| Document Title | Specification of a Transport Layer for SAE J1939 |
| --- | --- |
| Document Owner | AUTOSAR |
| Document Responsibility | AUTOSAR |
| Document Identification No | 425 |
| Document Classification | Standard |

| Document Status | Final |
| --- | --- |
| Part of AUTOSAR Release | 4.2.2 |

# Document Change History

| Release | Changed by | Change Description |
| --- | --- | --- |
| 4.2.2 | AUTOSAR Release Management | • Fixed retry behavior<br>• Clarified effect of MetaData on SA/DA/Protocol<br>• Introduction of further error classes |
| 4.2.1 | AUTOSAR Release Management | • Removed obsolete configuration elements<br>• Harmonized with SWS BSW General |
| 4.1.3 | AUTOSAR Release Management | • Allow for parallel reception via TP (BAM/CMDT) and directly<br>• Clarified ambiguities regarding CTS parameters |
| 4.1.2 | AUTOSAR Release Management | • NotifResultType replaced by StdReturnType<br>• Additional development errors for protocol problems<br>• Connection parameters moved to channel configuration<br>• Editorial changes<br>• Removed chapter(s) on change documentation |
| 4.1.1 | AUTOSAR Administration | • Support for dynamic CAN IDs via MetaData<br>• Support for variable block size and retry<br>• TP API improvements and fixes<br>• Adapted to new BSW General |
| 4.0.3 | AUTOSAR Administration | • API changes: CancelTransmit, CancelReceive, and ChangeParameter were changed to synchronous behavior |
| 4.0.2 | AUTOSAR Administration | • Fixed service API IDs<br>• Removed duplicate requirement J1939TP0099<br>• Clarified requirements SWS_J1939Tp_00125 and SWS_J1939Tp_00189 |
| 4.0.1 | AUTOSAR Administration | • Initial release |

**Disclaimer**

This specification 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 specification.

The material contained in this specification is protected by copyright and other types of Intellectual Property Rights. The commercial exploitation of the material contained in this specification requires a license to such Intellectual Property Rights.

This specification 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 specification may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

The AUTOSAR specifications have been developed for automotive applications only. They have neither been developed, nor tested for non-automotive applications.

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

**Advice for users**

AUTOSAR specifications may contain exemplary items (exemplary reference models, "use cases", and/or references to exemplary technical solutions, devices, processes or software).

Any such exemplary items are contained in the specifications for illustration purposes only, and they themselves are not part of the AUTOSAR Standard. Neither their presence in such specifications, nor any later documentation of AUTOSAR conformance of products actually implementing such exemplary items, imply that intellectual property rights covering such exemplary items are licensed under the same rules as applicable to the AUTOSAR Standard.

# Table of Contents

Document ID 425: AUTOSAR_SWS_SAEJ1939TransportLayer
- AUTOSAR confidential -

# 1 Introduction and Functional Overview

This specification describes the functionality, the API, and the configuration of the AUTOSAR Basic Software module J1939Tp, which implements an SAE J1939 compatible transport layer for AUTOSAR. The terms J1939Tp and J1939 Transport Layer module are used synonymously in this document.

SAE J1939 has a broad acceptance in the truck domain, and consists of several documents describing the layers of the communication protocol from the physical layer to diagnostics and the application layer. SAE J1939-21 describes the data link and transport layer, which includes two transport protocol variants:
BAM (Broadcast Announce Message) for broadcast messages, and
CMDT (Connection Mode Data Transfer) for point-to-point connections.

This specification defines how the transport protocol of SAE J1939-21 can be implemented in the AUTOSAR architecture. It only describes those parts of the implementation that are relevant to the AUTOSAR architecture. Protocol specific details like exact timings are not part of this specification. Therefore, to be able to implement the J1939 Transport Layer module, the reader of this specification must have access to the original SAE J1939-21specification document.

The module J1939Tp interfaces to the PDU Router and the CAN Interface, as shown in the following figure:
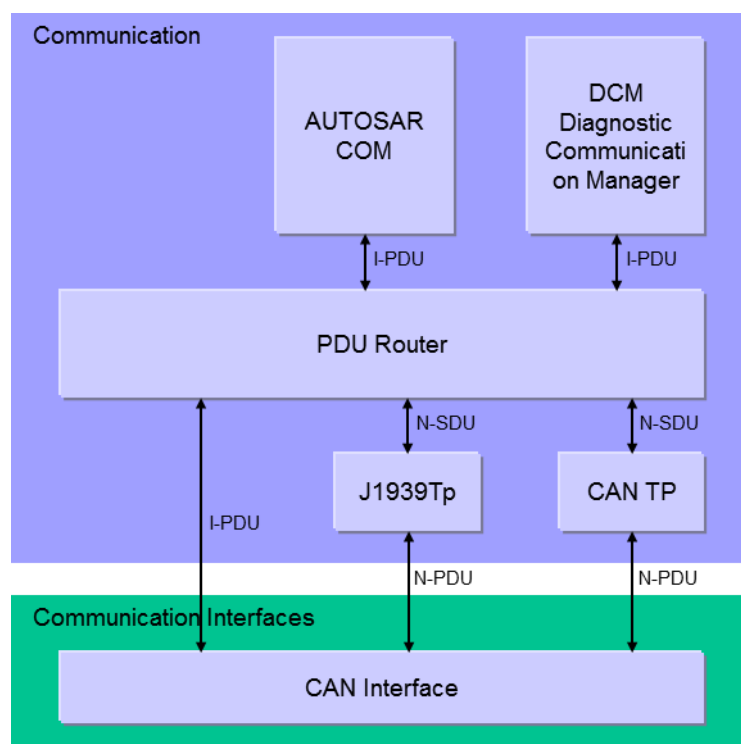
**Figure 1-1 : AUTOSAR Communication Stack**

The purpose of J1939Tp is to segment and reassemble J1939 PGNs (N-SDUs) that are longer than 8 bytes. The segmented messages are sent and received via the CAN Interface.

Document ID 425: AUTOSAR_SWS_SAEJ1939TransportLayer
- AUTOSAR confidential -

The PDU-Router (PduR) deploys AUTOSAR COM and DCM I-PDUs onto different communication protocols (e.g. CAN or J1939, segmented via transport layer or directly). At runtime, the PduR decides where to route I-PDUs based on the I-PDU (L-SDU or N-SDU) identifier. In the same way, the CanIf uses the L-SDU (I-PDU or N-PDU) identifier to decide whether a received message must be processed by one of the available transport layer modules or may be forwarded directly to the PduR.

J1939Tp supports fixed and variable size N-SDUs (I-PDUs) with more than 8 bytes. I-PDUs that do not exceed 8 bytes are exchanged directly between PduR and CanIf.

Fixed size N-SDUs are always segmented by the J1939Tp, while variable size N-SDUs are only segmented when they exceed 8 bytes. J1939Tp forwards variable size N-SDUs with an actual size of 8bytes or less and a configured maximum size of more than 8 bytes directly to the CanIf. The transport protocol variant (BAM or CMDT) is chosen based on received N-PDUs when a large N-SDU is received, and on the configuration and the actual DA when a large N-SDU is transmitted.

J1939Tp supports handling of N-PDUs and N-SDUs with variable SA, DA, and Priority. In this case, the N-PDUs and N-SDUs will contain parts of the CAN ID in the MetaData (attached to the payload).

In summary, J1939Tp provides the following functionality:
- Segmentation and direct transmission of data in transmit direction
- Reassembling and direct reception of data in receive direction
- Control of data flow
- Timeout supervision
- Detection of errors during segmentation or reassembly

# 2 Glossary, Acronyms, and Abbreviations

The following table presents a glossary of J1939 specific terms. For all other terms, please check the AUTOSAR Glossary.

| Glossary Term | Explanation |
|---|---|
| Address Claiming | Address Claiming forms the network management of SAE J1939 defined in the standard document SAE J1939/81. Address claiming assigns a temporary 8-bit identifier to each ECU connected to one J1939 network. Within this network, the 8-bit identifier is unique. The 8-bit identifier is used as source and target address of parameter groups (messages) transferred via the J1939 network. The address claiming procedure is based on the exchange of AddressClaimed messages (PGN 00EE00). |
| J1939 Diagnostics | The SAE J1939 diagnostic layer is defined in the standard document SAE J1939/73. The J1939 diagnostics is functionally similar to the UDS diagnostics, and has recently been extended to support OBD for emission relevant values. |
| Parameter | A parameter is a signal of the SAE J1939 application layer. Parameters are uniquely identified by the SPN. |
| Parameter Group | A parameter group is a message of the SAE J1939 application layer. Each parameter group contains several parameters (signals), and is uniquely identified by the PGN. |
| Transport Protocol | The SAE J1939 transport protocol is used for the segmented transmission of messages with more than 8 bytes of data. The transport protocol is defined in the network layer standard document (SAE J1939/21). |

The following table lists acronyms and abbreviations that are specific to the J1939 transport layer. For all other abbreviations, please check the AUTOSAR Glossary and the AUTOSAR BSW Module List.

| Acronym / Abbreviation | Description |
|---|---|
| BAM | Broadcast Announce Message, broadcast variant of SAE J1939 transport protocol |
| CMDT | Connection Mode Data Transfer, peer-to-peer variant of SAE J1939 transport protocol |
| DA | Destination Address, part of the 29 bit identifier of SAE J1939 messages |
| DET | Default Error Tracer, supports development and runtime error reporting |
| DMx | Diagnostic messages of the SAE J1939 diagnostics layer |
| NAME | Unique 64 bit identifier of each ECU connected to an SAE J1939 network |
| PDUF | PDU Format, part of the 29 bit identifier of SAE J1939 messages which identifies the message and determines the layout of the 29 bit identifier |
| PDUS | PDU Specific, part of the 29 bit identifier of SAE J1939 messages which identifies broadcast messages which do not have a destination address |
| PG | Parameter Group, SAE J1939 term for a specific message layout, corresponds to an N-SDU of J1939Tp |
| PGN | Parameter Group Number, unique identifier of an SAE J1939 parameter group |
| SA | Source Address, part of the 29 bit identifier of SAE J1939 messages |
| SPN | Suspect Parameter Number, unique identifier of an SAE J1939 parameter |
| TP.CM | Connection Management message (PGN 00EC00) used by SAE J1939 transport protocol, corresponds to an N-PDU of J1939Tp |
| TP.CM_BAM | Broadcast Announce Message, variant of TP.CM that initiates a BAM transmission |
| TP.CM_CTS | Connection Mode Clear To Send, variant of TP.CM that is used for handshake during CMDT transmission |
| TP.CM_EndOfMsgAck | End Of Message Acknowledge, variant of TP.CM that acknowledges correct reception of a CMDT transmission |
| TP.CM_RTS | Connection Mode Request To Send, variant of TP.CM that initiates a CMDT transmission |
| TP.Conn_Abort | Connection Abort, variant of TP.CM that terminates a CMDT transmission |
| TP.DT | Data Transfer message (PGN 00EB00) used by SAE J1939 transport protocol, corresponds to an N-PDU of J1939Tp |

# 3 Related Documentation

## 3.1 Input Documents

[1] List of Basic Software Modules
AUTOSAR_TR_BSWModuleList.pdf

[2] Layered Software Architecture
AUTOSAR_EXP_LayeredSoftwareArchitecture.pdf

[3] General Requirements on Basic Software Modules
AUTOSAR_SRS_BSWGeneral.pdf

[4] Specification of COM
AUTOSAR_SWS_COM.pdf

[5] Requirements on CAN
AUTOSAR_SRS_CAN.pdf

[6] Specification of CAN Interface
AUTOSAR_SWS_CANInterface.pdf

[7] Requirements on a Transport Layer for SAE J1939
AUTOSAR_SRS_SAEJ1939TransportLayer.pdf

[8] Specification of PDU Router
AUTOSAR_SWS_PDURouter.pdf

[9] Specification of BSW Scheduler
AUTOSAR_SWS_Scheduler.pdf

[10]	Specification of Default Error Tracer
AUTOSAR_SWS_ DefaultErrorTracer.pdf

[11]	Basic Software Module Description Template
AUTOSAR_SRS_BSWGeneral.pdf

[12]	Specification of ECU Configuration
AUTOSAR_TPS_ECUConfiguration.pdf

[13]	Specification of System Template
AUTOSAR_TPS_SystemTemplate.pdf

[14]	Specification of Memory Mapping
AUTOSAR_SWS_MemoryMapping.pdf

[15]	General Specification of Basic Software Modules
AUTOSAR_SWS_BSWGeneral.pdf

## 3.2  Related Standard Documents

[16]    SAE J1939-21(2006-12), Data Link Layer

[17]    SAE J1939-7x(2006-xx), Application Layer


## 3.3  Related specification

AUTOSAR provides a General Specification of Basic Software modules [15] (SWS BSW General), which is also valid for SAE J1939 Transport Layer.

