I-PDU. The segment in the source I-PDU that is located at the IpduMSegmentPosition is copied to the same position in the destination I-PDU.

Parameters included	
Parameter name	Multiplicity
IpduMRxSelectorValue	1..1
IpduMOutgoingDynamicPduRef	1..1

Parameter Name	IpduMRxSelectorValue
Description	This is the selector value that this container refers to.
Multiplicity	1..1
Type	INTEGER
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMOutgoingDynamicPduRef
Description	When the new I-PDU is sent out it is sent with this I-PDU ID. Reference to the sent PDU representation in the ECU Configuration Description exchange file.
Multiplicity	1..1
Type	REFERENCE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

4.3.1.11. IpduMSegment

Parameters included	
Parameter name	Multiplicity
IpduMSegmentLength	1..1
IpduMSegmentPosition	1..1
IpduMDestinationBit	0..1

Parameter Name	IpduMSegmentLength	
Description	Length of the segment in bits.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=2032	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMSegmentPosition	
Description	Segments bit position in the multiplexed PDU.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMDestinationBit
Description	<p>Warning: this parameter is only available in this release for backward compatibility with IpduM ASR 3.x. For new configurations this parameter shall not be used any more. In ASR 4.x this parameter has been removed as IpduMSegmentPosition describes the position in both multiplexed and de-multiplexed Pdu.</p> <p>Note: when importing a communication matrix, please select the option which imports an ASR 4.x conform configuration.</p> <p>Note: Changing the configuration to an ASR 4.x conform configuration requires also an update of the configuration of the Com module.</p> <p>Bit position in an I-PDU of the start of the destination bit field for the copy. The resulting destination field must fit inside the I-PDU.</p>

Multiplicity	0..1
Type	INTEGER
Configuration class	PostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

4.3.1.12. IpduMRxStaticPart

Containers included		
Container name	Multiplicity	Description
IpduMSegment	1..n	This contains the location and the length of a segment. A segment must fit inside the I-PDU. The segment in the source I-PDU that is located at the IpduMSegmentPosition is copied to the same position in the destination I-PDU.

Parameters included	
Parameter name	Multiplicity
IpduMOutgoingStaticPduRef	1..1

Parameter Name	IpduMOutgoingStaticPduRef
Description	When the new I-PDU is sent out it is sent with this I-PDU ID. Reference to the sent PDU representation in the ECU Configuration Description exchange file.
Multiplicity	1..1
Type	REFERENCE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

4.3.1.13. IpduMSegment

Parameters included	
Parameter name	Multiplicity
IpduMSegmentLength	1..1
IpduMSegmentPosition	1..1

Parameters included	
IpduMDestinationBit	0..1

Parameter Name	IpduMSegmentLength	
Description	Length of the segment in bits.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=2032	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMSegmentPosition	
Description	Segments bit position in the multiplexed PDU.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMDestinationBit	
Description	<p>Warning: this parameter is only available in this release for backward compatibility with IpduM ASR 3.x. For new configurations this parameter shall not be used any more. In ASR 4.x this parameter has been removed as IpduMSegmentPosition describes the position in both multiplexed and de-multiplexed Pdu.</p> <p>Note: when importing a communication matrix, please select the option which imports an ASR 4.x conform configuration.</p> <p>Note: Changing the configuration to an ASR 4.x conform configuration requires also an update of the configuration of the Com module.</p> <p>Bit position in an I-PDU of the start of the destination bit field for the copy. The resulting destination field must fit inside the I-PDU.</p>	
Multiplicity	0..1	
Type	INTEGER	
Configuration class	PostBuild:	VariantPostBuild

Origin	Elektrobit Automotive GmbH
---------------	----------------------------

4.3.1.14. IpduMSelectorFieldPosition

Parameters included	
Parameter name	Multiplicity
IpduMSelectorFieldLength	1..1
IpduMSelectorFieldPosition	1..1

Parameter Name	IpduMSelectorFieldLength	
Description	Length of the selector field in bits.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=8	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMSelectorFieldPosition	
Description	Selector field bit position in the multiplexed PDU.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.3.1.15. IpduMTxPathway

Containers included		
Container name	Multiplicity	Description
IpduMTxRequest	1..1	This is used to specify the configuration for Transmit requests. There will one instance of this container for each I-PDU that

Containers included		
		can be requested for transmission (the outgoing I-PDUs) by the IpduM.

4.3.1.16. IpduMTxRequest

Containers included		
Container name	Multiplicity	Description
IpduMSelectorFieldPosition	1..1	This contains the location and the length of the selector field.
IpduMTxDynamicPart	1..256	Configuration parameters for an instance of a TxRequest call into the IpduM. When a Tx Request with the IpduMTxDynamicHandleId is received by the IpduM, all segments as defined by this container are copied from the incoming I-PDU into the outgoing I-PDU buffer and then the send mode honoured. This container is used by the dynamic part of a TxRequest configuration. Therefore, for each outgoing I-PDU there will be one instance of this container for the dynamic part.
IpduMTxStaticPart	0..1	Configuration parameters for an instance of a TxRequest call into the IpduM. When a Tx Request with the IpduMTxStaticHandleId is received by the IpduM, all segments as defined by this container are copied from the incoming I-PDU into the outgoing I-PDU buffer and then the send mode honoured. This container is used for the static part of a TxRequest configuration. Therefore, for each outgoing I-PDU there will be one instance of this container for the static part if it exists.

Parameters included	
Parameter name	Multiplicity
IpduMByteOrder	1..1
IpduMIPduUnusedAreasDefault	0..1
IpduMTxConfirmation-PduId	1..1
IpduMTxConfirmation-Timeout	0..1
IpduMTxTriggerMode	1..1

Parameters included	
IpduMInitialDynamicPart	1..1
IpduMOutgoingPduRef	1..1
IpduMQueueSize	1..1

Parameter Name	IpduMByteOrder	
Description	This parameter defines the ByteOrder for all IpduMSegments (static and dynamic part) and for the selectorField within the MultiplexedPdu. The absolute position of a segment in the MultiplexedIPdu is determined by the definition of the ByteOrder parameter: If BIG_ENDIAN is specified, the SegmentPosition indicates the bit position of the most significant bit in an IPDU. If LITTLE_ENDIAN is specified, the SegmentPosition indicates the bit position of the least significant bit in an IPDU.	
Multiplicity	1..1	
Type	ENUMERATION	
Default value	LITTLE_ENDIAN	
Range	BIG_ENDIAN	
	LITTLE_ENDIAN	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMIPduUnusedAreasDefault	
Description	IpduM module fills not used areas of an I-PDU with this bit-pattern. If this attribute is omitted the IpduM module does not fill the I-PDU.	
Multiplicity	0..1	
Type	INTEGER	
Range	<=255	
	>=0	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMTxConfirmationPduld	
Description	The handle Id to be used by the PduR to confirm the transmission of this PDU. The existence of this parameter is essential for the PduR generation tool to actually find a symbolicNameValue for the OutgoingPdu.	
Multiplicity	1..1	

Type	INTEGER
Range	<=65535
	>=0
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMTxConfirmationTimeout
Description	This timeout (in seconds) defines the timeout period for monitoring the reception of the TxConfirmation. It is not used when an I-PDU is requested using the trigger transmit API.
Multiplicity	0..1
Type	FLOAT
Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMTxTriggerMode
Description	Selects whether to send the multiplexed I-PDU immediately or at some later date.
Multiplicity	1..1
Type	ENUMERATION
Range	DYNAMIC_PART_TRIGGER
	NONE
	STATIC_OR_DYNAMIC_PART_TRIGGER
	STATIC_PART_TRIGGER
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMInitialDynamicPart
Description	Reference to the dynamic part that shall be used to initialize this multiplexed TX-I-PDU.
Multiplicity	1..1
Type	REFERENCE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMOutgoingPduRef	
Description	Reference to the PDU defining the outgoing I-PDU. When the outgoing I-PDU is sent this is the I-PDU ID to give it. It is the IpduM I-PDU ID of the assembled I-PDU.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMQueueSize	
Description	This value is specifies the queue size. A value of '0' means not using a queue at all.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Configuration class	Link:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

4.3.1.17. IpduMSelectorFieldPosition

Parameters included	
Parameter name	Multiplicity
IpduMSelectorFieldLength	1..1
IpduMSelectorFieldPosition	1..1

Parameter Name	IpduMSelectorFieldLength	
Description	Length of the selector field in bits.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=8	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	AUTOSAR_ECUC	
Parameter Name	IpduMSelectorFieldPosition	
Description	Selector field bit position in the multiplexed PDU.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.3.1.18. IpduMTxDynamicPart

Containers included		
Container name	Multiplicity	Description
IpduMSegment	1..n	This contains the location and the length of a segment. A segment must fit inside the I-PDU. The segment in the source I-PDU that is located at the IpduMSegmentPosition is copied to the same position in the destination I-PDU.

Parameters included	
Parameter name	Multiplicity
IpduMJitUpdate	0..1
IpduMTxDynamicConfirmation	1..1
IpduMTxDynamicHandleId	1..1
IpduMTxDynamicPduRef	1..1
IpduMTxDynamicPriority	1..1
IpduMTxSelectorValue	1..1

Parameter Name	IpduMJitUpdate
Description	If configured to true fetch the data of this part Just-In-Time via the triggerTransmit API of the PduR.
Multiplicity	0..1
Type	BOOLEAN
Default value	false

Configuration class	PostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMTxDynamicConfirmation
Description	A transmit request can be confirmed by the lower layer. If this parameter is set to true a confirmation of the I-PDU in COM representing the dynamic part is generated.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMTxDynamicHandleId
Description	This is an incoming handle id. When the handle of an incoming Tx Request matches this, the bits fields (see IPduMSegment) are copied and the IpduMTx-TriggerMode is honored.
Multiplicity	1..1
Type	INTEGER
Range	<div><=65535</div> <div>>=0</div>
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMTxDynamicPduRef
Description	Reference to the PDU representation in the ECU Configuration Description exchange file to be transmitted.
Multiplicity	1..1
Type	REFERENCE
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMTxDynamicPriority
Description	The priority of each HandleId. 0 is the highest priority
Multiplicity	1..1

Type	INTEGER	
Default value	0	
Range	<=127	
	>=0	
Configuration class	PostBuild:	VariantPostBuild

Parameter Name	IpduMTxSelectorValue	
Description	If IpduMTxAutomaticSelector is enabled IpduMTxSelectorValue defines the selector value which is set for this transmit PDU.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	PostBuild:	VariantPostBuild

4.3.1.19. IpduMSegment

Parameters included	
Parameter name	Multiplicity
IpduMSegmentLength	1..1
IpduMSegmentPosition	1..1
IpduMDestinationBit	0..1

Parameter Name	IpduMSegmentLength	
Description	Length of the segment in bits.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=2032	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMSegmentPosition	
Description	Segments bit position in the multiplexed PDU.	
Multiplicity	1..1	
Type	INTEGER	

Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMDestinationBit	
Description	<p>Warning: this parameter is only available in this release for backward compatibility with IpduM ASR 3.x. For new configurations this parameter shall not be used any more. In ASR 4.x this parameter has been removed as IpduMSegmentPosition describes the position in both multiplexed and de-multiplexed Pdu.</p> <p>Note: when importing a communication matrix, please select the option which imports an ASR 4.x conform configuration.</p> <p>Note: Changing the configuration to an ASR 4.x conform configuration requires also an update of the configuration of the Com module.</p> <p>Bit position in an I-PDU of the start of the destination bit field for the copy. The resulting destination field must fit inside the I-PDU.</p>	
Multiplicity	0..1	
Type	INTEGER	
Configuration class	PostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

4.3.1.20. IpduMTxStaticPart

Containers included		
Container name	Multiplicity	Description
IpduMSegment	1..n	This contains the location and the length of a segment. A segment must fit inside the I-PDU. The segment in the source I-PDU that is located at the IpduMSegmentPosition is copied to the same position in the destination I-PDU.

