Document Title	Requirements on I-PDU Multiplexer
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	181

Document Status	published
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	R22-11

Document Change History			
Date	**Release**	**Changed by**	**Description**
2022-11-24	R22-11	AUTOSAR Release Management	• Fixed number of possible multiplexed PDUs
2021-11-25	R21-11	AUTOSAR Release Management	• No content changes
2020-11-30	R20-11	AUTOSAR Release Management	• No content changes
2019-11-28	R19-11	AUTOSAR Release Management	• No content changes • Changed Document Status from Final to published
2018-10-31	4.4.0	AUTOSAR Release Management	• position of I-PDUs inside a Container are dynamic when priority is used
2017-12-08	4.3.1	AUTOSAR Release Management	• Editorial changes
2016-11-30	4.3.0	AUTOSAR Release Management	• minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation
2014-10-31	4.2.1	AUTOSAR Release Management	• added Multiple PDU to Container Mapping • extension of IpduMSelectorFieldLength

2013-03-15	4.1.1	AUTOSAR Administration	• Formal updates • Updated with respect to the requirement tracing
2010-09-30	3.1.5	AUTOSAR Administration	• Legal disclaimer revised
2008-08-13	3.1.1	AUTOSAR Administration	• Legal disclaimer revised
2007-12-21	3.0.1	AUTOSAR Administration	• Document meta information extended • Small layout adaptations made
2007-01-24	2.1.15	AUTOSAR Administration	• "Advice for users" revised • "Revision Information" added
2006-11-28	2.1	AUTOSAR Administration	• Legal disclaimer revised
2006-05-16	2.0	AUTOSAR Administration	• Initial release

Disclaimer

This work (specification and/or software implementation) and the material contained in it, as released by AUTOSAR, is for the purpose of information only. AUTOSAR and the companies that have contributed to it shall not be liable for any use of the work.

The material contained in this work is protected by copyright and other types of intellectual property rights. The commercial exploitation of the material contained in this work requires a license to such intellectual property rights.

This work may be utilized or reproduced without any modification, in any form or by any means, for informational purposes only. For any other purpose, no part of the work may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

The work has been developed for automotive applications only. It has neither been developed, nor tested for non-automotive applications.

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

Contents

1 Scope of Document

The goal of this document is to define the functional and non-functional requirements on the AUTOSAR I-PDU-Multiplexer module (IPduM).

2 Conventions to be used

The representation of requirements in AUTOSAR documents follows the table specified in [TPS_STDT_00078], see Standardization Template [1], chapter Support for Traceability.

The verbal forms for the expression of obligation specified in [TPS_STDT_00053] shall be used to indicate requirements, see Standardization Template [1], chapter Support for Traceability.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as follows.

Note that the requirement level of the document in which they are used modifies the force of these words.

- MUST: This word, or the adjective "LEGALLY REQUIRED", means that the definition is an absolute requirement of the specification due to legal issues.

- MUST NOT: This phrase, or the phrase "MUST NOT", means that the definition is an absolute prohibition of the specification due to legal issues.

- SHALL: This phrase, or the adjective "REQUIRED", means that the definition is an absolute requirement of the specification.

- SHALL NOT: This phrase means that the definition is an absolute prohibition of the specification.

- SHOULD: This word, or the adjective "RECOMMENDED", means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

- SHOULD NOT: This phrase, or the phrase "NOT RECOMMENDED", means that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.

- MAY: This word, or the adjective "OPTIONAL", means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item.

An implementation, which does not include a particular option, SHALL be prepared to interoperate with another implementation, which does include the option, though perhaps with reduced functionality. In the same vein an implementation, which does include a particular option, SHALL be prepared to interoperate with another implementation, which does not include the option (except, of course, for the feature the option provides.)

3   Acronyms and abbreviations

The glossary below includes acronyms and abbreviations relevant to I-PDU Multiplexer that are not included in the AUTOSAR Glossary [2].