Thus, the specification SWS BSW General shall be considered as additional and required specification for SAE J1939 Transport Layer.

# 4 Constraints and Assumptions

## 4.1 Limitations

The AUTOSAR architecture contains several communication system specific transport layers (J1939Tp, CanTp, FrTp, etc.). All of these modules need to have identical APIs, with the exception of API functions for which the PduR has separate configuration abilities.

The J1939Tp module does not implement the TriggerTransmit API, because it is only needed for time triggered bus architectures.

## 4.2 Applicability to Automotive Domains

The J1939 transport layer supports the implementation of ECUs that are designed to operate in a J1939 network.

# 5 Module Architecture

## 5.1 Dependencies on Other Modules

This section describes the relations between the J1939Tp and other AUTOSAR basic software modules. Besides the standard modules DET, EcuM, and SchM, which have interfaces to all BSW modules, J1939Tp only interacts with the PduR and the CanIf. The interfaces of J1939Tp are similar to the interfaces of CanTp.

The figure below shows the interactions between J1939Tp, PduR, and CanIf.
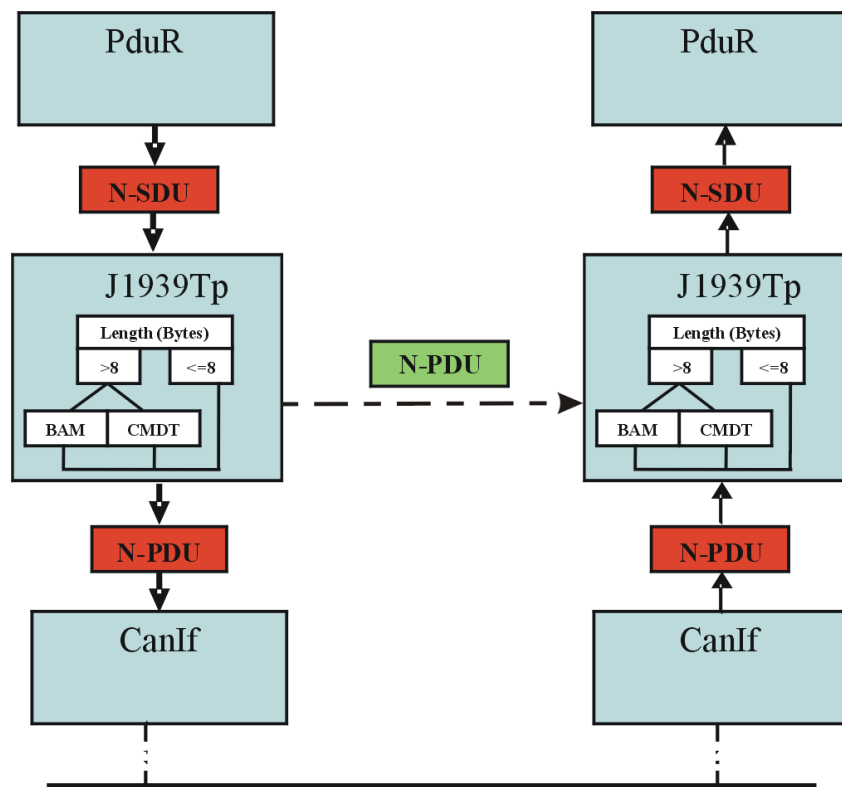


**Figure 5-1: J1939 Transport Layer interactions**

The J1939Tp's upper interface offers the PduR module access to transmitted and received N-SDUs corresponding to J1939 PGs with a maximum length of more than 8 bytes of data.

J1939 PGs with a maximum length of 8 bytes or less are exchanged directly between PduR and CanIf.

## 5.2 File Structure

AUTOSAR specifies that an ECU can be created from modules provided as object code, source code (generated or static), or both.

The decision to provide a module as object code or source code is based on a compromise between IP protection, test coverage, code efficiency and configurability at system generation time. Depending on the configurability requirements of the OEM, suppliers may deliver the J1939Tp module as object code or source code.

The file hierarchy defined in this section allows the separation of platform, compiler, and implementation specific definitions and declarations from general definitions, as well as the separation of source code and configuration.

### 5.2.1 Code File Structure

For details, refer to the chapter 5.1.6 "Code file structure" in SWS BSW General [15].

**[SWS_J1939Tp_00007]** ⌈
Internally used data types and functions shall be defined locally in the implementation source files. ⌋ ( )

This prevents visibility of these symbols outside of the J1939Tp module.

### 5.2.2 Header File Structure

The following picture shows the include file hierarchy used by the J1939 Transport Layer module.
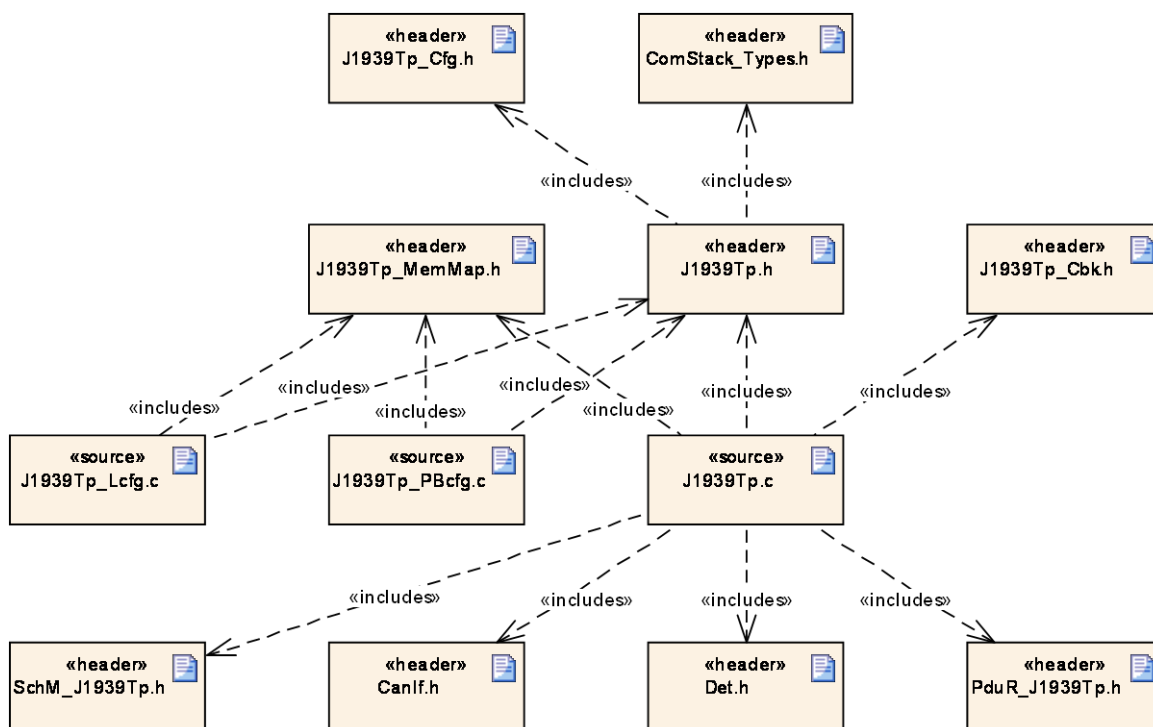


**Figure 5-2: File Structure**

**[SWS_J1939Tp_00016]** ⌈

The implementation header files shall include *ComStack_Types.h.* ⌋ ( )

**[SWS_J1939Tp_00193]** ⌈

The implementation source files shall include *Det.h* to import the default error notification API. This API is optional; the header is included depending on **ECUC_J1939Tp_00042 :** . See also section 8.6.2. ⌋ ( )

**[SWS_J1939Tp_00015]** ⌈

The implementation source files shall include *PduR_J1939Tp.h*, which contains the callbacks functions of the PduR module that are used by the J1939Tp module. See also section 8.6.1. ⌋ ( )

**[SWS_J1939Tp_00172]** ⌈

The implementation source files shall include *CanIf.h*, which contains the service functions of the CanIf module that are used by the J1939Tp module. See also section 8.6.1. ⌋ ( )

# 6 Requirements Traceability

| Requirement | Description | Satisfied by |
|---|---|---|
| - | - | SWS_J1939Tp_00007 |
| - | - | SWS_J1939Tp_00015 |
| - | - | SWS_J1939Tp_00016 |
| - | - | SWS_J1939Tp_00020 |
| - | - | SWS_J1939Tp_00022 |
| - | - | SWS_J1939Tp_00026 |
| - | - | SWS_J1939Tp_00030 |
| - | - | SWS_J1939Tp_00031 |
| - | - | SWS_J1939Tp_00032 |
| - | - | SWS_J1939Tp_00035 |
| - | - | SWS_J1939Tp_00038 |
| - | - | SWS_J1939Tp_00039 |
| - | - | SWS_J1939Tp_00040 |
| - | - | SWS_J1939Tp_00041 |
| - | - | SWS_J1939Tp_00043 |
| - | - | SWS_J1939Tp_00045 |
| - | - | SWS_J1939Tp_00046 |
| - | - | SWS_J1939Tp_00047 |
| - | - | SWS_J1939Tp_00048 |
| - | - | SWS_J1939Tp_00057 |
| - | - | SWS_J1939Tp_00060 |
| - | - | SWS_J1939Tp_00071 |
| - | - | SWS_J1939Tp_00076 |
| - | - | SWS_J1939Tp_00094 |
| - | - | SWS_J1939Tp_00095 |
| - | - | SWS_J1939Tp_00097 |
| - | - | SWS_J1939Tp_00098 |
| - | - | SWS_J1939Tp_00100 |
| - | - | SWS_J1939Tp_00101 |
| - | - | SWS_J1939Tp_00106 |
| - | - | SWS_J1939Tp_00116 |
| - | - | SWS_J1939Tp_00118 |
| - | - | SWS_J1939Tp_00119 |
| - | - | SWS_J1939Tp_00120 |
| - | - | SWS_J1939Tp_00159 |
| - | - | SWS_J1939Tp_00160 |