Parameters included	
Parameter name	Multiplicity
IpduMJitUpdate	0..1
IpduMTxStaticConfirmation	1..1
IpduMTxStaticHandleId	1..1
IpduMTxStaticPduRef	1..1

Parameter Name	IpduMJitUpdate	
Description	If configured to true fetch the data of this part Just-In-Time via the triggerTransmit API of the PduR.	
Multiplicity	0..1	
Type	BOOLEAN	
Default value	false	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMTxStaticConfirmation	
Description	A transmit request can be confirmed by the lower layer. If this parameter is set to true a confirmation of the I-PDU in COM representing the static part is generated.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMTxStaticHandleId	
Description	This is an incoming handle id. When the handle of an incoming Tx Request matches this, the bits fields (see IPduMSegment) are copied and the IpduMTx-TriggerMode is honored.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=65535	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMTxStaticPduRef	
Description	Reference to the PDU representation in the ECU Configuration Description exchange file to be transmitted.	
Multiplicity	1..1	
Type	REFERENCE	

Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.3.1.21. IpduMSegment

Parameters included	
Parameter name	Multiplicity
IpduMSegmentLength	1..1
IpduMSegmentPosition	1..1
IpduMDestinationBit	0..1

Parameter Name	IpduMSegmentLength	
Description	Length of the segment in bits.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=2032	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMSegmentPosition	
Description	Segments bit position in the multiplexed PDU.	
Multiplicity	1..1	
Type	INTEGER	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	IpduMDestinationBit
Description	<p>Warning: this parameter is only available in this release for backward compatibility with IpduM ASR 3.x. For new configurations this parameter shall not be used any more. In ASR 4.x this parameter has been removed as IpduMSegmentPosition describes the position in both multiplexed and de-multiplexed Pdu.</p> <p>Note: when importing a communication matrix, please select the option which imports an ASR 4.x conform configuration.</p>

	<p>Note: Changing the configuration to an ASR 4.x conform configuration requires also an update of the configuration of the Com module.</p> <p>Bit position in an I-PDU of the start of the destination bit field for the copy. The resulting destination field must fit inside the I-PDU.</p>	
Multiplicity	0..1	
Type	INTEGER	
Configuration class	PostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

4.3.1.22. IpduMGeneral

Containers included		
Container name	Multiplicity	Description
IpduMRxProcessing	0..n	Configuration of a dedicated RX MainFunction. The name of the generated function uses the pattern <code>IpduM_MainFunctionRx_-"Short-Name"</code> .
IpduMTxProcessing	0..n	Configuration of a dedicated TX MainFunction. The name of the generated function uses the pattern <code>IpduM_MainFunctionTx_-"Short-Name"</code> .

Parameters included	
Parameter name	Multiplicity
IpduMRxTimeBase	1..1
IpduMTxTimeBase	1..1
IpduMDevErrorDetect	1..1
IpduMZeroCopy	1..1
IpduMByteCopy	1..1
IpduMDynamic-PartQueue	1..1
IpduMTxAutomaticSelector	1..1
IpduMDataMemSize	0..1
IpduMInitialization-BySignalValue	1..1

Parameters included	
IpduMEnableJitUpdate	1..1
IpduMDedicat- edIpduProcessingSup- port	1..1
IpduMContainerP- duHandlingEnable	1..1
IpduMContainerQueuein- gRx	1..1
IpduMContainerQueue- ingTx	1..1
IpduMContained Collec- tQueuedTx	1..1
IpduMMaxContain- erTxLength	1..1
IpduMMaxContain- erRxLength	1..1
IpduMDequeueInTx- Conf	1..1
IpduMRelocatablePbcf- gEnable	1..1
IpduMHeaderByteOrder	0..1
IpduMStaticPartExists	1..1
IpduMVersionInfoApi	1..1

Parameter Name	IpduMRxTimeBase
Description	The period between successive calls to IpduM_MainFunctionRx in seconds. This parameter may be used by the IpduM generator to transform the values of the reception related timing configuration parameters of the IpduM module to internal implementation specific counter or tick values. The IpduM module's internal timing handling is implementation specific. The IpduM module (generator) may rely on the fact that IpduM_MainFunctionRx is scheduled according to the value configured here.
Multiplicity	1..1
Type	FLOAT
Default value	0.005
Range	>=0

Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	IpduMTxTimeBase
Description	The period between successive calls to IpduM_MainFunctionTx in seconds. This parameter may be used by the IpduM generator to transform the values of the reception related timing configuration parameters of the IpduM module to internal implementation specific counter or tick values. The IpduM module's internal timing handling is implementation specific. The IpduM module (generator) may rely on the fact that IpduM_MainFunctionTx is scheduled according to the value configured here.
Multiplicity	1..1
Type	FLOAT
Default value	0.005
Range	>=0
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	IpduMDevErrorDetect
Description	<p>Active/Deactivate the detection of development errors, for production code this parameter has to be False.</p> <ul style="list-style-type: none"> ▶ <code>True</code>: error detection activated ▶ <code>False</code>: error detection deactivated <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. ▶ Execution time reduction (code): Disabling this parameter reduces the execution time of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	IpduMZeroCopy
-----------------------	----------------------

Description	<p>Do not allocate memory for data in the IpduM. Only pointers for static and dynamic parts will be passed. Zero copy is only in some circumstances possible. Please refer the users guide.</p> <p>Destination bits will be taken as zero, if this parameter is set.</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): Enabling this parameter reduces the execution time of the module code. ▶ ROM reduction (code): Enabling this parameter reduces the ROM consumption of the module code. ▶ ROM reduction (config): Enabling this parameter reduces the ROM consumption of the module configuration. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMByteCopy	
Description	<p>Use byte-wise copy routines. Only possible if static and dynamic part is already byte-aligned.</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): Enabling this parameter reduces the execution time of the module code. ▶ ROM reduction (code): Enabling this parameter reduces the ROM consumption of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMDynamicPartQueue	
Description	<p>This specifies if queuing is enabled for dynamic PDUs.</p> <p>Optimization Effect:</p>	

	<ul style="list-style-type: none"> ▶ Execution time reduction (code): Disabling this parameter reduces the execution time of the module code. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. ▶ ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMTxAutomaticSelector	
Description	<p>If enabled the selector values for the transmit PDUs are set by the IpduM. If disabled the selector value is not set by the IpduM.</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): Disabling this parameter reduces the execution time of the module code. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. ▶ ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMDataMemSize	
Description	<p>Size of internal IpduM data in units of bytes (static memory allocation) - Memory required by post-build config must be smaller than this constant</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ RAM reduction (config): Decreasing this parameter reduces the RAM consumption of the module configuration. 	

Multiplicity	0..1
Type	INTEGER
Configuration class	Link: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	IpduMInitializationBySignalValue
Description	If IpduMInitializationBySignalValue is enabled, the static and dynamic parts are initialized in retrieving signal values from the upper layer module by IpduM_Init. If IpduMInitializationBySignalValue is disabled the static and dynamic parts are only initialized by the unused area pattern configured.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	IpduMEnableJitUpdate
Description	If IpduMEnableJitUpdate is enabled, the initial dynamic part is used for JIT update.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	IpduMDedicatedIpduProcessingSupport
Description	<p>Enable the mapping of Containers/TxPathways to specific MainFunctions.</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): Disabling this parameter reduces the execution time of the module code. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	false

Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMContainerPduHandlingEnable	
Description	<p>Enables or disables the Multiple-PDU-to-Container handling.</p> <ul style="list-style-type: none"> ▶ True: Multiple-PDU-to-Container handling is enabled. ▶ False: Multiple-PDU-to-Container handling is disabled. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. ▶ Execution time reduction (code): Disabling this parameter reduces the execution time of the module code. ▶ ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. ▶ RAM reduction (code): Disabling this parameter reduces the RAM consumption of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	node:exists(..../IpduMConfig/*[1]/IpduMContainerTxPdu/*) or node:exists(..../IpduMConfig/*[1]/IpduMContainerRxPdu/*)	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMContainerQueuingRx	
Description	<p>Enables or disables the queuing of Multiple-PDU-to-Container PDUs during reception.</p> <ul style="list-style-type: none"> ▶ True: Multiple-PDU-to-Container queuing is enabled (RX). ▶ False: Multiple-PDU-to-Container queuing is disabled (RX). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. ▶ Execution time reduction (code): Disabling this parameter reduces the execution time of the module code. 	

	<ul style="list-style-type: none"> ▶ ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. ▶ RAM reduction (code): Disabling this parameter reduces the RAM consumption of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	PreCompile: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	IpduMContainerQueuingTx
Description	<p>Enables or disables the queuing of Multiple-PDU-to-Container PDUs during transmission.</p> <ul style="list-style-type: none"> ▶ True: Multiple-PDU-to-Container queuing is enabled (TX). ▶ False: Multiple-PDU-to-Container queuing is disabled (TX). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. ▶ Execution time reduction (code): Disabling this parameter reduces the execution time of the module code. ▶ ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. ▶ RAM reduction (code): Disabling this parameter reduces the RAM consumption of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	PreCompile: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	IpduMContainedCollectQueuedTx
Description	<p>Enables or disables the handling of Contained PDUs with IpduMContainedTxPduCollectionSemantics set to IPDUM_COLLECT_QUEUED.</p> <ul style="list-style-type: none"> ▶ True: IPDUM_COLLECT_QUEUED is supported.

	<p>► <code>False</code>: Only <code>IPDUM_COLLECT_LAST_IS_BEST</code> is supported.</p> <p>Optimization Effect:</p> <p>► ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.</p> <p>► Execution time reduction (code): Disabling this parameter reduces the execution time of the module code.</p> <p>► ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration.</p> <p>► RAM reduction (code): Disabling this parameter reduces the RAM consumption of the module code.</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMMaxContainerTxLength	
Description	This value is specifies the length of the buffer allocated on the stack during the transmission of Container PDUs. The value must be equal to the length of the largest referenced Pdu.	
Multiplicity	1..1	
Type	INTEGER	
Default value	64	
Range	>=0	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMMaxContainerRxLength	
Description	This value is specifies the length of the buffer allocated on the stack during the reception of Container PDUs. The value must be equal to the length of the largest referenced Pdu.	
Multiplicity	1..1	
Type	INTEGER	
Default value	64	

Range	>=0	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMDequeueInTxConf	
Description	Enables or disables dequeuing (transmission) in addition in the context of an <code>IpduM_TxConfirmation()</code> call. Based on the value, transmission would occur: <ul style="list-style-type: none"> ▶ True: additionally in <code>IpduM_TxConfirmation()</code> ▶ False: only in <code>IpduM_MainFunctionTx()</code> 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMRelocatablePbcfgEnable	
Description	Enables/disable support for relocatable postbuild configuration. <ul style="list-style-type: none"> ▶ True: Postbuild configuration relocatable in memory. ▶ False: Postbuild configuration not relocatable in memory. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMHeaderByteOrder	
Description	This parameter defines the ByteOrder of the headers inside a Container I-PDU.	
Multiplicity	0..1	
Type	ENUMERATION	
Range	IPDUM_BIG_ENDIAN	
	IPDUM_LITTLE_ENDIAN	
Configuration class	PreCompile:	VariantPostBuild