Acronym:	Description:
contained I-PDU	I-PDU being transported inside a Container I-PDU
Container I-PDU	I-PDU serving as the Container collecting several I-PDUs to be transported together (e.g. in one frame)
dynamic part	According to the value of the selector field some parts of the I-PDU have a different layout. These parts of the I-PDU that can contain different signals are called dynamic part. The dynamic part must not necessarily be contiguous.
Multi-PDU-to-Container Mapping	Multi-PDU-to-Container Mapping means using the same Container I-PDU with more than one contained I-PDU. To be able to recognize each I-PDU at reception a header containing an ID and the length is placed in front of each contained I-PDU.
multiplexed I-PDU	I-PDU multiplexing means using the same PCI of an I-PDU with more than one unique layout of its SDU. A selector field is a part of the SDU of the multiplexed I-PDU. It is used to distinguish the different layouts of the multiplexed I-PDUs from each other
PCI	"Protocol Control Information" A description can be found in AUTOSAR_EXP_LayeredSoftwareArchitecture [3].
PDU	"Protocol Data Unit" A description can be found in AUTOSAR_EXP_LayeredSoftwareArchitecture [3].
PduR	Module that transfers I-PDUs from one module to another module. The I-PDU Router can be utilized for gateway operations and for internal routing purposes.
SDU	"Service Data Unit" A description can be found in AUTOSAR_EXP_LayeredSoftwareArchitecture [3].
selector field	The selector field is part of a multiplexed I-PDU. It consists of contiguous bits. The value of the selector field selects the layout of the multiplexed part of the I-PDU.
static part	Some parts/signals of the I-PDU may be the same regardless of the selector field. Such a part is called static part. The static part must not necessarily be contiguous.

4 Requirements Specification

This chapter describes all requirements driving the work to define the <Name of Specification>.

4.1 Functional Overview

The AUTOSAR IPduM is placed next to the PduR-Module within the AUTOSAR layered architecture.
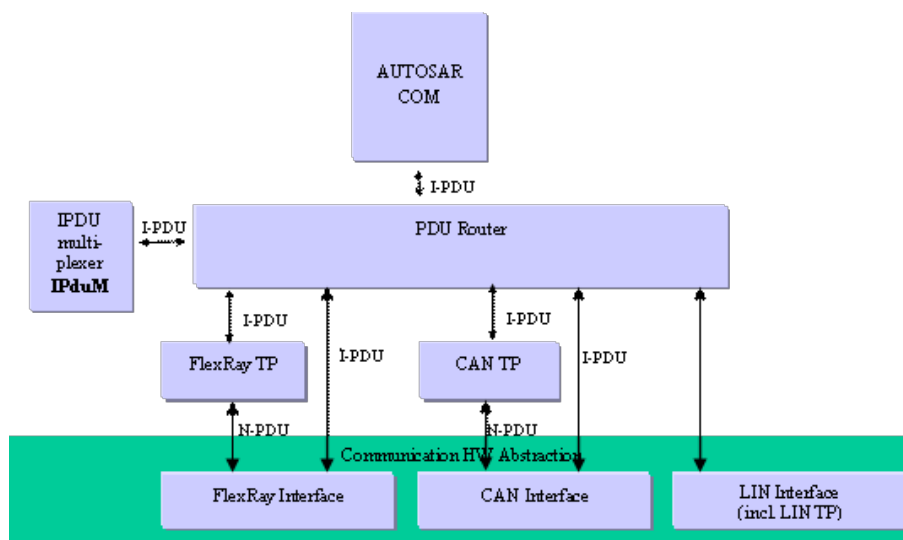
Figure 4.1: Placement of the IPduM in the AUTOSAR layered architecture.

I-PDU multiplexing means using the same PCI of an I-PDU with more than one unique layout of its SDU. A selector field is a part of the SDU of the multiplexed I-PDU. It is used to distinguish the different layouts of the multiplexed I-PDUs from each other.

Multiplexing of I-PDUs is currently known from CAN, but is not restricted to this communication system.

The I-PDU Multiplexer Module is responsible to combine appropriate I-PDUs received from COM via IPDU-router to new, multiplexed I-PDUs sent back to the I-PDU Router on sender-side. On receiver-side it is responsible to interpret the content of multiplexed I-PDUs and provide COM via IPDU-router with its appropriate separated I-PDUs by taking into account the value of the selector field.