| - | - | SWS_J1939Tp_00162 |
|---|---|---|
| - | - | SWS_J1939Tp_00163 |
| - | - | SWS_J1939Tp_00165 |
| - | - | SWS_J1939Tp_00172 |
| - | - | SWS_J1939Tp_00173 |
| - | - | SWS_J1939Tp_00175 |
| - | - | SWS_J1939Tp_00176 |
| - | - | SWS_J1939Tp_00177 |
| - | - | SWS_J1939Tp_00180 |
| - | - | SWS_J1939Tp_00192 |
| - | - | SWS_J1939Tp_00193 |
| - | - | SWS_J1939Tp_00194 |
| - | - | SWS_J1939Tp_00195 |
| - | - | SWS_J1939Tp_00198 |
| - | - | SWS_J1939Tp_00199 |
| - | - | SWS_J1939Tp_00200 |
| - | - | SWS_J1939Tp_00203 |
| - | - | SWS_J1939Tp_00204 |
| - | - | SWS_J1939Tp_00206 |
| - | - | SWS_J1939Tp_00207 |
| - | - | SWS_J1939Tp_00208 |
| - | - | SWS_J1939Tp_00209 |
| - | - | SWS_J1939Tp_00210 |
| - | - | SWS_J1939Tp_00211 |
| - | - | SWS_J1939Tp_00212 |
| - | - | SWS_J1939Tp_00213 |
| - | - | SWS_J1939Tp_00214 |
| - | - | SWS_J1939Tp_00215 |
| - | - | SWS_J1939Tp_00216 |
| - | - | SWS_J1939Tp_00217 |
| - | - | SWS_J1939Tp_00218 |
| - | - | SWS_J1939Tp_00219 |
| - | - | SWS_J1939Tp_00220 |
| - | - | SWS_J1939Tp_00221 |
| - | - | SWS_J1939Tp_00222 |
| - | - | SWS_J1939Tp_00223 |
| - | - | SWS_J1939Tp_00224 |
| - | - | SWS_J1939Tp_00225 |
| - | - | SWS_J1939Tp_00226 |

| - | - | SWS_J1939Tp_00227 |
|---|---|---|
| - | - | SWS_J1939Tp_00228 |
| - | - | SWS_J1939Tp_00229 |
| - | - | SWS_J1939Tp_00230 |
| - | - | SWS_J1939Tp_00231 |
| - | - | SWS_J1939Tp_00232 |
| SRS_BSW_00005 | Modules of the ÂµC Abstraction Layer (MCAL) may not have hard coded horizontal interfaces | SWS_J1939Tp_99999 |
| SRS_BSW_00101 | The Basic Software Module shall be able to initialize variables and hardware in a separate initialization function | SWS_J1939Tp_00087 |
| SRS_BSW_00161 | The AUTOSAR Basic Software shall provide a microcontroller abstraction layer which provides a standardized interface to higher software layers | SWS_J1939Tp_99999 |
| SRS_BSW_00162 | The AUTOSAR Basic Software shall provide a hardware abstraction layer | SWS_J1939Tp_99999 |
| SRS_BSW_00164 | The Implementation of interrupt service routines shall be done by the Operating System, complex drivers or modules | SWS_J1939Tp_99999 |
| SRS_BSW_00167 | All AUTOSAR Basic Software Modules shall provide configuration rules and constraints to enable plausibility checks | SWS_J1939Tp_00084 |
| SRS_BSW_00168 | SW components shall be tested by a function defined in a common API in the Basis-SW | SWS_J1939Tp_99999 |
| SRS_BSW_00170 | The AUTOSAR SW Components shall provide information about their dependency from faults, signal qualities, driver demands | SWS_J1939Tp_99999 |
| SRS_BSW_00171 | Optional functionality of a Basic-SW component that is not required in the ECU shall be configurable at pre-compile-time | SWS_J1939Tp_00125 |
| SRS_BSW_00314 | All internal driver modules shall separate the interrupt frame definition from the service routine | SWS_J1939Tp_99999 |
| SRS_BSW_00323 | All AUTOSAR Basic Software Modules shall check passed API parameters for validity | SWS_J1939Tp_00188 |
| SRS_BSW_00325 | The runtime of interrupt service routines and functions that are running in interrupt context shall be kept short | SWS_J1939Tp_99999 |
| SRS_BSW_00327 | Error values naming convention | SWS_J1939Tp_00115 |
| SRS_BSW_00333 | For each callback function it shall be specified if it is called from interrupt context or not | SWS_J1939Tp_00110, SWS_J1939Tp_00114 |
| SRS_BSW_00335 | Status values naming convention | SWS_J1939Tp_00019 |
| SRS_BSW_00336 | Basic SW module shall be able to shutdown | SWS_J1939Tp_00093 |
| SRS_BSW_00337 | Classification of development errors | SWS_J1939Tp_00115 |
| SRS_BSW_00341 | Module documentation shall contains all | SWS_J1939Tp_99999 |

| | | |
|---|---|---|
| | needed informations | |
| SRS_BSW_00347 | A Naming seperation of different instances of BSW drivers shall be in place | SWS_J1939Tp_99999 |
| SRS_BSW_00357 | For success/failure of an API call a standard return type shall be defined | SWS_J1939Tp_00096 |
| SRS_BSW_00358 | The return type of init() functions implemented by AUTOSAR Basic Software Modules shall be void | SWS_J1939Tp_00087 |
| SRS_BSW_00359 | All AUTOSAR Basic Software Modules callback functions shall avoid return types other than void if possible | SWS_J1939Tp_00108, SWS_J1939Tp_00112 |
| SRS_BSW_00360 | AUTOSAR Basic Software Modules callback functions are allowed to have parameters | SWS_J1939Tp_00108, SWS_J1939Tp_00112 |
| SRS_BSW_00373 | The main processing function of each AUTOSAR Basic Software Module shall be named according the defined convention | SWS_J1939Tp_00104 |
| SRS_BSW_00375 | Basic Software Modules shall report wake-up reasons | SWS_J1939Tp_99999 |
| SRS_BSW_00377 | A Basic Software Module can return a module specific types | SWS_J1939Tp_99999 |
| SRS_BSW_00385 | List possible error notifications | SWS_J1939Tp_00115 |
| SRS_BSW_00400 | Parameter shall be selected from multiple sets of parameters after code has been loaded and started | SWS_J1939Tp_00187 |
| SRS_BSW_00405 | BSW Modules shall support multiple configuration sets | SWS_J1939Tp_00187 |
| SRS_BSW_00406 | A static status variable denoting if a BSW module is initialized shall be initialized with value 0 before any APIs of the BSW module is called | SWS_J1939Tp_00023 |
| SRS_BSW_00407 | Each BSW module shall provide a function to read out the version information of a dedicated module implementation | SWS_J1939Tp_00025, SWS_J1939Tp_00089 |
| SRS_BSW_00413 | An index-based accessing of the instances of BSW modules shall be done | SWS_J1939Tp_99999 |
| SRS_BSW_00414 | Init functions shall have a pointer to a configuration structure as single parameter | SWS_J1939Tp_00187 |
| SRS_BSW_00415 | Interfaces which are provided exclusively for one module shall be separated into a dedicated header file | SWS_J1939Tp_99999 |
| SRS_BSW_00416 | The sequence of modules to be initialized shall be configurable | SWS_J1939Tp_99999 |
| SRS_BSW_00417 | Software which is not part of the SW-C shall report error events only after the DEM is fully operational. | SWS_J1939Tp_99999 |
| SRS_BSW_00419 | If a pre-compile time configuration parameter is implemented as "const" it should be placed into a separate c-file | SWS_J1939Tp_99999 |
| SRS_BSW_00423 | BSW modules with AUTOSAR interfaces | SWS_J1939Tp_99999 |

| | | |
|---|---|---|
| | shall be describable with the means of the SW-C Template | |
| SRS_BSW_00427 | ISR functions shall be defined and documented in the BSW module description template | SWS_J1939Tp_99999 |
| SRS_BSW_00433 | Main processing functions are only allowed to be called from task bodies provided by the BSW Scheduler | SWS_J1939Tp_99999 |
| SRS_BSW_00437 | Memory mapping shall provide the possibility to define RAM segments which are not to be initialized during startup | SWS_J1939Tp_99999 |
| SRS_BSW_00438 | Configuration data shall be defined in a structure | SWS_J1939Tp_00187 |
| SRS_BSW_00439 | Enable BSW modules to handle interrupts | SWS_J1939Tp_99999 |
| SRS_BSW_00440 | The callback function invocation by the BSW module shall follow the signature provided by RTE to invoke servers via Rte_Call API | SWS_J1939Tp_99999 |
| SRS_BSW_00441 | Naming convention for type, macro and function | SWS_J1939Tp_00019, SWS_J1939Tp_00115 |
| SRS_BSW_00447 | Standardizing Include file structure of BSW Modules Implementing Autosar Service | SWS_J1939Tp_99999 |
| SRS_BSW_00449 | BSW Service APIs used by Autosar Application Software shall return a Std_ReturnType | SWS_J1939Tp_99999 |
| SRS_BSW_00450 | A Main function of a un-initialized module shall return immediately | SWS_J1939Tp_00023 |
| SRS_BSW_00453 | BSW Modules shall be harmonized | SWS_J1939Tp_99999 |
| SRS_J1939_00001 | The J1939 Transport Layer module shall be configurable to support only transport protocol variant BAM | SWS_J1939Tp_00125 |
| SRS_J1939_00010 | The J1939 Transport Layer module shall implement an interface for initialization | SWS_J1939Tp_00024 |
| SRS_J1939_00011 | The J1939 Transport Layer services shall not be operational before initializing the module | SWS_J1939Tp_00023, SWS_J1939Tp_00025 |
| SRS_J1939_00019 | The J1939 Transport Layer module shall support the transport protocol variant BAM | SWS_J1939Tp_00155 |
| SRS_J1939_00020 | The AUTOSAR J1939 Transport Layer module shall support the transport protocol variant CMDT | SWS_J1939Tp_00155 |
| SRS_J1939_00022 | The timeout values of the J1939 transport protocol variants shall be supervised | SWS_J1939Tp_00018, SWS_J1939Tp_00121, SWS_J1939Tp_00123 |
| SRS_J1939_00023 | The J1939 Transport Layer module shall handle unexpected PDUs according to the SAE J1939 specification | SWS_J1939Tp_00018, SWS_J1939Tp_00064 |
| SRS_J1939_00024 | Unused Bytes in N-PDUs shall be padded | SWS_J1939Tp_00068 |
| SRS_J1939_00025 | The J1939 Transport Layer module shall be able to manage connections via BAM and CMDT in parallel | SWS_J1939Tp_00062 |

| SRS_J1939_00040 | The AUTOSAR J1939 Transport Layer module shall be based on SAE J1939 specifications | SWS_J1939Tp_00018 |
|---|---|---|
| SRS_J1939_00042 | The J1939 Transport Layer interface shall be independent of its internal configuration | SWS_J1939Tp_00157 |

# 7 Functional Specification

This chapter describes the functionality of the AUTOSAR J1939 Transport Layer. It explains the services provided to the upper and lower layers and the internal behavior of J1939Tp.