Origin	AUTOSAR_ECUC	
Parameter Name	IpduMStaticPartExists	
Description	<p>This is to allow optimizations in the case the IpduM will never be used with a static part.</p> <p>Note that this is a pre-compile option. If this is set to False then it will not be possible to add static parts after compilation.</p> <ul style="list-style-type: none"> ▶ True: A static part may exist. ▶ False: A static part will never exist. <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ Execution time reduction (code): Disabling this parameter reduces the execution time of the module code. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. ▶ ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	
Parameter Name	IpduMVersionInfoApi	
Description	<p>Active/Deactivate the version information API.</p> <ul style="list-style-type: none"> ▶ true: version information activated ▶ false: version information deactivated <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	AUTOSAR_ECUC
--------	--------------

4.3.1.23. IpduMRxProcessing

Parameters included	
Parameter name	Multiplicity
IpduMTimeBase	1..1
IpduMPartitionRef	1..1
IpduMRxPduRef	1..n

Parameter Name	IpduMTimeBase	
Description	TimeBase for this specific MainFunction in seconds (similar to IpduMRxTimeBase).	
Multiplicity	1..1	
Type	FLOAT	
Range	>=0	
	<=3600	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMPartitionRef	
Description	Reference to EcucPartition to allow for grouping of MainFunction according to EcucPartition elements.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMRxPduRef	
Description	Reference to IpduMContainerRxPdu which is assigned to this MainFunction.	
Multiplicity	1..n	
Type	REFERENCE	
Configuration class	PostBuild:	VariantPostBuild

Origin	Elektrobit Automotive GmbH
---------------	----------------------------

4.3.1.24. IpduMTxProcessing

Parameters included	
Parameter name	Multiplicity
IpduMTimeBase	1..1
IpduMPartitionRef	1..1
IpduMTxPduRef	1..n

Parameter Name	IpduMTimeBase	
Description	TimeBase for this specific MainFunction in seconds (similar to IpduMTxTime-Base).	
Multiplicity	1..1	
Type	FLOAT	
Range	>=0	
	<=3600	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMPartitionRef	
Description	Reference to EcucPartition to allow for grouping of MainFunction according to EcucPartition elements.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMTxPduRef	
Description	Reference to IpduMContainerTxPdu/IpduMTxPathway which is assigned to this MainFunction.	
Multiplicity	1..n	
Type	CHOICE-REFERENCE	
Configuration class	PostBuild:	VariantPostBuild

Origin	Elektrobit Automotive GmbH
---------------	----------------------------

4.3.1.25. IpduMPublishedInformation

Parameters included	
Parameter name	Multiplicity
IpduMRxDirectComInvocation	1..1

Parameter Name	IpduMRxDirectComInvocation
Description	If set to TRUE the COM invocation optimization as defined in. IPDUM140 is implemented.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

4.3.1.26. IpduMRequestMessageConfiguration

Containers included		
Container name	Multiplicity	Description
IpduMRequestMessageMapInfo	1..n	This is used to specify the mapping information.

Parameters included	
Parameter name	Multiplicity
IpduMRequestMessageIdLength	1..1
IpduMRequestMessageIdBytePos	1..1

Parameter Name	IpduMRequestMessageIdLength
Description	Defines the length of the data field (number of bytes) in the received message from which the requested message ID is to be extracted starting from IPDUM_REQUEST_MESSAGE_ID_BYTEPOS.

	Optimization Effect: <ul style="list-style-type: none"> ▶ Execution time reduction (code): Decreasing this parameter reduces the execution time of the module code. ▶ ROM reduction (code): Decreasing this parameter reduces the ROM consumption of the module code. ▶ RAM reduction (code): Decreasing this parameter reduces the RAM consumption of the module code. 	
Multiplicity	1..1	
Type	INTEGER	
Range	<=4	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMRequestMessageldBytePos	
Description	Defines the Byte position of the data field in the received message from which the requested message ID is to be extracted.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=7	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

4.3.1.27. IpduMRequestMessageMapInfo

Parameters included	
Parameter name	Multiplicity
IpduMRequestedMessageld	1..1
IpduMRequestedMessagePduRef	1..1

Parameter Name	IpduMRequestedMessageld
-----------------------	--------------------------------

Description	Defines the requested message ID, which is sent from the requestor. This requested message ID is matched with the requested message ID extracted from the data field in the received message. This will be later mapped to IPDUM_REQUESTED_MESSAGE_PDU_REF.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=4294967295	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	IpduMRequestedMessagePduRef	
Description	Reference to the global EcuC Pdu (defined in EcuC's PduCollection) that corresponds to the Com I-Pdu that shall be triggered.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

4.3.1.28. PublishedInformation

Parameters included	
Parameter name	Multiplicity
PbcfgMSupport	1..1

Parameter Name	PbcfgMSupport	
Label	PbcfgM support	
Description	Specifies whether or not the IpduM can use the PbcfgM module for post-build support.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	PublishedInformation:	
Origin	Elektrobit Automotive GmbH	

4.3.2. Application programming interface (API)

4.3.2.1. Macro constants

4.3.2.1.1. IPDUM_E_GLOBAL_ECUID

Purpose	Development Error Code.
Value	0x80
Description	Global EcuID unknown to ECU.

4.3.2.1.2. IPDUM_SID_PROCESS_REQUEST_PDU

Purpose	Service Message API service ID.
Value	0x20
Description	Definition of service ID for IpduM_ProcessRequestPdu.

4.3.2.2. Functions

4.3.2.2.1. IpduM_GetVersionInfo

Purpose	Return module version information.	
Synopsis	<pre>void IpduM_GetVersionInfo (Std_VersionInfoType *const versioninfo);</pre>	
Service ID	0x01	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (out)	versioninfo	Version information
Description	This function returns the IpduM version information in the memory area versioninfo references. Preconditions:	

► The parameter versioninfo may not be a NULL pointer

4.3.2.2.2. IpduM_Init

Purpose	Initializes all module-related global variables.	
Synopsis	<pre>void IpduM_Init (const IpduM_ConfigType * Config);</pre>	
Service ID	0x00	
Sync/Async	synchronous	
Reentrancy	non reentrant	
Parameters (in)	Config	Pointer to post build configuration of the IpduM
Description	Initializes all module-related global variables including default values, default selector field and state of timeout monitors.	

4.3.2.2.3. IpduM_MainFunctionRx

Purpose	IpduM RX main function.	
Synopsis	<pre>void IpduM_MainFunctionRx (void);</pre>	
Service ID	0x11	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Description	Processes DEFERRED ContainerRxPdus This function has to be called periodically by a task controlled by the BSW scheduler.	

4.3.2.2.4. IpduM_MainFunctionTx

Purpose	IpduM TX main function.	
Synopsis	<pre>void IpduM_MainFunctionTx (void);</pre>	
Service ID	0x12	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	

Description	<p>Performs the processes of the activities that are not directly initiated by the calls from PDU-R. This includes at least the TxConfirmation time observation.</p> <p>This function has to be called periodically by a task controlled by the BSW scheduler.</p>
--------------------	--

4.3.2.2.5. IpduM_ProcessRequestPdu

Purpose	Process a request PDU.	
Synopsis	<pre>boolean IpduM_ProcessRequestPdu (PduIdType PduRx- PduId , const PduInfoType * RxRequestPduInfoPtr);</pre>	
Service ID	0x20	
Sync/Async	Synchronous	
Reentrancy	Non-Reentrant	
Parameters (in)	PduRxPduId	Unused parameter
	RxRequestPduInfoPtr	The PDU data which contains the service message ID
Return Value	Returns always FALSE to avoid further processing by Com	
Description	<p>This function has to be called as a Com call out function for a Com Rx-PDU that is a requesting message. The function extracts the requested service message ID from the data of the PDU and triggers the sending of the appropriate Com PDU via Com_TriggerIPDUSend().</p> <p>Preconditions:</p> <ul style="list-style-type: none"> ► The parameter SduDataPtr may not be a NULL pointer 	

4.3.2.2.6. IpduM_RxIndication

Purpose	Receive indication callback function.	
Synopsis	<pre>void IpduM_RxIndication (PduIdType Rx- PduId , PduInfoType * PduInfoPtr);</pre>	
Service ID	0x42	
Sync/Async	synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId	
Parameters (in)	RxPduId	ID of I-PDU that has been received.

	PduInfoPtr	Contains the length (SduLength) of the received I-PDU and a pointer to a buffer (SduDataPtr) containing the I-PDU.
Description	This is the receive indication callback function. It de-multiplex the incoming PDU and calls the corresponding upper layer receive indication callback function	

4.3.2.2.7. IpduM_Transmit

Purpose	Transmit an I-PDU.	
Synopsis	<pre>Std_ReturnType IpduM_Transmit (PduIdType Tx- PduId , const PduInfoType * PduInfoPtr);</pre>	
Service ID	0x03	
Sync/Async	synchronous	
Reentrancy	Non Reentrant for the same PDU-ID. Reentrant for different PDU-ID	
Parameters (in)	TxPduId	ID of I-PDU to be transmitted.
	PduInfoPtr	A pointer to a structure with I-PDU related data that shall be transmitted: data length and pointer to I-SDU buffer
Return Value	Standard Return Code	
	E_OK	The request was accepted by IpduM.
	E_NOT_OK	The request was not accepted by IpduM, a detailed error condition was sent to DET.
Description	This function transmits the data given through PduInfoPtr through the I-PDU given by TxPduId.	

4.3.2.2.8. IpduM_TriggerTransmit

Purpose	Copy data to PDU-router memory.	
Synopsis	<pre>Std_ReturnType IpduM_TriggerTransmit (PduId- Type TxPduId , PduInfoType * PduInfoPtr);</pre>	
Service ID	0x41	
Sync/Async	synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId	

Parameters (in)	TxPduId	ID of IpduM I-PDU that is requested to be transmitted by IpduM
	PduInfoPtr	Contains a pointer to a buffer (SduDataPtr) to where the SDU shall be copied to. On return, the service will indicate the length of the copied SDU data in SduLength.
Return Value	E_OK	SDU has been copied and SduLength indicates the number of copied bytes.
E_NOT_OK	No SDU has been copied. PduInfoPtr must not be used since it may contain a NULL pointer or point to invalid data.	
Description	The lower layer communication module requests the buffer of the SDU for transmission from the upper layer module.	

4.3.2.2.9. IpduM_TxConfirmation

Purpose	Transmit confirmation callback function.	
Synopsis	<pre>void IpduM_TxConfirmation (PduIdType TxPduId);</pre>	
Service ID	0x40	
Sync/Async	synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId	
Parameters (in)	TxPduId	ID of multiplexed I-PDU that has been transmitted.
Description	This is the transmit confirmation callback function. It gets the PDU handle for the transmitted I-PDU, translates it for the upper layer and then calls the upper layer callback functions configured for this handle.	

4.3.3. Integration notes

4.3.3.1. Exclusive areas

This section describes the exclusive areas used by the IpduM module.

4.3.3.1.1. SCHM_IPDUM_EXCLUSIVE_AREA_0

Protected data structures	All shared data that shall be protected from mutual access.
Recommended locking mechanism	This exclusive area must always be protected by a locking mechanism. The options for locking are described in the EB tresos AutoCore Generic documentation . Refer to the section Mapping exclusive areas in the basic software modules in the Integration notes section for details.

4.3.3.2. Production errors

Production errors are not reported by the `IpduM` module.

4.3.3.3. Memory mapping

General information about memory mapping is provided in the [EB tresos AutoCore Generic documentation](#). Refer to the section [Memory mapping and compiler abstraction](#) in the [Integration notes](#) section for details.