4.2 Functional Requirements

4.2.1 Configuration

4.2.1.1 [SRS_IpduM_02800] For a multiplexed I-PDU there shall be exactly one selector field

[SRS_IpduM_02800] For a multiplexed I-PDU there shall be exactly one selector field ⌈

Description:	For a multiplexed I-PDU there shall be exactly one selector field.
Rationale:	reduce complexity
Use Case:	–
Dependencies:	–
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.1.2 [SRS_IpduM_02801] The size in bits of the selector field shall be configurable

[SRS_IpduM_02801] The size in bits of the selector field shall be configurable ⌈

Description:	The size in bits of the selector field shall be configurable in a range of one bit and 16 bits.
Rationale:	The size of the selector field should be according to the number of different states of the multiplexer required. Typically a four-bit selector field is sufficient.
Use Case:	A four-way multiplexing requires two bits
Dependencies:	SRS_IpduM_02803
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.1.3 [SRS_IpduM_02802] The position of the selector field within the PDU shall be configurable

[SRS_IpduM_02802] The position of the selector field within the PDU shall be configurable ⌈

Description:	The position of the selector field within the PDU shall be configurable. The configuration shall follow the following rules: ● contiguous bits ● static size
Rationale:	Downwards compatibility. Do not change layout of static signals when turning an I-PDU into a multiplexed I-PDU.
Use Case:	Some CAN messages became multiplexed messages because the space was not sufficient for a new signal anymore. The selector field was placed somewhere in the PDU without moving the 'old' signals.
Dependencies:	–
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.1.4 [SRS_IpduM_02815] The position and the size shall be configurable at compile time

[SRS_IpduM_02815] The position and the size shall be configurable at compile time ⌈

Description:	The position and the size shall be configurable at compile time.
Rationale:	Minimize configuration overhead by requiring only compile time configuration
Use Case:	–
Dependencies:	SRS_IpduM_02801, SRS_IpduM_02802
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.1.5 [SRS_IpduM_02803] It shall be possible not to assign a SDU layout to the unused selector field values

[SRS_IpduM_02803] It shall be possible not to assign a SDU layout to the unused selector field values ⌈

Description:	The number of possible multiplexed SDUs within a I-PDU is 2^n, where n is the size in bits of the selector field. In some cases not all possible selector field values are needed and it shall be possible not to assign a SDU layout to these unused selector field values.
Rationale:	1. Usage of an arbitrary number of multiplexed layouts that is not necessarily a power of 2. 2. To specify the selector field with some reserved values to allow an extension of the multiplexed layouts in the future maintaining backwards compatibility.
Use Case:	If three different I-PDU layouts are needed at least a two bit selector field has to be spent (four-way multiplexer)
Dependencies:	–
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.1.6 [SRS_IpduM_02804] For each used selector field value a dynamic and static layout shall be configurable

[SRS_IpduM_02804] For each used selector field value a dynamic and static layout shall be configurable ⌈

Description:	For each used selector field value a dynamic layout shall be configurable. The static part shall be configurable per I-PDU.
Rationale:	If a selector field is introduced to a PDU this allows to leave not-multiplexed signals unchanged (downward compatibility).
Use Case:	A selector field is attached to a PDU which is currently not multiplexed. The layout of some of the 'old' signals should not be changed
Dependencies:	SRS_IpduM_02808
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.1.7 [SRS_IpduM_02808] It shall be possible that the static part of a I-PDU is zero bits long

[SRS_IpduM_02808] It shall be possible that the static part of a I-PDU is zero bits long ⌈

Description:	It shall be possible that the static part of a I-PDU is zero bits long.
Rationale:	It should be possible to multiplex the whole I-PDU.
Use Case:	BMW service messages do not use a static part
Dependencies:	SRS_IpduM_02804
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.2 Initialization

4.2.2.1 [SRS_IpduM_02809] The initial values of the static part shall be derived from the COM configuration

[SRS_IpduM_02809] The initial values of the static part shall be derived from the COM configuration ⌈

Description:	The initial values of the static part shall be derived from the COM configuration. The initial value of the dynamic part shall be derived from the COM configuration regarding the initial value of the selector field.
Rationale:	Reuse of existing COM functionality
Use Case:	–
Dependencies:	–
Supporting Material:	–

⌋*(RS_BRF_01560)*

Note:

The selector field is initialized like a normal COM signal within the dynamic part.