## 7.1 Basic Principles of SAE J1939

The SAE J1939 standard defines a set of Parameter Groups (PGs), each containing signals with defined content and semantics. The following information is provided for each PG:
- Payload length type: maximal number of bytes, fixed or variable size.
- Parameter Group Number (PGN): 18 bit value containing the following fields:
    - 2 bit data page information (DP and EDP)
    - 8 bit PDU-Format (PF)
    - 8 bit PDU-Specific (PS)

PGNs with PDU-Format < 240 (format 1) identify point-to-point messages, while PGNs with PDU-Format >= 240 (format 2) identify broadcast messages. The PDU-Specific field is only relevant for broadcast messages (format 2); it is always zero for point-to-point messages (format 1).

J1939 uses 29-bit CAN identifiers to identify each message. The CAN identifier contains a 3-bit-priority, the PGN, the Source Address (SA), and the Destination Address (DA, only for point-to-point messages).

Usually, an ECU has just one node address, which is used as DA in received messages and as SA in transmitted messages. However, a single ECU can also implement several different J1939 nodes at once, each with its own node address. In this case, the ECU accepts any of these node addresses as DA and sends with any of the defined node addresses as SA.

The SAE J1939 transport layer uses two special point-to-point messages identified by PGNs of format 1 to transport segmented messages, both with a fixed length of 8 bytes. These messages are called transport frames in the context of this document.

TP.CM is used for connection management. The first byte of the payload identifies its role, which may be one of the following:
TP.CM_BAM is used to initiate a BAM transfer.
TP.CM_RTS is transmitted to initiate a CMDT transfer.
TP.CM_CTS is used for flow control during a CMDT transfer.
TP.CM_EndOfMsgAck indicates the end of a CMDT transfer.
TP.Conn_Abort indicates an error and terminates the CMDT transfer.

TP.DT contains a sequence number in the first byte and 7 bytes of data.

A single TP.CM or TP.DT frame, identified by a certain CAN Identifier, is used for different PGs. The PGN of the transported PG is contained in the payload of the TP.CM frames as specified in [16].

Document ID 425: AUTOSAR_SWS_SAEJ1939TransportLayer

The destination address (DA) of CMDT related transport frames contains a legal node address and thus allows a point-to-point connection, while the DA of BAM related transport frames is always set to $FF_{16}$ to create a broadcast connection.

**[SWS_J1939Tp_00018]** ⌈
The J1939Tp module shall follow the recommendations of SAE J1939-21 [16] if they are not explicitly excluded in this document. ⌋ (SRS_J1939_00022, SRS_J1939_00023, SRS_J1939_00040)

## 7.2 Basic Functionality of J1939Tp

This section describes aspects of the functionality of J1939Tp that are not related to neighboring modules.

**[SWS_J1939Tp_00071]** ⌈
On errors and exceptions, the J1939Tp module shall not modify its current module state (see Figure 7-1: J1939 Transport Layer life cycle) but shall simply report the error event. ⌋ ( )

To assure a unique PDU router handling of all J1939 PGs which is independent of the Payload Length type (variable or fixed), the J1939Tp is used for the transmission of all Parameter Groups that are longer than 8 Bytes, independent of their length being fixed or variable.

This means that PGs with variable length that have a configured maximum size of more than 8 bytes but do not exceed 8 bytes at runtime shall be transported using J1939Tp even though no segmentation is necessary and a direct transmission from PDU router to CAN Interface would be possible.

**[SWS_J1939Tp_00155]** ⌈
The J1939 transport layer shall implement the following two J1939 transport protocol variants defined in [16]:
- BAM for broadcast transmission
- CMDT for point-to-point transmission⌋ (SRS_J1939_00019, SRS_J1939_00020)

**[SWS_J1939Tp_00125]** ⌈
The J1939 Transport Layer shall be configurable to either use both BAM and CMDT transport protocol variants, or only BAM, or only CMDT. ⌋ (SRS_BSW_00171, SRS_J1939_00001)

**[SWS_J1939Tp_00198]** ⌈
The layout and content of MetaData of N-SDUs and N-PDUs used by the J1939Tp is determined by MetaDataLength:

| MetaDataLength | Content of MetaData |
|---|---|
| 0 | [no meta data] |
| 1 | SA |

2      SA, DA

4      SA, DA, [don't care], Priority << 2 [lower 2 bits ignored]

Direct N-PDUs will contain the PGN in the ignored parts and instead of the DA in case of format 2 PGNs. ⌋ ( )

**[SWS_J1939Tp_00231]** ⌈
The J1939 Transport Layer module shall support several connections for the same PGN in parallel as long as they can be received or transmitted independently. ⌋ ( )

For the definition of connections see section 7.5.7.

## 7.3  Initialization and Shutdown

The following figure summarizes all the requirements concerning initialization and shut down:
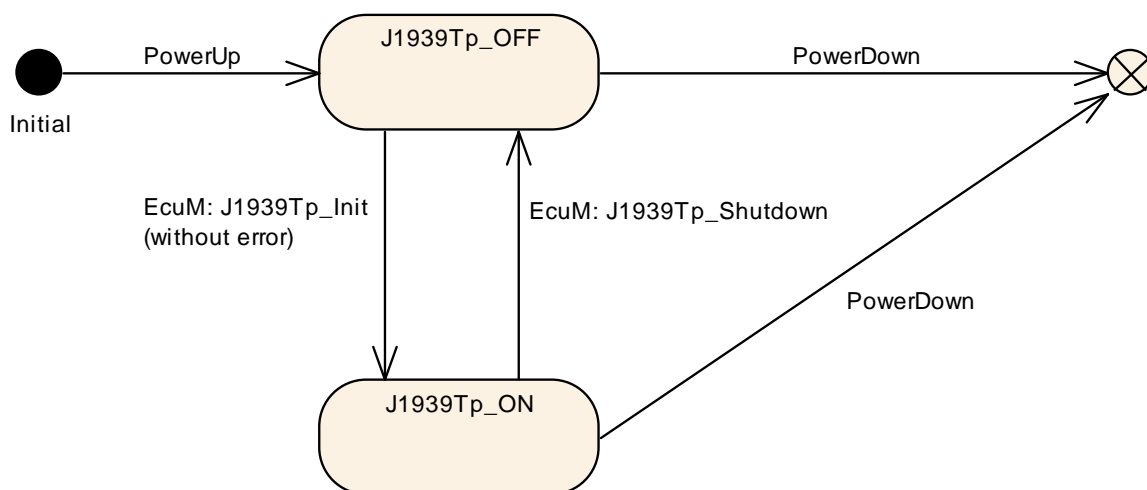


**Figure 7-1: J1939 Transport Layer life cycle**

**[SWS_J1939Tp_00019]** ⌈
The J1939Tp module shall have two global states, `J1939TP_ON` and `J1939TP_OFF`. ⌋ (SRS_BSW_00335, SRS_BSW_00441)

**[SWS_J1939Tp_00020]** ⌈
The J1939Tp module shall be in the `J1939TP_OFF` state after power up. ⌋ ( )

In the state `J1939TP_OFF,` the J1939Tp allows for an update of the post build configuration.

**[SWS_J1939Tp_00023]** ⌈

The J1939Tp module shall perform segmentation and reassembly tasks only when the J1939Tp is in the `J1939TP_ON` state. ⌋ (SRS_BSW_00406, SRS_BSW_00450, SRS_J1939_00011)

**[SWS_J1939Tp_00076]** ⌈
The global state (see **SWS_J1939Tp_00019**) shall be used to check if the module has been initialized before calling an API. ⌋ ()

**[SWS_J1939Tp_00025]** ⌈
If default error detection (DET) for the J1939Tp module is enabled, the J1939Tp module shall raise an error (J1939TP_E_UNINIT) when any function excepting J1939Tp_GetVersionInfo is called before the function `J1939Tp_Init()` has been called. ⌋ (SRS_BSW_00407, SRS_J1939_00011)

## 7.4 Communication with the Lower Layer

**[SWS_J1939Tp_00157]** ⌈
All service interfaces provided to the lower layer shall be independent of the internal communication configuration and implementation of the J1939Tp. ⌋ (SRS_J1939_00042)

**[SWS_J1939Tp_00041]** ⌈
J1939Tp shall send the transport frames TP.CM and TP.DT and direct frames using the service function `CanIf_Transmit()`.⌋ ()

### 7.4.1 Transmission Confirmation

The transmission confirmation `J1939Tp_TxConfirmation()` is called by CanIf to notify J1939 TP of successful transmission of an N-PDU.

**[SWS_J1939Tp_00035]** ⌈
When no `J1939Tp_TxConfirmation()` is received within J1939TpTxConfTimeout (see **ECUC_J1939Tp_00006 :** ) seconds after a call to `CanIf_Transmit()`, the J1939Tp module shall abort the corresponding session, and any late confirmation shall be ignored. ⌋ ()

For transmitting sessions, the session abort is defined in **SWS_J1939Tp_00032**, while **SWS_J1939Tp_00031** defines the behavior in case of a receiving session. The J1939Tp shall not try to send an abort frame (as defined by **SWS_J1939Tp_00097**) in this case.

### 7.4.2 Reception Indication

The J1939Tp module shall provide a J1939Tp_RxIndication() API to allow the CanIf to notify that a new N-PDU has been received (see **SWS_J1939Tp_00108**).
CanIf shall perform the Reception Indication according to its configuration (i.e. in ISR context if configured so).

## 7.5 Internal Behavior

### 7.5.1 Session Handling

#### 7.5.1.1 Close Transmission

**[SWS_J1939Tp_00119]** ⌈
When the transport transmission session is successfully completed, the J1939Tp module shall call a notification service of the upper layer, `PduR_J1939TpTxConfirmation`, with the result E_OK, to notify that the N-SDU transfer is successfully processed.⌋ ()

#### 7.5.1.2 Abort Transmission

**[SWS_J1939Tp_00032]** ⌈
An Abort Transmission feature shall indicate the upper layer that the transmission of a given N-SDU has been aborted. It uses the callback function `PduR_J1939TpTxConfirmation` with E_NOT_OK.⌋ ()

#### 7.5.1.3 Close Reception

**[SWS_J1939Tp_00118]** ⌈
A close connection feature shall indicate the upper layer that the transport reception session is completed. It uses the `PduR_J1939TpRxIndication()` with E_OK to indicate that the N-SDU reception is successfully processed.⌋ ()

#### 7.5.1.4 Abort Reception

**[SWS_J1939Tp_00031]** ⌈
An Abort Reception feature shall indicate the upper layer that the reception of a given N-SDU has been aborted. It uses the `PduR_J1939TpRxIndication()` with state E_NOT_OK.⌋ ()

#### 7.5.1.5 Abort CMDT Connection

**[SWS_J1939Tp_00097]** ⌈

An Abort CMDT Connection feature shall indicate to the other ECU participating in the concerned connection that the current CMDT session (transmission or reception) cannot be completed successfully. A TP.Conn_Abort abort message shall be transmitted to the other ECU via CAN as specified by [16]. ⌋ ( )

Sending a TP.Conn_Abort is necessary every time an error happens after TP.CM_CTS has been successfully sent or received, and is advisable already after transmission or reception of TP.CM_RTS. The connection abort reason of the TP.Conn_Abort shall be set according to [16].

### 7.5.2 N-SDU Reception

Reception of an N-SDU is always initiated by the reception of a TP.CM message. In case of CMDT, this is a TP.CM_RTS, in case of BAM a TP.CM_BAM.