The following table provides the list of sections that may be mapped for this module:

Memory section
CONST_32
VAR_NO_INIT_32
VAR_NO_INIT_16
VAR_NO_INIT_8
VAR_FAST_INIT_8
VAR_NO_INIT_BOOLEAN
CONST_UNSPECIFIED
CODE
VAR_FAST_INIT_UNSPECIFIED
VAR_INIT_UNSPECIFIED
VAR_NO_INIT_UNSPECIFIED
CONFIG_DATA_UNSPECIFIED

4.3.3.4. Integration requirements

WARNING Integration requirements list is not exhaustive



The following list of integration requirements helps you to integrate your product. However, this list is not exhaustive. You also require information from the user's guide, release notes, and EB tresos AutoCore known issues to successfully integrate your product.

Integration requirements are not listed for the IpduM module.

4.4. PduR

4.4.1. Configuration parameters

Containers included		
Container name	Multiplicity	Description
CommonPublishedInformation	1..1	Label: Common Published Information Common container, aggregated by all modules. It contains published information about vendor and versions.
PublishedInformation	1..1	Label: EB Published Information Additional published parameters not covered by CommonPublishedInformation container.
PduRBswModules	0..n	Each container describes a specific BSW module (upper/CDD/lower/IpduM) that the PDU Router shall interface to. The reason to have it as own configuration container instead of implication of the routing path is to be able to configure CDD:s properly and to force module's to be used in a post-build situation even though no routing is made to/from this module (future configurations may include these modules). <i>Note: The short name of the container PduRBswModules provides the Module Short Name MSN.</i>
PduRGeneral	1..1	This container is a subcontainer of PduR and specifies the general configuration parameters of the PDU Router.
PduRRoutingTables	1..n	Represents one table of routing paths. This routing table allows multiple configurations that can be used to create several routing tables in the same configuration. This is mainly used

Containers included		
		for post-build (e.g. post-build selectable) but can be used by pre-compile and link-time for variant handling.

Parameters included	
Parameter name	Multiplicity
IMPLEMENTATION_CONFIG_VARIANT	1..1

Parameter Name	IMPLEMENTATION_CONFIG_VARIANT
Label	Configuration Variant
Multiplicity	1..1
Type	ENUMERATION
Default value	VariantPostBuild
Range	VariantPostBuild

4.4.1.1. CommonPublishedInformation

Parameters included	
Parameter name	Multiplicity
ArMajorVersion	1..1
ArMinorVersion	1..1
ArPatchVersion	1..1
SwMajorVersion	1..1
SwMinorVersion	1..1
SwPatchVersion	1..1
ModuleId	1..1
VendorId	1..1
Release	1..1

Parameter Name	ArMajorVersion
Label	AUTOSAR Major Version
Description	Major version number of AUTOSAR specification on which the appropriate implementation is based on.

Multiplicity	1..1
Type	INTEGER_LABEL
Default value	3
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	ArMinorVersion
Label	AUTOSAR Minor Version
Description	Minor version number of AUTOSAR specification on which the appropriate implementation is based on.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	2
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	ArPatchVersion
Label	AUTOSAR Patch Version
Description	Patch level version number of AUTOSAR specification on which the appropriate implementation is based on.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	0
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	SwMajorVersion
Label	Software Major Version
Description	Major version number of the vendor specific implementation of the module.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	5
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	SwMinorVersion
Label	Software Minor Version
Description	Minor version number of the vendor specific implementation of the module. The numbering is vendor specific.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	3
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	SwPatchVersion
Label	Software Patch Version
Description	Patch level version number of the vendor specific implementation of the module. The numbering is vendor specific.
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	30
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	ModuleId
Label	Numeric Module ID
Description	Module ID of this module from Module List
Multiplicity	1..1
Type	INTEGER_LABEL
Default value	51
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	VendorId
Label	Vendor ID
Description	Vendor ID of the dedicated implementation of this module according to the AUTOSAR vendor list
Multiplicity	1..1

Type	INTEGER_LABEL
Default value	1
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

Parameter Name	Release
Label	Release Information
Multiplicity	1..1
Type	STRING_LABEL
Default value	
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

4.4.1.2. PublishedInformation

Parameters included	
Parameter name	Multiplicity
PbcfgMSupport	1..1

Parameter Name	PbcfgMSupport
Label	PbcfgM support
Description	Specifies whether or not the PduR can use the PbcfgM module for post-build support.
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	PublishedInformation:
Origin	Elektrobit Automotive GmbH

4.4.1.3. PduRBswModules

Parameters included	
Parameter name	Multiplicity
PduRCancelReceive	1..1

Parameters included	
PduRCancelTransmit	1..1
PduRChangeParameterRequestApi	1..1
PduRCommunicationInterface	1..1
PduRLowerModule	1..1
PduRRetransmission	1..1
PduRTransportProtocol	1..1
PduRTriggertransmit	1..1
PduRTxConfirmation	1..1
PduRUpperModule	1..1
PduRUseTag	1..1
PduRBswModuleRef	1..1
PduRBswModuleIsEnabled	1..1
PduRStaticPduLengthSupport	1..1
PduRBswModuleApiDefinition	1..1
PduRCalculateHandleId	1..1
PduRMaxRxPduId	1..1
PduRMaxTxPduId	1..1

Parameter Name	PduRCancelReceive
Description	<p>Specifies if the Transport protocol module supports the CancelReceive API or not.</p> <ul style="list-style-type: none"> ▶ TRUE: Cancel Receive Functionality is enabled (switched on). ▶ FALSE: Cancel Receive Functionality is disabled (switched off). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Disabling all of the parameters <code><parameter>PduRCancelReceive</parameter></code> reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.

Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	PreCompile: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	PduRCancelTransmit
Description	<p>Specifies if the BSW module supports the CancelTransmit API or not.</p> <ul style="list-style-type: none"> ▶ TRUE: Cancel transmit functionality is enabled (switched on). ▶ FALSE: Cancel transmit functionality is disabled (switched off). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Disabling all of the parameters <code><parameter>PduRCancelTransmit</parameter></code> reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	PreCompile: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	PduRChangeParameterRequestApi
Description	<p>Specifies if the BSW module supports the ChangeParameter API or not.</p> <ul style="list-style-type: none"> ▶ TRUE: Change parameter functionality is enabled (switched on). ▶ FALSE: Change parameter functionality is disabled (switched off). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Disabling all of the parameters <code><parameter>PduRChangeParameterRequestApi</parameter></code> reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.
Multiplicity	1..1

Type	BOOLEAN
Default value	false
Configuration class	PreCompile: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	PduRCommunicationInterface
Description	Specifies if the BSW module supports the Communication Interface APIs or not. Value true the APIs are supported. A module can have both Communication Interface APIs and Transport Protocol APIs (e.g. the COM module).
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	PduRLowerModule
Description	The PduRLowerModule will decide who will call the APIs and who will implement the APIs. For example, if the CanIf module is referenced then the PDU Router module will implement the PduR_CanIfRxIndication API. And the PDUR module will call the CanIf_Transmit API. Other APIs are of course also covered. An upper module can also be a lower module (e.g. the IpduM module).
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	PduRRetransmission
Description	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>If set to true this means that the destination transport protocol module will use the retransmission feature. This parameter might be set to false if the retransmission feature is not used, even though the destination transport protocol is supporting it. This parameter is only valid for transport protocol modules and gateway operations. If transmission from a local upper layer module this module will handle the retransmission.</p>

Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	PduRTransportProtocol
Description	The PDU Router module shall use the API parameters specified for transport protocol interface.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	PduRTriggertransmit
Description	Specifies if the BSW module supports the TriggerTransmit API or not. Value true means that the BSW module supports the TriggerTransmit interface which a lower layer module can call and also that it can call the TriggerTransmit interface of an upper layer module. Value false means that the BSW module does not support the TriggerTransmit interface which a lower layer module can call and also that it shall not call the TriggerTransmit interface of an upper layer module.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	PreCompile: VariantPostBuild
Origin	AUTOSAR_ECUC

Parameter Name	PduRTxConfirmation
Description	Specifies if the BSW module supports the TxConfirmation API or not. Value true the API is supported.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	PreCompile: VariantPostBuild

Origin	AUTOSAR_ECUC
--------	--------------

Parameter Name	PduRUpperModule	
Description	The PduRUpperModule will decide who will call the APIs and who will implement the APIs. For example, if the COM module is referenced then the PDU Router module will implement the PduR_Transmit API. And the PDUR module will call the Com_RxIndication API. Other APIs are of course also covered. An upper module can also be an lower module (e.g. the IpduM module).	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRUseTag	
Description	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>This parameter, if set to true, enables the usage of the tag (<up>) in the following API calls: * PduR_<Up>CancelReceive * PduR_<Up>CancelTransmit * PduR_<Up>ChangeParameter Example: If used by COM and the parameter is enabled the PduR_ComCancelTransmit is used. The background is that upper layer modules differ in usage of this tag (e.g. COM is using the tag, DCM is not).</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRBswModuleRef	
Description	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>To identify the adjacent module by the properties, the PduRBswModules container name shall be equal to the module name. This is a reference to one BSW module's configuration (i.e. not the ECUC parameter definition template). Example, there could be several configurations of LinIf and this reference selects one of them.</p>	
Multiplicity	1..1	
Type	FOREIGN-REFERENCE	

Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRBswModuleIsEnabled	
Description	<p>Specifies if the Bsw Module is available.</p> <ul style="list-style-type: none"> ▶ TRUE: The Bsw Module is available (switched on). ▶ FALSE: The Bsw Module is not available (switched off). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Disabling the parameter reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild

Parameter Name	PduRStaticPduLengthSupport	
Description	<p>Solely I-PDUs with fixed payload are gatewayed by the module.</p> <ul style="list-style-type: none"> ▶ TRUE: Only static communication interface I-PDUs are received for gatewaying (switched on). ▶ FALSE: Also communication interface I-PDUs variable in length might be received for gatewaying (switched off). <p><i>Note: The lower layer must always provide a buffer which is not smaller than the length specified in EcuC. Rationale: If the actual length of the receive buffer provided by the lower layer is smaller than the length used by the upper layer, the upper layer will read more bytes than available in the provided buffer.</i></p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Enabling this parameter reduces the ROM consumption of the module configuration. ▶ RAM reduction (config): Enabling this parameter reduces the RAM consumption of the module configuration. 	
Multiplicity	1..1	

Type	BOOLEAN
Default value	false
Configuration class	PreCompile: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRBswModuleApiDefinition
Description	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>Selects the way the APIs of the module are to be defined:</p> <ul style="list-style-type: none"> ▶ FUNCTION: Module APIs are defined as functions. ▶ MACRO: Module APIs are defined as macros.
Multiplicity	1..1
Type	ENUMERATION
Default value	FUNCTION
Range	FUNCTION MACRO
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRCalculateHandleId
Description	<p>Specifies if the Handle IDs shall be calculated.</p> <ul style="list-style-type: none"> ▶ TRUE: The Handle IDs are calculated (switched on). ▶ FALSE: The Handle IDs are not calculated (switched off).
Multiplicity	1..1
Type	BOOLEAN
Default value	true
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRMaxRxPduld
Description	<p>Specifies the maximum RxPduld that might be provided by the AUTOSAR 3.2 upper layer module.</p> <p><i>Note: The RxPdulds of the upper layer module with TP interface shall be zero-based and dense.</i></p>

	Optimization Effect: ► RAM reduction (config): Decreasing this parameter reduces the RAM consumption of the module configuration.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	>=0	
	<=255	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	PduRMaxTxPduld	
Description	Specifies the maximum TxPduld that might be provided by the AUTOSAR 3.2 upper layer module. <i>Note: The TxPdulds of the upper layer module with TP interface shall be zero-based and dense.</i> Optimization Effect: ► RAM reduction (config): Decreasing this parameter reduces the RAM consumption of the module configuration.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	>=0	
	<=255	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

4.4.1.4. PduRGeneral

Parameters included	
Parameter name	Multiplicity
PduRDevErrorDetect	1..1
PduRVersionInfoApi	1..1