4.2.3 I-PDU Multiplexing

4.2.3.1 [SRS_IpduM_02806] The three parts of each multiplexed I-PDU must not necessarily be contiguous

[SRS_IpduM_02806] The three parts of each multiplexed I-PDU must not necessarily be contiguous ⌈

Description:	Each multiplexed I-PDU consists of three different parts. These parts must not necessarily be contiguous. 1. selector field 2. dynamic part 3. static part (optional, see SRS_IpduM_02808) For signals within these parts the normal signal placing rules apply, as defined in AUTOSAR_SWS_COM [4].
Rationale:	Backwards compatibility to current solutions.
Use Case:	This behaviour is currently used in Volkswagen ECUs. Example: S1 • SF is the selector field • The dynamic part consists of two subparts D1 and D2 D1 and D2 contain different signals according to the selector field value (see next table) • The static part consists of three subparts S1, S2 and S3. S1, S2 and S3 contain signals which are not multiplexed For each dynamic subpart a different signal layout may be configured. Then different signals shall be sent out according to the selector field value. Example (m1, m2, ... is a signal within the dynamic part of a I-PDU): selector field value 0 1 2 All individual signals of the multiplexed I-PDUs shall be sent and received like non multiplexed COM signals.
Dependencies:	SRS_IpduM_02804
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.3.2 [SRS_IpduM_02816] On sender side the IPduM shall combine the static and the appropriate dynamic part within IPduM

[SRS_IpduM_02816] On sender side the IPduM shall combine the static and the appropriate dynamic part within IPduM ⌈

Description:	On sender side the IPduM combines the static and the appropriate dynamic part within IPduM after receiving the I-PDUs from the PduR. When the configured triggering condition (see SRS_IpduM_02811) is fulfilled the multiplexed I-PDU is sent to the lower layer via the PduR.
Rationale:	Having a static and a dynamic part. Implementation of I-PDU multiplexing next to PduR.
Use Case:	–
Dependencies:	SRS_IpduM_02811, SRS_IpduM_02814, SRS_IpduM_02810
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.3.3 [SRS_IpduM_02819] There shall be no queuing of transmission requests on sender side

[SRS_IpduM_02819] There shall be no queuing of transmission requests on sender side ⌈

Description:	Unless a transmission request for an I-PDU of the IPduM is either • confirmed by the underlying layer • has got an internal timeout every other transmission request to the same I-PDU shall result in an error. The internal timeout shall be configurable per I-PDU.
Rationale:	Avoid IPduM internal queuing of transmission requests
Use Case:	–
Dependencies:	SRS_IpduM_02816
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.3.4 [SRS_IpduM_02811] There shall be three different triggering conditions configurable that define when the combined multiplexed I-PDUs are sent to the lower layer

[SRS_IpduM_02811] There shall be three different triggering conditions configurable that define when the combined multiplexed I-PDUs are sent to the lower layer ⌈

Description:	On sender side the IPduM combines the dynamic and static part after receiving them from COM. There shall be three different triggering conditions configurable that define when the combined multiplexed I-PDUs are sent to the lower layer (via the PduR). • SEND_ON_STATIC - the I-PDU will be sent after receiving a new static part • SEND_ON_DYNAMIC - the I-PDU will be sent after receiving a new dynamic part • SEND_ALWAYS - the I-PDU will be sent always after receiving a new static or new dynamic part
Rationale:	Implementation of I-PDU multiplexing in an own module next to PduR.
Use Case:	–
Dependencies:	SRS_IpduM_02810
Supporting Material:	–

⌋(*RS_BRF_01560*)

Note:

COM knows only about the dynamic and static part as (source) I-PDUs and not about the combined multiplexed I-PDU. COM can only guarantee the minimum temporal distance of the source I-PDUs. Therefore the minimum temporal distance for the multiplexed IPDU might be violated, for example if two different dynamic source I-PDUs are sent, having SEND_ON_DYNAMIC mode configured.