**[SWS_J1939Tp_00043]** ⌈
Depending on the control byte of the initializing TP.CM frame (BAM or RTS), the J1939Tp module shall use the variant BAM or CMDT of the J1939 transport protocol to handle the data reception. ⌋ ( )

**[SWS_J1939Tp_00038]** ⌈
When receiving an N-PDU containing a TP.CM_BAM or a TP.CM_RTS the J1939Tp module shall first notify the upper layer (PDU Router) before processing the frame reassembly. It uses the `PduR_J1939TpStartOfReception` function with the following parameters:
- the Identifier of the corresponding N-SDU,
- depending on MetaDataLength: the SA, DA and Priority,
- the total Data Length (after reassembly) and
- a pointer to a location where the upper layer stores its currently available buffer size. ⌋ ( )

**[SWS_J1939Tp_00162]** ⌈
After the reception of an N-PDU containing a TP.DT frame, the function `PduR_J1939TpCopyRxData` shall be called with the following parameters:
- the Identifier of the corresponding N-SDU,
- PduInfoPtr with max. 7 bytes of data,
- a pointer to a location where the upper layer stores its currently available buffer size. ⌋ ( )

**[SWS_J1939Tp_00173]** ⌈
The J1939Tp module shall abort the reception silently if any of the following conditions occurs:
- The `PduR_J1939TpStartOfReception` function returns BUFREQ_E_NOT_OK or BUFREQ_E_OVFL or
- the protocol chosen in **SWS_J1939Tp_00043** does not match the transport protocol configured for the transported PGN (see **ECUC_J1939Tp_00029 :** ) or

- the SA/DA of N-PDUs with MetaData do not match the configured SA/DA (see **ECUC_J1939Tp_00179 :** /**ECUC_J1939Tp_00178 :** ).

In case of a BAM connection, no further activity is required.

In case of a CMDT connection, a CMDT Connection Abort shall be performed as described in **SWS_J1939Tp_00097**, and the connection abort reason shall be set to 1 (Already in one or more connection managed sessions and cannot support another). ⌋ ( )

**[SWS_J1939Tp_00040]** ⌈
The J1939Tp module shall abort the reception like indicated in **SWS_J1939Tp_00031** if any of the following conditions occurs:
- The value returned by `PduR_J1939TpStartOfReception` via bufferSizePtr is smaller than the total data length of the N-SDU when received via the direct N-PDU,
- the `PduR_J1939TpCopyRxData` function returns `BUFREQ_E_NOT_OK`, or
- the `J1939Tp_CancelReceive` function is called.

In case of a BAM connection, no further activity is required.

In case of a CMDT connection, a CMDT Connection Abort shall be performed as described in **SWS_J1939Tp_00097**, and the connection abort reason shall be set to 1 (Already in one or more connection managed sessions and cannot support another) after the call to `PduR_J1939TpStartOfReception`, and to 2 (System resources were needed for another task so this connection managed session was terminated) in the other two cases. ⌋ ( )

### 7.5.3 N-SDU Transmission

As described in section 7.3.2, the upper layer (PDU Router) asks for the transmission of an N-SDU by calling `J1939Tp_Transmit()`. The parameters of `J1939Tp_Transmit()` describe the Identifier of the N-SDU (NSduId) and a reference to a PduInfoType that indicates the full length of the N-SDU to transmit (full Tx N-SDU data length) and a pointer to the payload N-SDU, which may contain MetaData with SA, DA and Priority depending on the MetaDataLength of the N-SDU.

**[SWS_J1939Tp_00039]** ⌈
When configured, the transport protocol variant (BAM/CMDT, see **ECUC_J1939Tp_00137 :** ) and the SA/DA (see **ECUC_J1939Tp_00181 :** / **ECUC_J1939Tp_00180 :** ) shall be used for transmission. For N-SDUs with MetaInfo, these parameters are optional. If SA or DA is not configured, the value provided via the MetaData shall be used. If the transport protocol is not configured, it shall be chosen based on the actual DA: BAM when DA is 0xFF, CMDT otherwise. ⌋ ( )

**[SWS_J1939Tp_00045]** ⌈
The function J1939Tp_Transmit shall use the NSduId and the SduLength provided in the PduInfoType structure. It shall not use the payload of N-SDUs, only the contained MetaData. ⌋ ( )

**[SWS_J1939Tp_00047]** ⌈

After a transmission request from the upper layer, the J1939Tp module shall initiate the transmission by sending:
- For CMDT: a TP.CM_RTS frame
- For BAM: a TP.CM_BAM frame ⌋ ()

**[SWS_J1939Tp_00046]** ⌈

For each TP.DT frame to be sent, the J1939Tp module shall previously call `PduR_J1939TpCopyTxData` with the following parameters:
- the Identifier of the corresponding N-SDU,
- PduInfoType structure with up to 7 bytes as SduLength,
- the retry parameter, and
- a pointer to a location where the upper layer stores its currently available data.

  ⌋ ()

**[SWS_J1939Tp_00228]** ⌈

When `PduR_J1939TpCopyTxData` returns `BUFREQ_E_BUSY`, the J1939Tp shall retry the call to `PduR_J1939TpCopyTxData` until the data is available or a timeout occurs. ⌋ ()

**[SWS_J1939Tp_00218]** ⌈

If J1939TpTxRetrySupport is disabled, the parameter retry of `PduR_J1939TpCopyTxData` shall allways be set to the NULL_PTR. ⌋ ()

**[SWS_J1939Tp_00219]** ⌈

For BAM transmissions, the parameter retry of `PduR_J1939TpCopyTxData` shall allways be set to the NULL_PTR. ⌋ ()

**[SWS_J1939Tp_00220]** ⌈

If J1939TpTxRetrySupport is enabled, a valid RetryInfoType stuct shall be provided via the parameter retry of `PduR_J1939TpCopyTxData` during CMDT transmissions. ⌋ ()

See section 7.5.4.2 for a description how the J1939 Transport Layer module uses the RetryInfoType during CMDT transmission.

**[SWS_J1939Tp_00048]** ⌈

The J1939Tp module shall abort the transmission session like specified in **SWS_J1939Tp_00032** if any of the following conditions occur:
- The upper Layer returns the `PduR_J1939TpCopyTxData` function call with `BUFREQ_E_NOT_OK` or
- the `J1939Tp_CancelTransmit` function is called.

In case of a CMDT connection, a CMDT Connection Abort shall be performed as described in **SWS_J1939Tp_00097**, and the connection abort reason shall be set to

2 (System resources were needed for another task so this connection managed session was terminated). ⌋ ( )

### 7.5.4  Data Flow on the CAN Bus

#### 7.5.4.1  Data Flow using Direct Transmission

The following figure shows an example of direct message transmission between two ECUs using J1939Tp. This is the only case of transmission of a J1939Tp N-SDU using no TP.CM or TP.DT frame. The SA is always included in the CAN identifier. Depending on the PDU-Format of the concerned PG, the CAN Identifier might contain the DA.
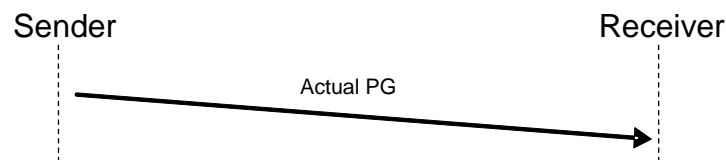
**Figure 7-2: Example of direct data flow for PGs of variable length <= 8 Bytes**

#### 7.5.4.2  Data Flow using CMDT

The following figure shows an example of segmented message transmission between two ECUs using CMDT as transport protocol variant. The CMDT transport protocol variant is used for peer-to-peer communication (i.e. 1 to 1 communication, like physical addressing in diagnostics). In the example, the transmitted PG has a total length of 16 bytes, which corresponds to 3 blocks of 7 bytes.
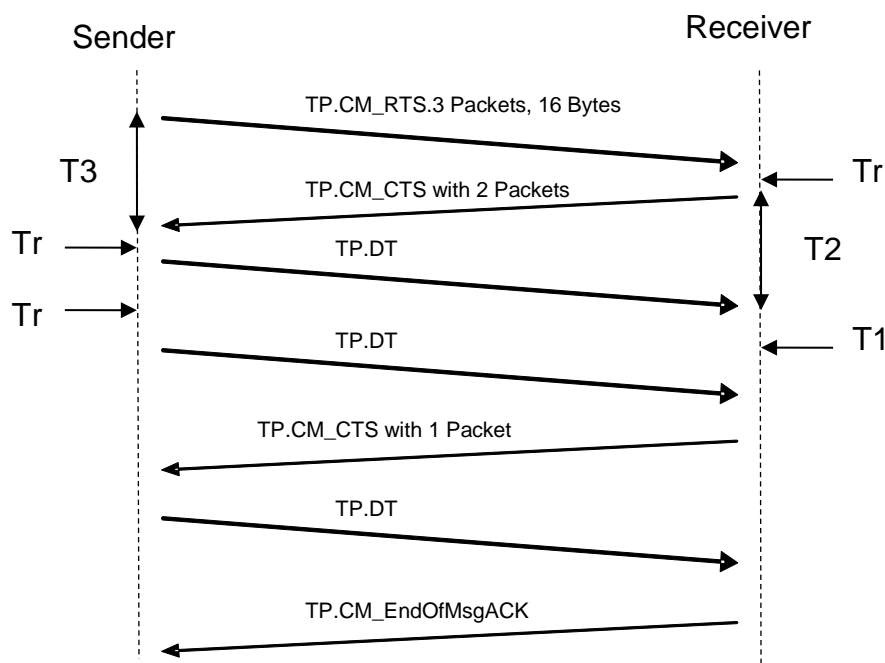
**Figure 7-3: Example of data flow without error using CMDT as transport protocol variant**

Document ID 425: AUTOSAR_SWS_SAEJ1939TransportLayer

The J1939 transport protocol uses the initial sequence of RTS and CTS to determine the number of packets per block.

**[SWS_J1939Tp_00165]** ⌈
If J1939TpTxDynamicBlockCalculation is enabled, after J1939Tp_Transmit has been called, the J1939Tp shall call `PduR_J1939TpCopyTxData` once with info->SduLength set to 0 and retry set to NULL_PTR to obtain the available amount of data via availableDataPtr. ⌋ ( )

**[SWS_J1939Tp_00207]** ⌈
If J1939TpTxDynamicBlockCalculation and J1939TpTxMaxPacketsPerBlock are enabled, the J1939Tp shall compare the available amount of data returned by `PduR_J1939TpCopyTxData` to J1939TpTxMaxPacketsPerBlock and use use the smaller of these two values to calculate the maximum number of packets field of the TP.CM_RTS message. ⌋ ( )

**[SWS_J1939Tp_00208]** ⌈
If J1939TpTxDynamicBlockCalculation is enabled but J1939TpTxMaxPacketsPerBlock is disabled, the J1939Tp shall use the available amount of data returned by `PduR_J1939TpCopyTxData` to calculate the maximum number of packets field of the TP.CM_RTS message. ⌋ ( )

**[SWS_J1939Tp_00209]** ⌈
If J1939TpTxDynamicBlockCalculation is disabled, the J1939Tp shall use J1939TpTxMaxPacketsPerBlock for the maximum number of packets field of the TP.CM_RTS message. ⌋ ( )

**[SWS_J1939Tp_00210]** ⌈
If J1939TpRxDynamicBlockCalculation and J1939TpRxPacketsPerBlock are enabled, the J1939Tp shall compare the value returned by `PduR_J1939TpStartOfReception` for the available buffer size to J1939TpRxPacketsPerBlock and use the lower value to calculate the number of packets field of the TP.CM_CTS message. ⌋ ( )