Parameters included	
PduRZeroCostOperation	1..1
PduRASR32RevisionCompatibility	1..1
PduRMultiTpTxRPathsMax	1..1
PduRIfGatewaySupport	1..1
PduRTpGatewaySupport	1..1
PduRTpGwQueueEnable	1..1
PduRRelocatableCfgEnable	1..1
PduRSbTxBufferSupport	1..1
PduRFifoTxBufferSupport	1..1
PduRNto1RoutingSupport	1..1
PduRMemorySize	0..1
PduRMemorySizeExtension	1..1
PduRMulticastTxConfirmation	1..1
PduRMulticastFromIfSupport	1..1
PduRMulticastToIfSupport	1..1
PduRMulticastLoTpToUpSupport	1..1
PduRMulticastUpToLoTpSupport	1..1

Parameter Name	PduRDevErrorDetect
Label	Enable Development Error Detection
Description	<p>Enables the error-reporting to the Development Error Tracer (DET).</p> <p>► TRUE: Development Error Detection mechanism is enabled (switched on).</p>

	<p>► FALSE: Development Error Detection mechanism is disabled (switched off).</p> <p>Optimization Effect:</p> <p>► ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.</p> <p>► Execution time reduction (code): Disabling this parameter reduces the execution time of the module code.</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	true	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRVersionInfoApi	
Label	Enable Version Info API	
Description	<p>Enables the PduR_GetVersionInfo API.</p> <p>► TRUE: PduR_GetVersionInfo API is available (switched on).</p> <p>► FALSE: PduR_GetVersionInfo API is available (switched off).</p> <p>Optimization Effect:</p> <p>► ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRZeroCostOperation	
Description	<p>If set the PduR configuration generator will report an error if zero-cost-operation cannot be fulfilled. This parameter shall be seen as an input requirement to the configuration generator.</p> <p>Optimization Effect:</p> <p>► ROM reduction (config): Enabling this parameter reduces the ROM consumption of the module configuration.</p>	

	<ul style="list-style-type: none"> ▶ ROM reduction (code): Enabling this parameter reduces the ROM consumption of the module code. ▶ Execution time reduction (code): Enabling this parameter reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRASR32RevisionCompatibility	
Description	<p>Enables/Disables the passing of return values of AUTOSAR 3.2 upper layer modules for dedicated AUTOSAR 3.2 revisions on Rx side.</p> <ul style="list-style-type: none"> ▶ TRUE: Return values are passed according to revision 1 and 2 (BUFREQ_E_BUSY is available). ▶ FALSE: Return values are passed according to revision 3 (BUFREQ_E_BUSY is NOT available, i.e. mapped to BUFREQ_E_OVFL). 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	PduRMultiTpTxRPathsMax	
Description	<p>Maximum number of simultaneously handled routing paths that route an I-PDU from an upper layer module to multiple lower layer TP modules.</p> <p><i>Note 1: If value 0 is chosen, the value is internally set to the number of routing paths configured that support routing of an I-PDU from an upper layer module to multiple lower layer TP modules.</i></p> <p><i>Note 2: If value 0 is chosen, the value is internally set at least to one.</i></p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ RAM reduction (config): Decreasing this parameter reduces the RAM consumption of the module configuration. 	
Multiplicity	1..1	
Type	INTEGER	

Default value	0
Range	>=0 <=255
Configuration class	PreCompile: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRIfGatewaySupport
Description	<p>Configuration parameter to enable or disable the PDU Router gateway operation between lower layer Interface modules.</p> <ul style="list-style-type: none"> ▶ TRUE: non-TP PDU gateway routing is enabled (switched on). ▶ FALSE: non-TP PDU gateway routing is disabled (switched off). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRTpGatewaySupport
Description	<p>Configuration parameter to enable or disable the PDU Router gateway operation from a lower layer transport protocol module to one or more lower layer transport protocol module(s).</p> <ul style="list-style-type: none"> ▶ TRUE: TP PDU gateway routing is enabled (switched on). ▶ FALSE: TP PDU gateway routing is disabled (switched off). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.

Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRTpGwQueueEnable
Description	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>Configuration parameter to enable or disable the queuing of incoming TP gateway requests if previous TP gateway sessions with the same N-SDU ID are currently ongoing. The queued requests shall be processed once the previous session has completed.</p> <p><i>Note: Be careful when configuring the TP buffers to be used. Enabling this feature causes the usage of more TP buffers for the same PDU-Id than without.</i></p> <ul style="list-style-type: none"> ▶ TRUE: Queuing of incoming TP PDUs is enabled (switched on). ▶ FALSE: Queuing of incoming TP PDUs is disabled (switched off). <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ RAM reduction (config): Disabling this parameter reduces the RAM consumption of the module configuration.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRRelocatableCfgEnable
Description	<p>Enables or disables the post-build-time configuration data to be used either by relative offsets to the configuration start address (relocatable) or by absolute pointers (not relocatable).</p> <ul style="list-style-type: none"> ▶ TRUE: Relocateable configuration is in use (switched on). ▶ FALSE: Relocateable configuration is not in use (switched off). <p>Note: If PbcfgMBswModuleRef contains a reference to this module, then this feature is managed by the parameter PbcfgMRelocatableCfgEnable of the PbcfgM.</p>

	Optimization Effect: <ul style="list-style-type: none"> ▶ ROM reduction (config): Enabling this parameter reduces the ROM consumption of the module configuration. ▶ Execution time reduction (code): Disabling this parameter reduces the execution time of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	PduRSbTxBufferSupport	
Description	<p>Configuration parameter to enable or disable PDU Router support for single buffers used by non-TP-PDU gateway operations.</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	PduRFifoTxBufferSupport	
Description	<p>Configuration parameter to enable or disable PDU Router support for FIFO buffers used by non-TP-PDU gateway operations.</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ▶ ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. ▶ ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. 	
Multiplicity	1..1	

Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRNto1RoutingSupport
Description	<p>Configuration parameter to enable or disable the N:1 routing (non-Autosar-feature).</p> <ul style="list-style-type: none"> ▶ TRUE: N:1 PDU routing is enabled (switched on). ▶ FALSE: N:1 PDU routing is disabled (switched off). <p>Note: The destination PDUs have to be configured completely identical!</p>
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRMemorySize
Description	Memory size reserved for the PDU Router. Only required for gateway operation. If the parameter is disabled, then the size is calculated for the current configuration by the module configuration generator.
Multiplicity	0..1
Type	INTEGER
Default value	0
Configuration class	PreCompile: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRMemorySizeExtension
Description	<p>Configuration parameter to allow RAM size usage of more than 64 KiB.</p> <ul style="list-style-type: none"> ▶ TRUE: RAM size NOT limited by 64 KiB at maximum (switched on). ▶ FALSE: RAM size limited by 64 KiB at maximum (switched off). <p>Optimization Effect:</p>

	► ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	PduRMulticastTxConfirmation	
Description	<p>Configuration parameter to enable multicast transmission confirmation. This allows to call Up_TxConfirmation with the last PduR_LoTxConfirmation received for a transmission to multiple communication interface modules. Receiving PduR_LoTxConfirmation for dedicated destination PDUs is enabled by parameter PduR-TransmissionConfirmation.</p> <p>► TRUE: Allows calling Up_TxConfirmation for multicast transmission.</p> <p>► FALSE: Disallows calling Up_TxConfirmation for multicast transmission.</p> <p>Optimization Effect:</p> <p>► ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration.</p> <p>► RAM reduction (config): Disabling this parameter reduces the RAM consumption of the module configuration.</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	PreCompile:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	PduRMulticastFromIfSupport	
Description	<p>Configuration parameter to enable or disable PDU Router support for multicast from a lower layer interface module to upper layer modules or lower layer interface modules.</p> <p>Optimization Effect:</p> <p>► ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration.</p>	

	► ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	PduRMulticastTolFSupport	
Description	<p>Configuration parameter to enable or disable PDU Router support for multicast from an upper layer module to lower layer interface modules.</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ► ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. ► ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. 	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

Parameter Name	PduRMulticastLoTpToUpSupport	
Description	<p>Configuration parameter to enable or disable PDU Router support for multicast from a lower layer TP module to an upper layer module and one or more lower layer TP modules. <i>Note: Enabling this switch is only necessary if an upper layer transport protocol module is involved. Otherwise enabling configuration parameter PduRTpGatewaySupport is sufficient.</i></p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ► ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. ► ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code. 	
Multiplicity	1..1	

Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

Parameter Name	PduRMulticastUpToLoTpSupport
Description	<p>Configuration parameter to enable or disable PDU Router support for multicast from an upper layer module to lower layer TP modules.</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ► ROM reduction (config): Disabling this parameter reduces the ROM consumption of the module configuration. ► ROM reduction (code): Disabling this parameter reduces the ROM consumption of the module code.
Multiplicity	1..1
Type	BOOLEAN
Default value	false
Configuration class	VariantPostBuild: VariantPostBuild
Origin	Elektrobit Automotive GmbH

4.4.1.5. PduRRoutingTables

Containers included		
Container name	Multiplicity	Description
PduRRoutingPathGroup	0..n	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>This container groups routing path destinations. Destinations are used instead of routing paths since a routing path can be 1:n. It is desirable to be able to enable/disable a specific bus (i.e. a destination) rather than a routing path. Of course it is possible to create groups that covers specific routing paths as well.</p> <p><i>Note: Enabling and disabling of routing path groups are made using the PduR API.</i></p>
PduRRoutingTable	0..n	Represents one container of routing paths. Each container is either minimum routing or not.

Containers included		
PduRTpBufferTable	0..1	This container will specify the needed buffers for gatewaying using TP. It is not connected to the specific routing path destination to allow a more efficient buffer handling.
PduRTxBufferTable	0..1	This container will specify the needed buffers for gatewaying using communication interface. It is not defined per routing path to allow reuse of buffers.

Parameters included	
Parameter name	Multiplicity
PduRConfigurationId	1..1

Parameter Name	PduRConfigurationId	
Description	Identification of the configuration of the PduR configuration. This identification can be read using the PduR API. <i>Note: The value 65535 is used as an invalid configuration ID.</i>	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<div><=65534</div> <div>>=0</div>	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.4.1.6. PduRRoutingPathGroup

Parameters included	
Parameter name	Multiplicity
PduRIsEnabledAtInit	1..1
PduRRoutingPath-GroupId	1..1
PduRDestPduRef	1..n

Parameter Name	PduRIsEnabledAtInit
----------------	---------------------

Description	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>If set to true this routing path group will be enabled after initializing the PDU Router module (i.e. enabled in the PduR_Init function).</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRRoutingPathGroupId	
Description	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>Identification of the routing group.</p> <p><i>Note: The identification will be used by the disable/enable API in the PDU Router module API.</i></p>	
Multiplicity	1..1	
Type	INTEGER	
Range	<=65535	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRDestPduRef	
Description	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>This reference selects one destination of the routing path.</p>	
Multiplicity	1..n	
Type	REFERENCE	
Configuration class	PostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.4.1.7. PduRRoutingTable

Containers included		
Container name	Multiplicity	Description

Containers included		
PduRRoutingPath	0..n	<p>This container specifies the routing path of a PDU.</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ► ROM reduction (config): Removing a Routing Path reduces the ROM consumption of the module configuration.

Parameters included	
Parameter name	Multiplicity
PduRIsMinimumRouting	1..1

Parameter Name	PduRIsMinimumRouting	
Description	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>Specifies if the container contains routing paths that are of the type minimum routing or not.</p>	
Multiplicity	1..1	
Type	BOOLEAN	
Default value	false	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.4.1.8. PduRRoutingPath

Containers included		
Container name	Multiplicity	Description
PduRDestPdu	1..n	This container is a subcontainer of PduRRoutingPath and specifies one destination for the PDU to be routed.
PduRSrcPdu	1..1	This container is a subcontainer of PduRRoutingPath and specifies the source of the PDU to be routed.