4.2.3.5 [SRS_IpduM_02817] On receiver side the IPduM extracts the static and dynamic parts of the multiplexed I-PDU

[SRS_IpduM_02817] On receiver side the IPduM extracts the static and dynamic parts of the multiplexed I-PDU ⌈

Description:	On receiver side the IPduM extracts the static and dynamic parts of the multiplexed I-PDU received from the PduR and sends them to COM as standard IPDUs via the PduR. In an optimization implementation it shall be allowed to send the I-PDU directly to COM.

▽

△

Rationale:	Having a static and a dynamic part.
	Implementation of I-PDU multiplexing next to PduR.
Use Case:	–
Dependencies:	–
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.3.6 [SRS_IpduM_02818] The IPduM confirms to COM the static part of the multiplexed I-PDU and the dynamic part

[SRS_IpduM_02818] The IPduM confirms to COM the static part of the multi-plexed I-PDU and the dynamic part ⌈

Description:	After receiving send confirmations from the PduR the IPduM confirms to COM the static part of the multiplexed I-PDU and the dynamic part according to value of the selector field.
Rationale:	Having a static and a dynamic part.
	Implementation of I-PDU multiplexing next to PduR.
Use Case:	–
Dependencies:	SRS_IpduM_02813, SRS_IpduM_02814
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.2.3.7 [SRS_IpduM_02814] The confirmation shall depend upon selector field

[SRS_IpduM_02814] The confirmation shall depend upon selector field ⌈

Description:	On I-PDU Multiplexing the same I-PDU-ID is sent out with different signal layouts. This must be considered while implementing the confirmation handling. That is the confirmation given by the IpduM to the upper layer shall match the correct transmission request of the multiplexed I-PDU instance.
Rationale:	Ensure that confirmations are valid according to the selector field value. Since the confirmations of the underlying layer are given per multiplexed I-PDU the correct assignment of these confirmations to the multiplexed I-PDUs must be ensured by the IpduM.
Use Case:	–
Dependencies:	SRS_IpduM_02810, SRS_IpduM_02813

▽

△

Supporting Material:	–

⌋(*RS_BRF_01560*)

4.2.4 Dynamic I-PDU to Container Mapping

4.2.4.1 [SRS_IpduM_02820] Dynamic I-PDU Mapping

[SRS_IpduM_02820] Dynamic I-PDU Mapping ⌈

Description:	One or more I-PDUs shall be mappable to a Container-PDU. If an I-PDU is mapped to a Container, it shall be subject to the Multi-PDU-to-Container Mapping.
Rationale:	Mapping multiple PDUs to one frame reduces bus load by improving usage of enhanced bandwidth.
Use Case:	Efficient transmission of small PDUs of high bandwidth busses
Dependencies:	A Container-PDU shall either support static or dynamic mapping (see SRS_IpduM_02825) but no mixture.
Supporting Material:	–

⌋(*RS_BRF_01716*)

4.2.4.2 [SRS_IpduM_02821] The temporal order of I-PDUs shall be preserved

[SRS_IpduM_02821] The temporal order of I-PDUs shall be preserved ⌈

Description:	The temporal order in which the I-PDUs are extracted from a Container PDU and passed to upper layer shall be the same as they have been passed to Ipdu M on Transmission side.
Rationale:	Multi-PDU-to-Container Mapping shall have no effect on the order, thus avoiding problems e.g. with sequence counters
Use Case:	Temporal order of instances of the same I-PDU must not be changed
Dependencies:	–
Supporting Material:	–

⌋(*RS_BRF_01716*)