**[SWS_J1939Tp_00211]** ⌈
If J1939TpRxDynamicBlockCalculation is enabled but J1939TpRxPacketsPerBlock is disabled, the J1939Tp shall use the value returned by `PduR_J1939TpStartOfReception` for the available buffer size to calculate the number of packets field of the TP.CM_CTS message. ⌋ ( )

**[SWS_J1939Tp_00226]** ⌈
After reception of the TP.CM_RTS and after reception of the last N-PDU of a block, if the reported buffer size is large enough for the next block, the J1939Tp shall transmit a TP.CM_CTS message requesting the next block. The number of packets requested

by TP.CM_CTS shall be constant during the complete reception of one N-SDU, only in the last TP.CM_CTS this number shall be reduced to the number of remaining packets. ⌋ ( )

**[SWS_J1939Tp_00227]** ⌈
When there is not enough buffer reported by `PduR_J1939TpStartOfReception` or `PduR_J1939TpCopyRxData` for the reception of a complete block, the J1939Tp shall call `PduR_J1939TpCopyRxData` with `info->SduLength` set to 0 until the buffer is large enough for one block, or a timeout occurs. ⌋ ( )

**[SWS_J1939Tp_00229]** ⌈
While monitoring the buffer state as defined by **SWS_J1939Tp_00227**, the J1939Tp shall send TP.CM_CTS wait frames (number of packets set to 0, see also [16]) according to the timing requirements defined in [16]. ⌋ ( )

**[SWS_J1939Tp_00212]** ⌈
If J1939TpRxRetrySupport is enabled, the J1939Tp shall adapt the value returned by `PduR_J1939TpStartOfReception` for the available buffer size according to J1939TpRxDynamicBufferRatio before using it to calculate the number of packets field of the TP.CM_CTS message as specified in **SWS_J1939Tp_00210** and **SWS_J1939Tp_00211**. ⌋ ( )

**[SWS_J1939Tp_00213]** ⌈
If J1939TpRxDynamicBlockCalculation is disabled, the J1939Tp shall use J1939TpRxPacketsPerBlock to calculate the number of packets field of the TP.CM_CTS message. ⌋ ( )

**[SWS_J1939Tp_00217]** ⌈
If J1939TpTxRetrySupport is enabled, the J1939Tp shall call `PduR_J1939TpCopyTxData` with
  - TpDataState set to TP_DATACONF for the first call after reception of a TP.CM_CTS, and
  - TpDataState set to TP_CONFPENDING for the following calls. ⌋ ( )

**[SWS_J1939Tp_00195]** ⌈
If a TP.CM_CTS wait frame (number of packets set to 0, see also [16]) is received, the J1939Tp shall wait for another TP.CM_CTS frame. ⌋ ( )

**[SWS_J1939Tp_00223]** ⌈
When the J1939Tp receives a TP.CM_CTS frame that requests data beyond the current position or preceding the position where the last TP.CM_CTS was received, it shall abort the transmission using the mechanisms described by **SWS_J1939Tp_00032** and **SWS_J1939Tp_00097** with reason $FF_{16}$ (SNA). ⌋ ( )

Document ID 425: AUTOSAR_SWS_SAEJ1939TransportLayer

**[SWS_J1939Tp_00221]** ⌈
If J1939TpTxRetrySupport is enabled, when the J1939Tp receives a TP.CM_CTS frame requesting already sent data, it shall call `PduR_J1939TpCopyTxData` with TpDataState set to TP_DATARETRY and TxTpDataCnt set to the number of bytes to be retransmitted. ⌋ ( )

The number of bytes that need to be retransmitted is calculated from the position of the requested package relative to the current package.

**[SWS_J1939Tp_00194]** ⌈
If J1939TpTxRetrySupport is disabled, when the J1939Tp receives a TP.CM_CTS frame requesting already sent data, it shall abort the transmission using the mechanisms described by **SWS_J1939Tp_00032** and **SWS_J1939Tp_00097** with reason 255 (unassigned). ⌋ ( )

**[SWS_J1939Tp_00222]** ⌈
If J1939TpRxRetrySupport is enabled, when a sequence error occurs during reception, the J1939Tp module shall, after the last TP.DT message of the block was received, send a TP.CM_CTS frame requesting the packages that follow the last correctly received package of the current block. ⌋ ( )

**[SWS_J1939Tp_00232]** ⌈
The content of a TP.DT message with a sequence error and of the TP.DT messages following a sequence error in the same block shall be discarded. ⌋ ( )

Note: This means that for such messages PduR_J1939TpCopyRxData shall not be called.

**[SWS_J1939Tp_00216]** ⌈
If J1939TpRxRetrySupport is disabled, when a sequence error occurs during reception, the J1939Tp module shall abort the reception session
- as specified in **SWS_J1939Tp_00031** and
- as specified in **SWS_J1939Tp_00097** with connection abort reason $FF_{16}$ (SNA). ⌋ ( )

**[SWS_J1939Tp_00123]** ⌈
J1939Tp shall implement all CMDT related timing constraints (Tr, Th, T1, T2, T3, T4) as described in [16]. They supervise the CMDT data flow. ⌋ (SRS_J1939_00022)

**[SWS_J1939Tp_00100]** ⌈
If a timeout occurs during CMDT transmission (see [16] for details) then the J1939Tp module shall abort the transmission session
- as specified in **SWS_J1939Tp_00032** and
- as specified in **SWS_J1939Tp_00097** with connection abort reason 3 (A timeout occurred and this is the connection abort to close the session) when

the timeout occurred after successful transmission of the TP.CM_RTS frame.
⌋ ( )

**[SWS_J1939Tp_00159]** ⌈
If a timeout occurs during reception (see [16] for details) then the J1939Tp module shall abort the reception session
- as specified in **SWS_J1939Tp_00031** and
- as specified in **SWS_J1939Tp_00097** with connection abort reason 3 (A timeout occurred and this is the connection abort to close the session). ⌋ ( )

**[SWS_J1939Tp_00098]** ⌈
If a TP.Conn_Abort frame is received after a TP.CM_RTS frame has been successfully sent by the ECU, the transmission shall be aborted like indicated in **SWS_J1939Tp_00032**. No transmission of TP.Conn_Abort is necessary. ⌋ ( )

**[SWS_J1939Tp_00163]** ⌈
If a TP.Conn_Abort frame is received after a TP.CM_RTS frame has been received, the reception shall be aborted like indicated in **SWS_J1939Tp_00031**. No transmission of TP.Conn_Abort is necessary. ⌋ ( )

### 7.5.4.3  Data Flow using BAM

**[SWS_J1939Tp_00121]** ⌈
J1939Tp shall implement the BAM related timing constraints (50ms, Tr, T1) as described in [16]. They supervise the BAM data flow. ⌋ (SRS_J1939_00022)

**[SWS_J1939Tp_00160]** ⌈
If a timeout occurs during the BAM reception, the J1939Tp module shall abort the reception session as specified in **SWS_J1939Tp_00031**. ⌋ ( )

**[SWS_J1939Tp_00192]** ⌈
After a sequence error, J1939Tp shall abort BAM connections as specified in **SWS_J1939Tp_00031**. ⌋ ( )

The following figure shows an example of segmented message transmission between two ECUs using BAM as transport protocol variant according to [16]. The BAM transport protocol variant is used for a broadcast communication (i.e. 1 to n communication, like functional addressing in diagnostics). In the example, the transmitted PG has a total length of 3 blocks of 7 bytes that have to be successively transmitted.

Sender                                           Receiver

TP.CM_BAM total Byte Length= 16

50ms .. Tr

T1

TP.DT

TP.DT

TP.DT
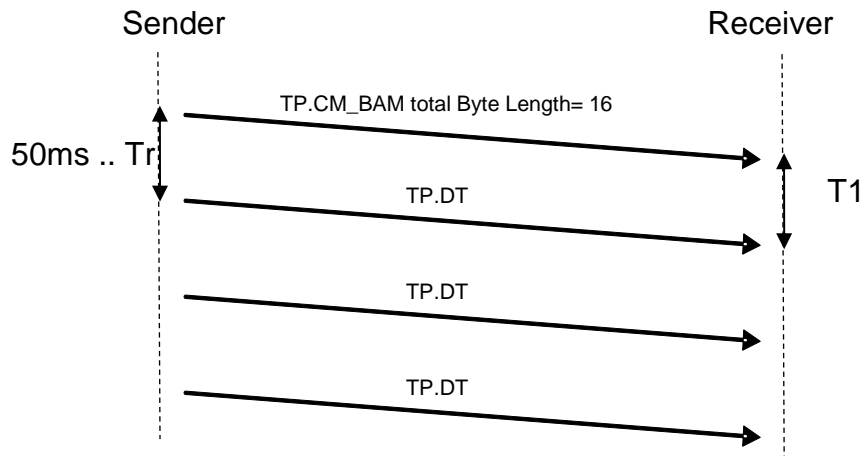
**Figure 7-4: Example of data flow using BAM as transport protocol variant**

### 7.5.5  N-SDU Buffer Management

J1939Tp shall have no internal PDU buffers. It requests data for sending directly from the upper layers via `PduR_J1939TpCopyTxData()` and provides received data directly to the upper layers via `PduR_J1939TpCopyRxData()`.

To guarantee data consistency, the complete buffer of the upper layers must be locked during the whole data transmission or reception.

Data transmission is initialized when the PduR calls J1939Tp_Transmit() and is active after J1939Tp executed this call successfully until J1939Tp calls `PduR_J1939TpTxConfirmation()`:

LOCK

J1939Tp_Transmit
=> E_OK

PduR_J1939TpTxConfirmation

UNLOCK
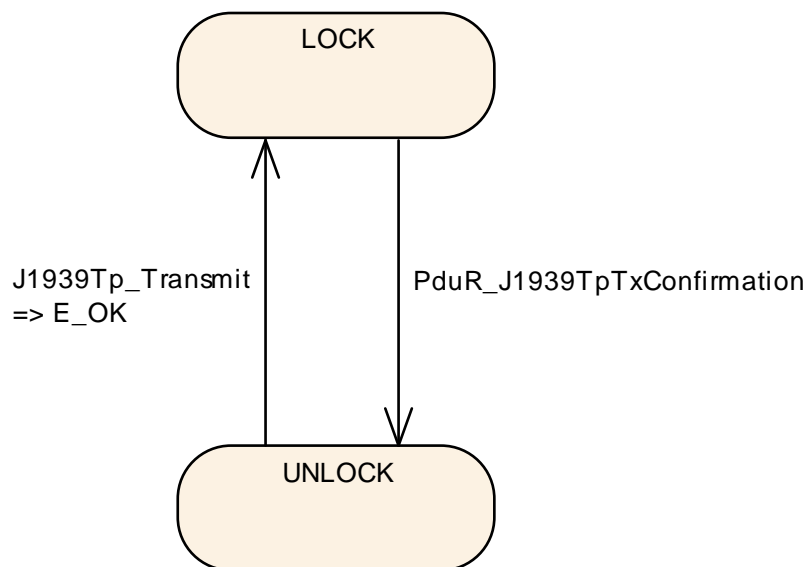
**Figure 7-5: Tx Buffer locking**