Parameters included	
Parameter name	Multiplicity
PduRTpGwQueueDepth	0..1

Parameter Name	PduRTpGwQueueDepth
----------------	--------------------

Description	<p><i>The functionality related to this parameter is not supported by the current implementation.</i></p> <p>Maximum number of TP buffers that can be queued for routing this path via the TP gateway including the TP buffer in use.</p> <p>This limitation avoids the occupation of all TP buffers by a single routing path.</p>	
Multiplicity	0..1	
Type	INTEGER	
Default value	1	
Range	>=1	
	<=255	
Configuration class	PostBuild:	VariantPostBuild
Origin	Elektrobit Automotive GmbH	

4.4.1.9. PduRDestPdu

Containers included		
Container name	Multiplicity	Description
PduRDefaultValue	0..1	Specifies the default value of the I-PDU. Only required for gateway operation and if at least one PDU specified by PduR-DestPdu uses TriggerTransmit Data provision. Represented as an array of IntegerParamDef.

Parameters included	
Parameter name	Multiplicity
PduRDestPduDataProvision	0..1
PduRDestPduHandleId	0..1
PduRTpThreshold	0..1
PduRTransmissionConfirmation	0..1
PduRDestPduRef	1..1
PduRDestTxBufferRef	0..1

Parameter Name	PduRDestPduDataProvision
-----------------------	---------------------------------

Description	<p>Specifies how data are provided:</p> <ul style="list-style-type: none"> ▶ PDUR_DIRECT: direct (as part of the Transmit call) ▶ PDUR_TRIGGERTRANSMIT: via the TriggerTransmit callback function <p>Only required for non-TP gatewayed I-PDUs.</p>	
Multiplicity	0..1	
Type	ENUMERATION	
Default value	PDUR_DIRECT	
Range	PDUR_DIRECT	
	PDUR_TRIGGERTRANSMIT	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRDestPduHandleId	
Description	PDU identifier assigned by PDU Router. Used by communication interface and transport protocol modules for confirmation.	
Multiplicity	0..1	
Type	INTEGER	
Range	<=65535	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRTpThreshold	
Description	<p>Defines the number of bytes which shall be received before transmission on the destination bus may start. Only required for routing-on-the-fly TP gateway PDUs. The threshold shall not be larger than the length of the related TP buffer.</p>	
Multiplicity	0..1	
Type	INTEGER	
Default value	1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRTransmissionConfirmation	
-----------------------	-------------------------------------	--

Description	This parameter is only for communication interfaces. Transport protocol modules will always call the TxConfirmation function. If set the destination communication interface module will call the TxConfirmation. However the TxConfirmation may be not called due to error. So the PduR shall not block until the TxConfirmation is called. One background for this parameter is for the PduR to know when all modules have confirmed a multicast operation. The support of this functionality is generally switched on/off by parameter PduRMulticastTxConfirmation.	
Multiplicity	0..1	
Type	BOOLEAN	
Default value	FALSE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRDestPduRef	
Description	Destination PDU reference; reference to unique PDU identifier which shall be used by the PDU Router instead of the source PDU ID when calling the related function of the destination module.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRDestTxBufferRef	
Description	Reference to a buffer that is allocated in the PduRTxBuffer. Having a global (for PduR) list of buffers allows reuse and hence less memory consumption.	
Multiplicity	0..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.4.1.10. PduRDefaultValue

Containers included		
Container name	Multiplicity	Description

Containers included		
PduRDefaultValueElement	0..n	Each value element is represented by the element and the position in an array.

4.4.1.11. PduRDefaultValueElement

Parameters included	
Parameter name	Multiplicity
PduRDefaultValueElement	1..1
PduRDefaultValueElementBytePosition	1..1

Parameter Name	PduRDefaultValueElement	
Description	The default value consists of a number of elements. Each element is one byte long and the number of elements is specified by PduLength. The position of this parameter in the container is specified by the PduRElementBytePosition parameter.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=255	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRDefaultValueElementBytePosition	
Description	This parameter specifies the byte position of the element within the default value.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=255	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild

Origin	AUTOSAR_ECUC
--------	--------------

4.4.1.12. PduRSrcPdu

Parameters included	
Parameter name	Multiplicity
PduRSourcePduHandleId	1..1
PduRSrcPduRef	1..1

Parameter Name	PduRSourcePduHandleId	
Description	PDU identifier assigned by PDU Router.	
Multiplicity	1..1	
Type	INTEGER	
Range	<=65535	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRSrcPduRef	
Description	Source PDU reference; reference to unique PDU identifier which shall be used for the requested PDU Router operation.	
Multiplicity	1..1	
Type	REFERENCE	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.4.1.13. PduRTpBufferTable

Containers included		
Container name	Multiplicity	Description
PduRTpBuffer	0..n	This container specifies a buffer for a TP gateway operation. <i>Note: A circular TP buffer implementation is applied for routing on the fly.</i>

Containers included		
		Optimization Effect: <ul style="list-style-type: none"> ► RAM reduction (config): Removing a TP buffer reduces the RAM consumption of the module configuration.
Parameters included		
Parameter name	Multiplicity	
PduRMaxTpBufferNumber	1..1	
Parameter Name	PduRMaxTpBufferNumber	
Description	<i>The functionality related to this parameter is not supported by the current implementation.</i> Maximum number of TP buffers used for TP gateway operation.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=65535	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.4.1.14. PduRTpBuffer

Parameters included		
Parameter name	Multiplicity	
PduRTpBufferLength	1..1	
Parameter Name	PduRTpBufferLength	
Description	Length of the TP buffer in number of bytes.	
Multiplicity	1..1	
Type	INTEGER	
Default value	8	
Range	<=65535	
	>=1	

Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.4.1.15. PduRTxBufferTable

Containers included		
Container name	Multiplicity	Description
PduRTxBuffer	0..n	<p>This container specifies a Transmit Buffer for a non-TP PDU.</p> <p>Optimization Effect:</p> <ul style="list-style-type: none"> ► RAM reduction (config): Removing a Transmit Buffer reduces the RAM consumption of the module configuration.

Parameters included	
Parameter name	Multiplicity
PduRMaxTxBufferNumber	1..1

Parameter Name	PduRMaxTxBufferNumber	
Description	<i>The functionality related to this parameter is not supported by the current implementation.</i>	
	Maximum number of transmit buffers used for non-TP gateway operations.	
Multiplicity	1..1	
Type	INTEGER	
Default value	0	
Range	<=65535	
	>=0	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.4.1.16. PduRTxBuffer

Parameters included	
Parameter name	Multiplicity

Parameters included	
PduRPduMaxLength	1..1
PduRTxBufferDepth	1..1

Parameter Name	PduRPduMaxLength	
Description	Length of the Tx buffer in number of bytes.	
Multiplicity	1..1	
Type	INTEGER	
Default value	8	
Range	<=255	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

Parameter Name	PduRTxBufferDepth	
Description	Number of Pdus that can be stored in the buffer. If value is 1 then the buffer semantic is "last is best". If the value is greater than 1 then the buffer semantic is a FiFo.	
Multiplicity	1..1	
Type	INTEGER	
Default value	1	
Range	<=255	
	>=1	
Configuration class	VariantPostBuild:	VariantPostBuild
Origin	AUTOSAR_ECUC	

4.4.2. Recommended configurations

4.4.2.1. PduRRecConfigurationGatewayEcu

Containers included	
Container name	Container definition

Containers included	
Com	PduRBswModules
Dcm	PduRBswModules
CanIf	PduRBswModules
LinIf	PduRBswModules
Frlf	PduRBswModules
CanTp	PduRBswModules
LinTp	PduRBswModules
FrTp	PduRBswModules
SoAd	PduRBswModules
DoIP	PduRBswModules
IpduM	PduRBswModules
PduRGeneral	PduRGeneral

Parameters included	
Parameter name	Value

4.4.2.1.1. Com

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	false
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true

Parameters included	
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.2. Dcm

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	false
PduRRetransmission	true
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.3. CanIf

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false

Parameters included	
PduRTransportProtocol	false
PduRTriggertransmit	false
PduRTxConfirmation	true
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.4. LinIf

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.5. FrIf

Parameters included	
Parameter name	Value

Parameters included	
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.6. CanTp

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	true
PduRCommunicationInterface	false
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	false
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false

Parameters included	
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.7. LinTp

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	true
PduRCommunicationInterface	false
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	false
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.8. FrTp

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	true
PduRCommunicationInterface	false
PduRLowerModule	true
PduRRetransmission	true

Parameters included	
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	false
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.9. SoAd

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	true
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.10. DoIP

Parameters included	
Parameter name	Value

Parameters included	
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	false
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.11. Ipdum

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false

Parameters included	
PduRBswModuleApiDefinition	FUNCTION

4.4.2.1.12. PduRGeneral

Parameters included	
Parameter name	Value
PduRIfGatewaySupport	true
PduRTpGatewaySupport	true
PduRSbTxBufferSupport	true
PduRFifoTxBufferSupport	true
PduRMulticastFromIfSupport	true
PduRMulticastToIfSupport	true
PduRMulticastLoTpToUpSupport	true
PduRMulticastUpToLoTpSupport	true

4.4.2.2. PduRRecConfigurationFrEcu

Containers included	
Container name	Container definition
Com	PduRBswModules
Dcm	PduRBswModules
Frlf	PduRBswModules
FrTp	PduRBswModules
IpduM	PduRBswModules

Parameters included	
Parameter name	Value

4.4.2.2.1. Com

Parameters included	
Parameter name	Value

Parameters included	
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	false
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.2.2. Dcm

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	false
PduRRetransmission	true
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false

Parameters included	
PduRBswModuleApiDefinition	FUNCTION

4.4.2.2.3. FrIf

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.2.4. FrTp

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	true
PduRCommunicationInterface	false
PduRLowerModule	true
PduRRetransmission	true
PduRTransportProtocol	true

Parameters included	
PduRTriggertransmit	false
PduRTxConfirmation	false
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.2.5. IpduM

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.3. PduRRecConfigurationCanEcu

Containers included	
Container name	Container definition

Containers included	
Com	PduRBswModules
Dcm	PduRBswModules
CanIf	PduRBswModules
CanTp	PduRBswModules
IpduM	PduRBswModules

Parameters included	
Parameter name	Value

4.4.2.3.1. Com

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	false
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUppperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.3.2. Dcm

Parameters included	
Parameter name	Value

Parameters included	
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	false
PduRRetransmission	true
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.3.3. CanIf

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	false
PduRTxConfirmation	true
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false

Parameters included	
PduRBswModuleApiDefinition	FUNCTION

4.4.2.3.4. CanTp

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	true
PduRCommunicationInterface	false
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	false
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.3.5. IpduM

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	false

Parameters included	
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.4. PduRRecConfigurationLinEcu

Containers included	
Container name	Container definition
Com	PduRBswModules
Dcm	PduRBswModules
LinIf	PduRBswModules
LinTp	PduRBswModules
IpduM	PduRBswModules

Parameters included	
Parameter name	Value

4.4.2.4.1. Com

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	false
PduRRetransmission	false

Parameters included	
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.4.2. Dcm

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	false
PduRRetransmission	true
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.4.3. LinIf

Parameters included	
Parameter name	Value

Parameters included	
PduRCancelReceive	false
PduRCancelTransmit	true
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.4.4. LinTp

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	true
PduRCommunicationInterface	false
PduRLowerModule	true
PduRRetransmission	true
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	false
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false

Parameters included	
PduRBswModuleApiDefinition	FUNCTION

4.4.2.4.5. IpduM

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.5. PduRRecConfigurationEthernetEcu