4.2.4.3 [SRS_IpduM_02822] Two different Header Sizes shall be supported

[SRS_IpduM_02822] Two different Header Sizes shall be supported ⌈

Description:	The Multi-PDU-to-Container Mapping shall support headers of 64 bits and 32 bits.
Rationale:	On Ethernet 64 bits are used which is not suitable for busses with smaller frame sizes.
Use Case:	64 bits header size on Ethernet, 32 bits on all other busses.
Dependencies:	–
Supporting Material:	–

⌋*(RS_BRF_01716)*

4.2.4.4 [SRS_IpduM_02823] The position of I-PDUs inside a Container shall be dynamic

[SRS_IpduM_02823] The position of I-PDUs inside a Container shall be dynamic ⌈

Description:	The position of I-PDUs inside a Container PDU only depends on the size of the payload and headers of I-PDUs that have been added earlier.
Rationale:	To avoid overhead on the bus a Container PDU only contains I-PDUs actually beeing transmitted.
Use Case:	Efficient transmission of small I-PDUs on high bandwidth busses
Dependencies:	–
Supporting Material:	–

⌋*(RS_BRF_01716)*

4.2.4.5 [SRS_IpduM_02824] The ID used in the header shall be independent of the Container

[SRS_IpduM_02824] The ID used in the header shall be independent of the Container ⌈

Description:	Generally, the position of I-PDUs inside a Container PDU depends on the size of the payload and headers of I-PDUs that have been added earlier. Additionally, the position could be influenced, if using priorities for I-PDUs. I-PDUs with decreasing priority are put one after another in the Container PDU and I-PDUs with the same priority are put to the Container I-PDU with respect to their occurrence (earliest first, a.s.o)

▽

△

Rationale:	Receiving ECUs shall be able to extract contained I-PDUs independently of the actual Container they are received in.
Use Case:	Efficient transmission of small I-PDUs on high bandwidth buses. If using priority, grant availability in the network of high priority I-PDUs.
Dependencies:	–
Supporting Material:	–

⌋*(RS_BRF_01716)*

4.2.4.6 [SRS_IpduM_02826] The position of I-PDUs inside a Container shall be dynamic

[SRS_IpduM_02826] ⌈

Description:	The position of I-PDUs inside a Container shall be dynamic
Rationale:	Generally, the position of I-PDUs inside a Container PDU depends on the size of the payload and headers of I-PDUs that have been added earlier. Additionally, the position could be influenced, if using priorities for I-PDUs. I-PDUs with decreasing priority are put one after another in the Container PDU and I-PDUs with the same priority are put to the Container I-PDU with respect to their occurrence (earliest first, a.s.o).
Use Case:	Efficient transmission of small I-PDUs on high bandwidth buses. If using priority, grant availability in the network of high priority I-PDUs.
Dependencies:	–
Supporting Material:	–

⌋*(RS_BRF_01716)*

4.2.5 Static I-PDU to Container Mapping

4.2.5.1 [SRS_IpduM_02825] Static I-PDU Mapping

[SRS_IpduM_02825]{DRAFT} **Static I-PDU Mapping** ⌈

Description:	Multiple I-PDUs shall be mappable to a static position into a Container-PDU. If an I-PDU is mapped statically to a Container it is always transported in the same position of the Container-PDU.
Rationale:	Support scenarios of Container Mapping where no header shall be used to reduce bandwidth in case always all contained PDU shall be sent.

▽

△

Use Case:	Efficient transmission of small PDUs of high bandwidth busses
Dependencies:	A Container-PDU shall either support static or dynamic mapping (see SRS_ IpduM_02820) but no mixture.
Supporting Material:	–

⌋(*RS_BRF_01716*)

4.2.6 Shutdown Operation

Actual no special shutdown operations are required.

4.2.7 Fault Operation

Actual no special fault operations are required.