Data reception is initialized when J1939Tp calls `PduR_J1939TpStartOfReception()` and is active after the PduR executed this call successfully until J1939Tp calls `PduR_J1939TpRxIndication()`:
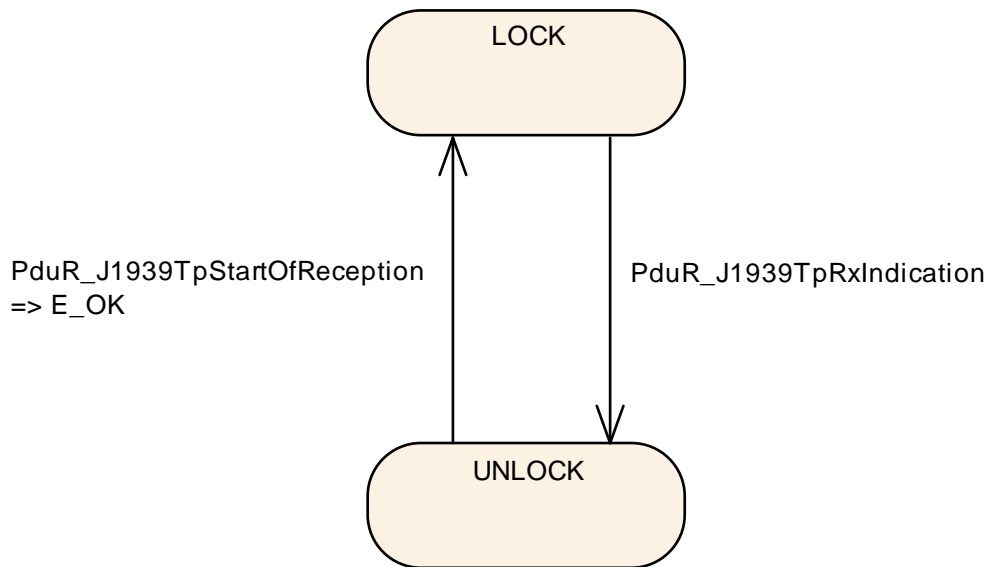


**Figure 7-6: Rx Buffer locking**

### 7.5.6 Relationship between N-SDU and N-PDU in J1939Tp

This section describes the relation that exists between an N-SDU and the set of N-PDUs that is required to transport the N-SDU data, as shown in the following figure.
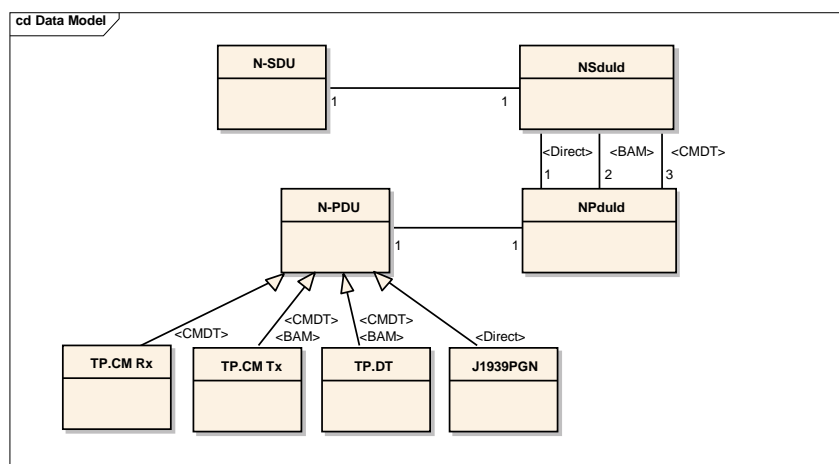


**Figure 7-7: Relation between N-SDU and N-PDU**

The N-PDUs as well as the N-SDU may use MetaData. In that case, the N-SDU stands for a certain PGN, and the N-PDUs stand for TP.CM, TP.DT, and direct PG, regardless of source and/or destination addresses.

**[SWS_J1939Tp_00057]** ⌈

During reception, the N-SDU is identified by a combination of the PGN included in the payload of TP.CM and, when the N-SDU has no MetaData, from the addressing information implicitly encoded in the NPduIds or explicitly provided via the MetaData of the N-PDUs. ⌋ ( )

**[SWS_J1939Tp_00199]** ⌈

During transmission, the relevant set of N-PDUs is identified by the configured SA/DA of N-SDUs without MetaData, or by the SA/DA explicitly provided in the MetaData by the upper layer. ⌋ ( )

### 7.5.7  Concurrent Connections

Connections only concern internal J1939Tp purposes. They are transparent for the upper and lower layers of J1939Tp but influence the handling of J1939Tp.

A J1939Tp connection is characterized by its direction (Receiving /Sending) and its type (BAM / CMDT / Direct). A J1939Tp connection of type BAM or CMDT uses the following transport related frames:
- control (TP.CM)
- data (TP.DT)

The CAN-Identifier corresponding to those transport specific frames is the same for all J1939 PGs longer than 8 bytes:
- sent from a given SA - in the BAM case
- sent from a given SA to a given DA - in the CMDT case

This reduces the possibility for J1939Tp to process concurrent connections.

**[SWS_J1939Tp_00120]** ⌈

The J1939Tp shall be able to handle connections in parallel for all N-SDUs that do not interfere in the usage of TP.DT frames with the same SA and DA. For channels with defined SA/DA and protocol type, only one TP connection and one direct connection can be open at any time. For channels with variable SA/DA, the maximum number of parallel connections is limited by the number of N-SDUs assigned to this channel. ⌋ ( )

J1939Tp shall only support concurrent connections as described in section 5.10.5 of [16]. Note that one AUTOSAR ECU can represent several J1939 nodes and thus may have more than one address (used as SA or DA) assigned to it.

**[SWS_J1939Tp_00062]** ⌈

Each connection shall be independent of the other connections. ⌋ (SRS_J1939_00025)

This means that a connection shall use its own resources, such as timer or state machine.

### 7.5.8 N-PDU Padding

**[SWS_J1939Tp_00200]** ⌈
The J1939 Transport Layer module shall send TP.DT frames always with 8 bytes according to [16]. ⌋ ( )

**[SWS_J1939Tp_00068]** ⌈
According to [16], all unused data bytes within the last TP.DT frame or the direct frame shall be set to 0xFF. ⌋ (SRS_J1939_00024)

### 7.5.9 Handling of Unexpected N-PDU Arrivals

**[SWS_J1939Tp_00064]** ⌈
The J1939Tp shall ignore unexpected N-PDUs that do not correspond to a currently active connection. ⌋ (SRS_J1939_00023)

**[SWS_J1939Tp_00224]** ⌈
If J1939TpTxRetrySupport is disabled, when a TP.CM_CTS is received while TP.DT messages are being transmitted, J1939Tp shall abort the connection as specified in **SWS_J1939Tp_00032** and in **SWS_J1939Tp_00097** with connection abort reason 4 (CTS messages received when data transfer is in progress). ⌋ ()

**[SWS_J1939Tp_00225]** ⌈
When a TP.CM_RTS is received for a currently active connection, the J1939Tp shall stop this connection as specified in **SWS_J1939Tp_00031** and start a new connection as described in **SWS_J1939Tp_00038**. ⌋ ()

## 7.6 Error Classification

The J1939 Transport Layer module supports reporting of development and runtime errors.

### 7.6.1 Development Errors

**[SWS_J1939Tp_00115]** ⌈
J1939Tp shall use the following development errors:

| Type or error | Relevance | Related error code | Value [hex] |
|---|---|---|---|
| API service used in state J1939TP_OFF. | Development | J1939TP_E_UNINIT | 0x01 |
| J1939Tp_Init() called in state J1939TP_ON. | Development | J1939TP_E_REINIT | 0x02 |
| J1939Nm_Init was called with an invalid configuration pointer | Development | J1939NM_E_INIT_FAILED | 0x03 |
| API service called with null pointer. | Development | J1939TP_E_PARAM_POINTER | 0x10 |

| API service called with wrong ID. | Development | J1939TP_E_INVALID_PDU_SDU_ID | 0x11 |
|---|---|---|---|
| Timeout occurred on receiver side after reception of an intermediate TP.DT frame of a block. | Development | J1939TP_E_TIMEOUT_T1 | 0x30 |
| Timeout occurred on receiver side after transmission of a TP.CM/CTS frame. | Development | J1939TP_E_TIMEOUT_T2 | 0x31 |
| Timeout occurred on transmitter side after transmission of the last TP.DT frame of a block. | Development | J1939TP_E_TIMEOUT_T3 | 0x32 |
| Timeout occurred on transmitter side after reception of a TP.CM/CTS(0) frame. | Development | J1939TP_E_TIMEOUT_T4 | 0x33 |
| Timeout occurred on transmitter or receiver side while trying to send the next TP.DT or TP.CM frame. | Development | J1939TP_E_TIMEOUT_TR | 0x34 |
| Timeout occurred on receiver side while trying to send the next TP.CM/CTS frame after a TP.CM/CTS(0) frame. | Development | J1939TP_E_TIMEOUT_TH | 0x35 |
| Invalid value for "total message size" in received TP.CM/RTS frame. | Development | J1939TP_E_INVALID_TMS | 0x40 |
| Value for "total number of packets" in received TP.CM/RTS frame does not match the "total message size". | Development | J1939TP_E_INVALID_TNOP | 0x41 |
| Invalid value for "maximum number of packets" in received TP.CM/RTS frame. | Development | J1939TP_E_INVALID_MNOP | 0x42 |
| Unexpected PGN in received TP.CM frame. | Development | J1939TP_E_INVALID_PGN | 0x43 |
| Invalid value for "number of packets" in received TP.CM/CTS frame. | Development | J1939TP_E_INVALID_NOP | 0x44 |
| Invalid value for "next packet number" in received TP.CM/CTS frame. | Development | J1939TP_E_INVALID_NPN | 0x45 |
| Invalid value for "connection abort reason" in received TP.Conn_Abort frame. | Development | J1939TP_E_INVALID_CAR | 0x46 |
| Unexpected serial number in received TP.DT frame. | Development | J1939TP_E_INVALID_SN | 0x47 |

⌋ (SRS_BSW_00327, SRS_BSW_00337, SRS_BSW_00385, SRS_BSW_00441)


### 7.6.2 Runtime Errors

Runtime errors have not yet been classified.


### 7.6.3 Transient Faults

There are no transient faults.


### 7.6.4 Production Errors

There are no production errors.

### 7.6.5 Extended Production Errors

There are no extended production errors.


## 7.7 API Parameter Checking

J1939TP_E_PARAM_POINTER shall be reported as specified in [15] by SWS_BSW_00212.


**[SWS_J1939Tp_00188]** ⌈
If default error detection for the J1939Tp is enabled, all APIs using a SDU- or PDU-Identifier shall check the input Identifier and raise the development error: J1939TP_E_INVALID_PDU_SDU_ID in case the API has been called for a not configured PDU or SDU. ⌋ (SRS_BSW_00323)

# 8 API Specification

## 8.1 Imported Types

This section lists all externally defined types that are used by J1939Tp. These types are included from the headers corresponding to the module names listed in the table below.

**[SWS_J1939Tp_00230]** ⌈

| Module | Imported Type |
|---|---|
| ComStack_Types | BufReq_ReturnType |
| | PduIdType |
| | PduInfoType |
| | PduLengthType |
| | RetryInfoType |
| | TPParameterType |
| Std_Types | Std_ReturnType |
| | Std_VersionInfoType |

⌋ ( )

## 8.2 Type Definitions

This section lists the types defined by J1939Tp.