Containers included	
Container name	Container definition
Com	PduRBswModules
Dcm	PduRBswModules
SoAd	PduRBswModules
DoIP	PduRBswModules
IpduM	PduRBswModules

Parameters included	
Parameter name	Value

4.4.2.5.1. Com

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	false
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.5.2. Dcm

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	true
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	false
PduRRetransmission	true
PduRTransportProtocol	true

Parameters included	
PduRTriggertransmit	false
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.5.3. SoAd

Parameters included	
Parameter name	Value
PduRCancelReceive	true
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	true
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.5.4. DoIP

Parameters included	
Parameter name	Value
PduRCancelReceive	false

Parameters included	
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	true
PduRTriggertransmit	false
PduRTxConfirmation	false
PduRUpperModule	false
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.2.5.5. Ipdum

Parameters included	
Parameter name	Value
PduRCancelReceive	false
PduRCancelTransmit	false
PduRChangeParameterRequestApi	false
PduRCommunicationInterface	true
PduRLowerModule	true
PduRRetransmission	false
PduRTransportProtocol	false
PduRTriggertransmit	true
PduRTxConfirmation	true
PduRUpperModule	true
PduRUseTag	true
PduRBswModuleIsEnabled	true
PduRStaticPduLengthSupport	false
PduRBswModuleApiDefinition	FUNCTION

4.4.3. Application programming interface (API)

4.4.3.1. Macro constants

4.4.3.1.1. PDUR_E_CONFIG_PTR_INVALID

Purpose	Error code for invalid configuration pointer.
Value	0x00U

4.4.3.1.2. PDUR_E_INVALID_REQUEST

Purpose	Error code if API service used without module initialization or PduR_Init called in any state other than PDUR_UNINIT.
Value	0x01U

4.4.3.1.3. PDUR_E_NULL_POINTER

Purpose	Pointer parameter is null. Note that specific API calls may disable this error.
Value	0x09U

4.4.3.1.4. PDUR_E_PDU_ID_INVALID

Purpose	Error code if invalid PDU identifier has been passed to a public API function.
Value	0x02U

4.4.3.1.5. PDUR_E_PDU_INSTANCES_LOST

Purpose	Loss of a PDU instance (FIFO flushed because of an overrun).
Value	0x0AU

4.4.3.1.6. PDUR_E_TP_TX_REQ_REJECTED

Purpose	Error code if TP module rejects a transmit request for a valid PDU identifier.
Value	0x03U

4.4.3.1.7. PDUR_INSTANCE_ID

Purpose	Id of instance of PDU Router provided to Det_ReportError().
Value	0x00U

4.4.3.1.8. PDUR_INVALID_CONFIGURATION_ID

Purpose	Macro definition for the invalid configuration Id returned by PduR_GetConfigurationId if the PduR is not initialized.
Value	0xFFFFFFFFU

4.4.3.1.9. PDUR_SID_GET_CONF_ID

Purpose	Definition of service ID for PduR_GetConfigurationId.
Value	0x10U

4.4.3.1.10. PDUR_SID_GET_VER_INF

Purpose	Definition of service ID for PduR_GetVersionInfo.
Value	0x02U

4.4.3.1.11. PDUR_SID_IFGW_RXIND_DF

Purpose	Definition of service ID for internal handler function PduR_GatelfRxIndicationDf(DynPyId).
Value	0x55U

4.4.3.1.12. PDUR_SID_IFGW_RXIND_SB

Purpose	Definition of service ID for internal handler function PduR_ GatelfRxIndicationSb(DynPyId).
Value	0x57U

4.4.3.1.13. PDUR_SID_IFGW_RXIND_TF

Purpose	Definition of service ID for internal handler function PduR_ GatelfRxIndicationTf(DynPyId).
Value	0x56U

4.4.3.1.14. PDUR_SID_IFGW_TRIGTX_SB

Purpose	Definition of service ID for internal handler function PduR_ GatelfTriggerTransmitSb(DynPyId).
Value	0x59U

4.4.3.1.15. PDUR_SID_IFGW_TRIGTX_TF

Purpose	Definition of service ID for internal handler function PduR_ GatelfTriggerTransmitTf(DynPyId).
Value	0x58U

4.4.3.1.16. PDUR_SID_INIT

Purpose	Definition of service ID for PduR_Init.
Value	0x01U

4.4.3.1.17. PDUR_SID_LOTP_COPY_RX_DATA

Purpose	Definition of service ID for PduR_LoTpCopyRxData.
----------------	---

Value	0x32U
--------------	-------

4.4.3.1.18. PDUR_SID_LOTP_COPY_TX_DATA

Purpose	Definition of service ID for PduR_LoTpCopyTxData.
Value	0x36U

4.4.3.1.19. PDUR_SID_LOTP_RXIND

Purpose	Definition of service ID for PduR_LoTpRxIndication.
Value	0x33U

4.4.3.1.20. PDUR_SID_LOTP_STRT_OF_RCPTN

Purpose	Definition of service ID for PduR_LoTpStartOfReception.
Value	0x34U

4.4.3.1.21. PDUR_SID_LOTP_TX_CONF

Purpose	Definition of service ID for PduR_LoTpTxConfirmation.
Value	0x37U

4.4.3.1.22. PDUR_SID_LO_RXIND

Purpose	Definition of service ID for PduR_LoRxIndication.
Value	0x42U

4.4.3.1.23. PDUR_SID_LO_TRIGTX

Purpose	Definition of service ID for PduR_LoTriggerTransmit.
----------------	--

Value	0x41U
--------------	-------

4.4.3.1.24. PDUR_SID_LO_TXCONF

Purpose	Definition of service ID for PduR_LoTxConfirmation.
Value	0x40U

4.4.3.1.25. PDUR_SID_UP_CANCELRXREQ

Purpose	Definition of service ID for PduR_UpCancelReceive.
Value	0x21U

4.4.3.1.26. PDUR_SID_UP_CANCELTXREQ

Purpose	Definition of service ID for PduR_UpCancelTransmit.
Value	0x1CU

4.4.3.1.27. PDUR_SID_UP_CHANGEPARAREQ

Purpose	Definition of service ID for PduR_UpChangeParameter.
Value	0x1DU

4.4.3.1.28. PDUR_SID_UP_TX

Purpose	Definition of service ID for PduR_UpTransmit.
Value	0x14U

4.4.3.1.29. PduR_GetVersionInfo

Purpose	Get version information.
----------------	--------------------------

Value	<pre>do { \ /* get version info of PduR module */ \ (versionInfo)->vendorID = PDUR_VENDOR_ID; \ (versionInfo)->moduleID = PDUR_MODULE_ID; \ (versionInfo)->sw_major_version = PDUR_SW_MAJOR_VERSION; \ (versionInfo)->sw_minor_version = PDUR_SW_MINOR_VERSION; \ (versionInfo)->sw_patch_version = PDUR_SW_PATCH_VERSION; \ } while(0)</pre>
Description	This service returns the version information of this module.

4.4.3.2. Objects

4.4.3.2.1. PduR_State

Purpose	Variable holding the State of the PDU Router.
Type	PduR_StateType

4.4.3.2.2. PduR_gConfigPtr

Purpose	PduR_gConfigPtr global variable for the pointer to the config of PduR.
Type	<code>const PduR_PBConfigType *</code>

4.4.3.3. Functions

4.4.3.3.1. PduR_GetConfigurationId

Purpose	Get configuration ID.
Synopsis	<pre>PduR_PBConfigIdType PduR_GetConfigurationId (void);</pre>
Service ID	0x10
Sync/Async	Synchronous
Reentrancy	Reentrant
Return Value	Identifier of the post-build time configuration. For enabled DET the invalid configuration Id 'PDUR_INVALID_CONFIGURATION_ID' is returned.

Description	Returns the unique identifier of the post-build time configuration of the PDU Router.
--------------------	---

4.4.3.3.2. PduR_Init

Purpose	PduR_Init - Initializes the PDU Router. Function to initialize the PduR module. First function to be called of PduR. The module calling the function PduR_Init has to include PduR_PBCfg.h. The invocation of the PduR_Init function is PduR_Init(&(PDUR_CONFIG_NAME.PduR_RootConfig));.	
Synopsis	<pre>void PduR_Init (const PduR_PBCfgType * ConfigPtr);</pre>	
Service ID	0x01	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	ConfigPtr	Pointer to post build configuration.

4.4.3.3.3. PduR_IsValidConfig

Purpose	Validate configuration.	
Synopsis	<pre>Std_ReturnType PduR_IsValidConfig (const void * ConfigPtr);</pre>	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (in)	ConfigPtr	Pointer to configuration structure that holds the PduR module post-build-time configuration data.
Return Value	Function execution success status	
	E_OK	the provided module configuration is valid
	E_NOT_OK	the provided module configuration is invalid
Description	<p>Checks if the post build configuration is valid. A configuration is invalid if</p> <ul style="list-style-type: none"> ▶ the platform signature does not match. ▶ the published information signature does not match. ▶ the link time signature does not match. ▶ the compile time signature does not match. ▶ the function is called with a null pointer. 	

4.4.3.3.4. PduR_LoRxIndication

Purpose	This service is called by the <Lo> module to indicate a received I-PDU.	
Synopsis	<pre>void PduR_LoRxIndication (PduIdType Rx- PduId , PduInfoType * PduInfoPtr);</pre>	
Service ID	0x42	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	RxPduId	ID of the received I-PDU.
	PduInfoPtr	Contains the length (SduLength) of the received I-PDU and a pointer to a buffer (SduDataPtr) containing the I-PDU.

4.4.3.3.5. PduR_LoTpCopyRxData

Purpose	CopyRxData function.	
Synopsis	<pre>BufReq_ReturnType PduR_LoTpCopyRxData (PduId- Type RxPduId , const PduInfoType * PduIn- foPtr , PduLengthType * BufferSizePtr);</pre>	
Service ID	0x32	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	RxPduId	Identification of the received I-PDU.
	PduInfoPtr	Pointer to the buffer (SduDataPtr) and its length (SduLength) containing the data to be copied by PDU Router module in case of gateway or upper layer module in case of reception.
Parameters (out)	BufferSizePtr	Available receive buffer after data has been copied.
Return Value	Result of buffer request	
	BUFREQ_OK	Data copied successfully.
	BUFREQ_E_NOT_OK	Data was not copied because an error occurred.
Description	This function is called when a transport protocol module has data to copy for the receiving module. Several calls may be made during one transportation of an I-PDU.	