4.3 Non-Functional Requirements (Qualities)

4.3.1 Timing Requirements / Resource Usage

4.3.1.1 [SRS_IpduM_02807] The I-PDU Multiplexer module shall be designed in a way that it does not produce any additional runtime

[SRS_IpduM_02807] The I-PDU Multiplexer module shall be designed in a way that it does not produce any additional runtime ⌈

Description:	The I-PDU Multiplexer module shall be designed in a way that it does not produce any additional runtime compared to normal COM signal processing. To reach this goal it may be required to make I-PDU multiplexing a pre-compile time option. This assumption can be made if necessary.
Rationale:	Do not put additional load on systems that do not require I-PDU-multiplexing.
Use Case:	DC and Volvo do not plan to use I-PDU-multiplexing
Dependencies:	–
Supporting Material:	–

⌋(*RS_BRF_01560*)

4.4 Requirements to other modules

4.4.1 Configuration of PduR

4.4.1.1 [SRS_IpduM_02810] The PduR shall be configured to send parts of multiplexed I-PDUs to the IPduM on sender side

[SRS_IpduM_02810] The PduR shall be configured to send parts of multiplexed I-PDUs to the IPduM on sender side ⌈

Description:	The PduR shall be configured to send parts of multiplexed I-PDUs to the IPduM after they were received from COM on sender side.
Rationale:	Implementation of I-PDU multiplexing next to PduR.
Use Case:	–
Dependencies:	SRS_IpduM_02811, SRS_IpduM_02814, SRS_IpduM_02816
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.4.1.2 [SRS_IpduM_02813] The PduR shall be configured to send confirmations related to multiplexed I-PDUs to IPduM after receiving them from the lower layer

[SRS_IpduM_02813] The PduR shall be configured to send confirmations related to multiplexed I-PDUs to IPduM after receiving them from the lower layer ⌈

Description:	The PduR shall be configured to send confirmations related to multiplexed I-PDUs to IPduM after receiving them from the lower layer.
Rationale:	Implementation of I-PDU multiplexing next to PduR.
Use Case:	–
Dependencies:	SRS_IpduM_02814, SRS_IpduM_02816, SRS_IpduM_02818
Supporting Material:	–

⌋*(RS_BRF_01560)*

4.4.1.3 [SRS_IpduM_02812] The PduR shall be configured to send multiplexed I-PDUs for de-multiplexing to the IPduM after they were received from the lower layer

[SRS_IpduM_02812] The PduR shall be configured to send multiplexed I-PDUs for de-multiplexing to the IPduM after they were received from the lower layer ⌈

Description:	The PduR shall be configured to send multiplexed I-PDUs for de-multiplexing to the IPduM after they were received from the lower layer.
Rationale:	Implementation of I-PDU multiplexing next to PduR.
Use Case:	–
Dependencies:	SRS_IpduM_02810
Supporting Material:	–

⌋*(RS_BRF_01560)*

5 Requirements Tracing

The following table references the features specified in [5] and links to the fulfillments of these.

Feature	Description	Satisfied by
[RS_BRF_01560]	AUTOSAR communication shall support mapping of signals into transferrable protocol data units	[SRS_IpduM_02800] [SRS_IpduM_02801] [SRS_IpduM_02802] [SRS_IpduM_02803] [SRS_IpduM_02804] [SRS_IpduM_02806] [SRS_IpduM_02807] [SRS_IpduM_02808] [SRS_IpduM_02809] [SRS_IpduM_02810] [SRS_IpduM_02811] [SRS_IpduM_02812] [SRS_IpduM_02813] [SRS_IpduM_02814] [SRS_IpduM_02815] [SRS_IpduM_02816] [SRS_IpduM_02817] [SRS_IpduM_02818] [SRS_IpduM_02819]
[RS_BRF_01716]	AUTOSAR communication shall support to aggregate multiple PDUs to one PDU dynamically	[SRS_IpduM_02820] [SRS_IpduM_02821] [SRS_IpduM_02822] [SRS_IpduM_02823] [SRS_IpduM_02824] [SRS_IpduM_02825] [SRS_IpduM_02826]

6 References

[1] Standardization Template
AUTOSAR_TPS_StandardizationTemplate

[2] Glossary
AUTOSAR_TR_Glossary

[3] Layered Software Architecture
AUTOSAR_EXP_LayeredSoftwareArchitecture

[4] Specification of Communication
AUTOSAR_SWS_COM

[5] Requirements on AUTOSAR Features
AUTOSAR_RS_Features