	The service shall provide the currently available buffer size when invoked with info.SduLength equal to 0.
--	--

4.4.3.3.6. PduR_LoTpCopyTxData

Purpose	CopyTxData function.	
Synopsis	<pre>BufReq_ReturnType PduR_LoTpCopyTxData (PduIdType Tx- PduId , PduInfoType * PduInfoPtr , RetryInfoType * RetryInfoPtr , PduLengthType * AvailableDataPtr);</pre>	
Service ID	0x36	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	TxPduId	Identification of the transmitted I-PDU.
	RetryInfoPtr	<p>This parameter is used to retransmit data because problems during the last service call. If the I-PDU is transmitted from a local module (e.g. DCM) the PDU router module will just forward the parameter value without check. If the I-PDU is gatewayed from another bus, the PDU Router module will make the following interpretation: If the transmitted TP I-PDU does not support the retry feature a NULL_PTR is provided. It indicates that the copied transmit data can be removed from the buffer after it has been copied. If the retry feature is used by the Tx I-PDU, RetryInfoPtr must point to a valid RetryInfoType element. If TpDataState indicates TP_CONFENDING, the previously copied data must remain in the TP buffer to be available for error recovery. TP_DATA_CONF indicates that all data that have been copied so far are confirmed and can be removed from the TP buffer. Data copied by this API call are excluded and will be confirmed later. TP_DATARETRY indicates that this API call shall copy already copied data in order to recover from</p>

		an error. In this case AvailableDataPtr specifies the offset of the first byte to be copied by the API call.
Parameters (out)	PduInfoPtr	Provides destination buffer and the number of bytes to copy. In case of gateway the PDU Router module will copy, otherwise the source upper layer module will copy the data. If not enough transmit data is available, no data is copied. The transport protocol module will retry. A copy size of 0 can be used to indicate state changes in the retry parameter.
	AvailableDataPtr	Indicates the remaining number of bytes that are available in the PduR Tx buffer. AvailableDataPtr can be used by TP modules that support dynamic payload lengths (e.g. FlexRay ISO Transport Layer) to determine the size of the following CFs.
Return Value	Result of buffer request	
	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. TP layer might retry later on. No data has been copied.
	BUFREQ_E_NOT_OK	Data has not been copied. Request failed.
Description	This function is called by the transport protocol module to query the transmit data of an I-PDU segment. Each call to this function copies the next part of the transmit data until TpDataState indicates TP_DATARETRY. In this case the API restarts to copy the data beginning at the location indicated by AvailableDataPtr. The service shall provide the size of the remaining data when invoked with info.SduLength equal to 0.	

4.4.3.3.7. PduR_LoTpRxIndication

Purpose	TpRxIndication function.
Synopsis	<pre>void PduR_LoTpRxIndication (PduId- Type RxPduId , NotifResultType Result);</pre>
Service ID	0x33

Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	RxPduId	Identification of the received I-PDU.
	Result	Result of the reception.
Description	This service is called by the transport protocol module after an I-PDU has been received successfully or when an error occurred. It is also used to confirm cancellation of an I-PDU.	

4.4.3.3.8. PduR_LoTpStartOfReception

Purpose	StartOfReception function.	
Synopsis	BufReq_ReturnType PduR_LoTpStartOfReception (PduIdType RxPduId , PduLengthType TpSduLength , PduLengthType * BufferSizePtr);	
Service ID	0x34	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	RxPduId	Identification of the received I-PDU.
	TpSduLength	Total length of the PDU to be received.
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	Result of buffer request	
	BUFREQ_OK	Connection has been accepted. BufferSizePtr indicates the available receive buffer. Reception is continued.
	BUFREQ_E_OVFL	No Buffer of the required length can be provided. Reception is aborted. BufferSizePtr remains unchanged.
	BUFREQ_E_NOT_OK	Connection has been rejected. Reception is aborted. BufferSizePtr remains unchanged.
Description	This function will be called by the transport protocol module at the start of receiving an I-PDU. The I-PDU 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.	

4.4.3.3.9. PduR_LoTpTxConfirmation

Purpose	TpTxConfirmation function.	
Synopsis	<pre>void PduR_LoTpTxConfirmation (PduId- Type TxPduId , NotifResultType Result);</pre>	
Service ID	0x37	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	TxPduId	Identification of the transmitted I-PDU.
	Result	Result of the transmission of the I-PDU.
Description	This service is called by a transport protocol module after the I-PDU has been transmitted on its network, the result will reveal if the transmission was successful or not.	

4.4.3.3.10. PduR_LoTriggerTransmit

Purpose	The lower layer communication module requests the buffer of the SDU for transmission from the upper layer module.	
Synopsis	<pre>Std_ReturnType PduR_LoTriggerTransmit (PduId- Type TxPduId , PduInfoType * PduInfoPtr);</pre>	
Service ID	0x41	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	TxPduId	ID of the SDU that is requested to be transmitted.
	PduInfoPtr	Contains the length (SduLength) of the received I-PDU and a pointer to a buffer (SduDataPtr) containing the I-PDU.
Return Value	Function execution success status	
	E_OK	SDU has been copied and SduLength indicates the number of copied bytes.
	E_NOT_OK	No SDU has been copied. PduInfoPtr must not be used since it may contain a NULL pointer or point to invalid data.

4.4.3.3.11. PduR_LoTxConfirmation

Purpose	This service is called by the <Lo> module to confirm the transmission of an I-PDU.	
Synopsis	<code>void PduR_LoTxConfirmation (PduIdType TxPduId);</code>	
Service ID	0x40	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	TxPduId	ID of the I-PDU that has been transmitted.

4.4.3.3.12. PduR_UpCancelReceive

Purpose	This service is called by the <Up> module to request cancellation from an upper layer module of an I-PDU in a lower layer transport protocol module.	
Synopsis	<code>Std_ReturnType PduR_UpCancelReceive (PduIdType RxPduId);</code>	
Service ID	0x21	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	RxPduId	Identification of the Rx I-PDU.
Return Value	Function execution success status	
	E_OK	Request accepted (but not yet performed).
	E_NOT_OK	Request not accepted (e.g. cancellation not possible)

4.4.3.3.13. PduR_UpCancelTransmit

Purpose	This service is called by the <Up> module for cancellation of an ongoing transmission of an transport protocol module I-PDU.	
Synopsis	<code>Std_ReturnType PduR_UpCancelTransmit (PduIdType TxPduId);</code>	
Service ID	0x1C	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	TxPduId	Identification of the Tx-PDU ID to be canceled.
Return Value	Function execution success status	

	E_OK	Request is accepted by the destination module.
	E_NOT_OK	Request is not accepted by the destination module.

4.4.3.3.14. PduR_UpChangeParameter

Purpose	This service is called by the <Up> module to request to change a specific transport protocol parameter (e.g. block-size). The affected transport protocol module is selected using the Rx I-PDU ID.	
Synopsis	<pre>Std_ReturnType PduR_UpChangeParameter (PduIdType RxPduId , TPPParameterType TPPParameter , uint16 TPPParameterValue);</pre>	
Service ID	0x1D	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	RxPduId	Identification of the Rx I-PDU to which the parameter the request shall affect.
	TPPParameter	The selected parameter that the request shall changed.
	TPPParameterValue	The value that the request shall change to.
Return Value	Function execution success status	
	E_OK	Request is accepted.
	E_NOT_OK	Request is not accepted.

4.4.3.3.15. PduR_UpTransmit

Purpose	This function is called by <Up> module to request transmission of an I-PDU.	
Synopsis	<pre>Std_ReturnType PduR_UpTransmit (PduIdType Tx- PduId , const PduInfoType * PduInfoPtr);</pre>	
Service ID	0x14	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	TxPduId	Identification of the Tx-PDU ID.

	<code>PduInfoPtr</code>	Length and pointer to the buffer of the I-PDU
Return Value	Function execution success status	
	<code>E_OK</code>	Request is accepted by the destination module.
	<code>E_NOT_OK</code>	Request is not accepted by the destination module.

4.4.4. Integration notes

4.4.4.1. Exclusive areas

This section describes the exclusive areas used by the `PduR` module.

4.4.4.1.1. SCHM_PDUR_EXCLUSIVE_AREA_0

Protected data structures	All shared data that shall be protected from mutual access.
Recommended locking mechanism	This exclusive area must always be protected by a locking mechanism. The options for locking are described in the EB tresos AutoCore Generic documentation . Refer to the section Mapping exclusive areas in the basic software modules in the Integration notes section for details.

4.4.4.2. Production errors

Production errors are not reported by the `PduR` module.

4.4.4.3. Memory mapping

General information about memory mapping is provided in the [EB tresos AutoCore Generic documentation](#). Refer to the section [Memory mapping and compiler abstraction](#) in the [Integration notes](#) section for details.

The following table provides the list of sections that may be mapped for this module:

Memory section
VAR_NOINIT_UNSPECIFIED
VAR_FAST_NOINIT_UNSPECIFIED
VAR_INIT_UNSPECIFIED
CODE
CONFIG_DATA_UNSPECIFIED
CONST_8
CONST_32
CONST_UNSPECIFIED
CODE_CC_BLOCK

4.4.4.4. Integration requirements

WARNING



Integration requirements list is not exhaustive

The following list of integration requirements helps you to integrate your product. However, this list is not exhaustive. You also require information from the user's guide, release notes, and EB tresos AutoCore known issues to successfully integrate your product.

4.4.4.4.1. PduR.EB.IntReq.RestrictTpGwToSF

Description	Restriction of TP gateway to single frames. TP gateway (1:n, n>1) of an I-PDU from a source transport protocol module to multiple destination transport protocol modules is restricted to single frames (SF) on the Tx-side. This implies that the completely stored I-PDU is fetched with a single call by PduR_<LoTp>CopyTxData. Calling the function PduR_<LoTp>CopyTxData with a size PduInfoPtr->SduLength different to the complete I-PDU results in return value BUFREQ_E_BUSY. Calling the function PduR_<LoTp>CopyTxData after requesting the available size of data with PduInfoPtr->SduLength = 0 by PduR_<LoTp>CopyTxData returns an AvailableDataSize of the completely stored I-PDU except the last one which returns 0.
Rationale	On a transport protocol module, an I-PDU can be transported in multiple N-PDUs (FF and CFs) or in a single N-PDU (SF). The typical case is that an I-PDU transported in multiple N-PDUs does not multicast I-PDUs (i.e. physical addressing) and in a single N-PDU may be multicast I-PDUs (i.e. functional addressing). Furthermore, the consumption of hardware resources (RAM, run time) is reduced.

4.4.4.4.2. PduR.EB.IntReq.BlockLoTpCopyTxDataForTpMulticast

Description	Blocked PduR_LoTpCopyTxData unless all LoTp_Transmit are called for TP multicast transmission. PduR_LoTpCopyTxData() is blocked for TP multicast transmission (1:n, n>1) unless all calls of LoTp_Transmit() are executed within PduR_UpTransmit().
Rationale	This reduces the number of possible race conditions.

4.4.4.4.3. PduR.EB.IntReq.BlockLoTpCopyTxDataForDirectTpGw

Description	Blocked PduR_LoTpCopyTxData, unless all LoTp_Transmit are called for a direct TP gateway. PduR_LoTpCopyTxData() is blocked for a direct TP gateway, unless all calls of LoTp_Transmit() are executed within PduR_LoTpRxIndication().
Rationale	This reduces the number of possible race conditions.

4.4.4.4.4. PduR.EB.IntReq.RestrictNto1toSingleActivatedRPath

Description	For N:1 PDU routing, there shall be at most one activated source PDU of the gateway routing path.
Rationale	As stated in RfC 71728, comment #3, it will be ensured that at run-time at most one of these N sources will be active. Note that this has been refined in RfC 67569, comment #84. 'Active' means, that concurrent calls are not possible, e.g. a CopyRxData from CAN1 can not be interrupted by a StartOfReception from CAN2. I.e. a tester is either used on CAN1 or CAN2, not to both.

5. Bibliography

Bibliography

- [1] *Road Vehicles - Diagnostic Systems - Keyword Protocol 2000 - Part 3: Application Layer*,
1, rue de Varembe, Case postale 56
1211 Geneva 20, Switzerland, Publish date: 1999, Issue Version ISO/DIS 14230-3, Publisher: ISO
(International Organization for Standardization)

- [2] *Road Vehicles - Unified Diagnostic Services (UDS) - Part 1: Specification and Requirements*,
1, rue de Varembe, Case postale 56
1211 Geneva 20, Switzerland, Publish date: 2013, Issue Version ISO/DIS 14229-1, Publisher: ISO
(International Organization for Standardization)

- [3] *Road vehicles - Communication between vehicle and external equipment for emissions-related diagnostics - Part 5: Emissions-related diagnostic services*,
1, rue de Varembe, Case postale 56
1211 Geneva 20, Switzerland , Publish date: 2011, Issue Version ISO/DIS 15031-5, Publisher: ISO
(International Organization for Standardization)