**[SWS_J1939Tp_00175]** ⌈

| Name: | J1939Tp_ConfigType | |
|---|---|---|
| Type: | Structure | |
| Range: | implementation specific | The content of the initialization data structure is implementation specific. |
| Description: | Data structure containing post-build configuration data of J1939-TP. | |

⌋ ( )

The J1939Tp_ConfigType defines a structure that contains configuration parameters J1939Tp uses at run time. It is provided as an argument to J1939Tp_Init().

## 8.3 Function Definitions

This section defines a list of functions provided for upper layer modules. All these APIs shall provide the following development errors:

### 8.3.1 J1939Tp_Init

**[SWS_J1939Tp_00087]** ⌈

| Service name: | J1939Tp_Init |
|---|---|

| Syntax: | void J1939Tp_Init( |  |
|---|---|---|
|  | const J1939Tp_ConfigType* ConfigPtr |  |
|  | ) |  |
| Service ID[hex]: | 0x01 |  |
| Sync/Async: | Synchronous |  |
| Reentrancy: | Non Reentrant |  |
| Parameters (in): | ConfigPtr | Pointer to configuration data structure. |
| Parameters (inout): | None |  |
| Parameters (out): | None |  |
| Return value: | None |  |
| Description: | This function initializes the J1939Tp module. |  |

⌋ (SRS_BSW_00101, SRS_BSW_00358)

After power up, J1939Tp is in a state called J1939TP_OFF. In this state, J1939Tp is not yet configured, and therefore cannot perform any communication task. The J1939Tp module's environment (usually the ECU Manager) will call J1939Tp_Init() before using the J1939Tp module for further processing.

**[SWS_J1939Tp_00024]** ⌈
The function J1939Tp_Init() shall initialize all global variables of the module and reset all transport protocol connections. ⌋ (SRS_J1939_00010)

**[SWS_J1939Tp_00022]** ⌈
J1939Tp_Init() shall change to the internal state J1939TP_ON after successful initialization. ⌋ ( )

**[SWS_J1939Tp_00026]** ⌈
If called when the J1939Tp module is in the global state J1939TP_ON, the function J1939Tp_Init() shall raise the development error J1939TP_E_REINIT and do nothing. ⌋ ( )

**[SWS_J1939Tp_00187]** ⌈
The provided ConfigPtr shall only be used, when post-build configuration is enabled, or when different configuration variants must be supported. Otherwise, the parameters should be accessed directly to avoid indirection via the ConfigPtr. ⌋ (SRS_BSW_00400, SRS_BSW_00405, SRS_BSW_00414, SRS_BSW_00438)

The structure of type J1939Tp_ConfigType pointed to by the ConfigPtr contains post-build parameters of the J1939Tp module. In link time or pre-compile configured environments, the structure may contain a set of post-build parameters that differ between several configuration variants. It is expected that link time and pre-compile parameters may not change for different configuration variants.

J1939Tp_Init() has no return value because configuration data errors should be detected during configuration time (e.g. by the configuration tools). Furthermore, if a hardware error occurs, it will be reported via the error manager modules.

J1939TP_E_INIT_FAILED shall be reported as specified in [15] by SWS_BSW_00050.

### 8.3.2 J1939Tp_Shutdown

**[SWS_J1939Tp_00093]** ⌈

| | |
|---|---|
| *Service name:* | J1939Tp_Shutdown |
| *Syntax:* | `void J1939Tp_Shutdown(`<br>`    void`<br>`)` |
| *Service ID[hex]:* | 0x02 |
| *Sync/Async:* | Synchronous |
| *Reentrancy:* | Non Reentrant |
| *Parameters (in):* | None |
| *Parameters (inout):* | None |
| *Parameters (out):* | None |
| *Return value:* | None |
| *Description:* | This function is used to shutdown the J1939Tp module. |

⌋ (SRS_BSW_00336)

To close down communication, the state handling (usually the ECU Manager) calls `J1939Tp_Shutdown()`.

**[SWS_J1939Tp_00094]** ⌈
`J1939Tp_Shutdown()` shall close all pending transport protocol connections, free all resources and set the J1939Tp module into the global state `J1939TP_OFF` state.
⌋ ()

**[SWS_J1939Tp_00095]** ⌈
`J1939Tp_Shutdown()` shall not raise a notification about the pending frame transmission or reception. ⌋ ()

### 8.3.3 J1939Tp_GetVersionInfo

**[SWS_J1939Tp_00089]** ⌈

| | |
|---|---|
| *Service name:* | J1939Tp_GetVersionInfo |
| *Syntax:* | `void J1939Tp_GetVersionInfo(`<br>`    Std_VersionInfoType* VersionInfo`<br>`)` |
| *Service ID[hex]:* | 0x03 |
| *Sync/Async:* | Synchronous |
| *Reentrancy:* | Reentrant |
| *Parameters (in):* | None |
| *Parameters* | None |

| | | |
|---|---|---|
| *(inout):* | | |
| *Parameters (out):* | VersionInfo | Pointer to the location where the version information of J1939Tp shall be stored. |
| *Return value:* | None | |
| *Description:* | Returns the version information of J1939Tp. | |

⌋ (SRS_BSW_00407)

Note that the function J1939Tp_GetVersionInfo may be called in global state `J1939TP_OFF`, i.e. before initialization of the J1939Tp module.

### 8.3.4  J1939Tp_Transmit

**[SWS_J1939Tp_00096]** ⌈

| | | |
|---|---|---|
| *Service name:* | J1939Tp_Transmit | |
| *Syntax:* | `Std_ReturnType J1939Tp_Transmit(`<br>`    PduIdType TxSduId,`<br>`    const PduInfoType* TxInfoPtr`<br>`)` | |
| *Service ID[hex]:* | 0x05 | |
| *Sync/Async:* | Synchronous | |
| *Reentrancy:* | Reentrant | |
| *Parameters (in):* | TxSduId | ID of the J1939Tp N-SDU to be transmitted. The available IDs are configured via J1939TpTxNSduId. |
| | TxInfoPtr | Pointer to structure with length and content of the J1939Tp N-SDU that shall be transmitted. The content of this structure is only used to transfer addressing information and priority of N-SDUs with MetaData, and shall be NULL otherwise. |
| *Parameters (inout):* | None | |
| *Parameters (out):* | None | |
| *Return value:* | Std_ReturnType | E_OK: The request has been accepted.<br>E_NOT_OK: The request failed. This happens when a resource could not be allocated, e. g. when the requested transmission would use a channel that is currently active. |
| *Description:* | This service function is used to request the transfer of a J1939Tp N-SDU. | |

⌋ (SRS_BSW_00357)

As described in **SWS_J1939Tp_00119** and **SWS_J1939Tp_00032**, the J1939Tp module will notify the upper layer by calling the `PduR_J1939TpTxConfirmation` callback when the transmit request has been completed.

**[SWS_J1939Tp_00101]** ⌈
The function J1939Tp_Transmit shall reject a request, if the `J1939Tp_Transmit` service is called for an N-SDU identifier that is being used in a currently running J1939 Transport Layer session. ⌋ ( )

**[SWS_J1939Tp_00030]** ⌈
The function `J1939Tp_Transmit()` shall reject the transmit request and return the status value E_NOT_OK if the transmission needs a transport protocol and the channel is occupied (see also 7.5.7). ⌋ ( )

The term channel refers to a communication relation with identical SA and DA.

### 8.3.5 J1939Tp_CancelTransmit

**[SWS_J1939Tp_00177]** ⌈

| | | |
|---|---|---|
| *Service name:* | J1939Tp_CancelTransmit | |
| *Syntax:* | `Std_ReturnType J1939Tp_CancelTransmit(`<br>`    PduIdType TxSduId`<br>`)` | |
| *Service ID[hex]:* | 0x09 | |
| *Sync/Async:* | Synchronous | |
| *Reentrancy:* | Non Reentrant | |
| *Parameters (in):* | TxSduId | ID of the J1939Tp N-SDU to be canceled. The available IDs are configured via J1939TpTxNSduId. |
| *Parameters (inout):* | None | |
| *Parameters (out):* | None | |
| *Return value:* | Std_ReturnType | E_OK: The request has been accepted.<br>E_NOT_OK: The request failed. This happens when the provided N-SDU is currently not transmitted. |
| *Description:* | This service function is used to cancel the ongoing transmission of a J1939Tp N-SDU. | |

⌋()

**[SWS_J1939Tp_00203]** ⌈
`J1939Tp_CancelTransmit` shall return E_NOT_OK if the TxSduId is invalid, if TxSduId is currently not active, if currently a direct frame is transmitted, if the last TP.DT frame has already been transmitted during BAM transmission, or if the TP.CM_EOMAck frame has already been received during CMDT transmission.⌋()

**[SWS_J1939Tp_00214]** ⌈
The `J1939Tp_CancelTransmit` API shall only be available when J1939TpCancellationSupport is enabled. ⌋()

### 8.3.6 J1939Tp_CancelReceive

**[SWS_J1939Tp_00176]** ⌈

| | | |
|---|---|---|
| *Service name:* | J1939Tp_CancelReceive | |
| *Syntax:* | `Std_ReturnType J1939Tp_CancelReceive(`<br>`    PduIdType RxSduId`<br>`)` | |
| *Service ID[hex]:* | 0x0a | |
| *Sync/Async:* | Synchronous | |
| *Reentrancy:* | Non Reentrant | |
| *Parameters (in):* | RxSduId | ID of the J1939Tp N-SDU to be canceled. The available IDs are configured via J1939TpRxNSduId. |
| *Parameters (inout):* | None | |

| Parameters (out): | None | |
|---|---|---|
| Return value: | Std_ReturnType | E_OK: The request has been accepted.<br>E_NOT_OK: The request failed. This happens when the provided I-PDU does not refer to an N-SDU that is currently received. |
| Description: | This service function is used to cancel the ongoing reception of a J1939Tp N-SDU. | |

⌋()

### [SWS_J1939Tp_00204] ⌈

`J1939Tp_CancelReceive` shall return E_NOT_OK if the RxSduId is invalid, if RxSduId is currently not active, if currently a direct frame is received, if the last TP.DT frame has already been received during BAM reception, or if the TP.CM_EOMAck frame has already been sent during CMDT reception. ⌋()

### [SWS_J1939Tp_00215] ⌈

The `J1939Tp_CancelReceive` API shall only be available when J1939TpCancellationSupport is enabled. ⌋()

## 8.3.7  J1939Tp_ChangeParameter

### [SWS_J1939Tp_00180] ⌈

| Service name: | J1939Tp_ChangeParameter | |
|---|---|---|
| Syntax: | `Std_ReturnType J1939Tp_ChangeParameter(`<br>`    PduIdType RxSduId,`<br>`    TPParameterType Parameter,`<br>`    uint16 Value`<br>`)` | |
| Service ID[hex]: | 0x08 | |
| Sync/Async: | Synchronous | |
| Reentrancy: | Non Reentrant | |
| Parameters (in): | RxSduId | ID of the N-SDU for which parameters should be changed. The available IDs are configured via J1939TpRxNSduId. |
| | Parameter | ID of parameter that should be changed. |
| | Value | New value for changed parameter. |
| Parameters (inout): | None | |
| Parameters (out): | None | |
| Return value: | Std_ReturnType | E_OK: The request has been accepted.<br>E_NOT_OK: The request failed. This happens when the provided parameter does not exist. |
| Description: | This service function is used to change reception parameters of J1939Tp for a specific N-SDU. | |

⌋()

### [SWS_J1939Tp_00206] ⌈

`J1939Tp_ChangeParameter` shall return E_NOT_OK if the SduId is invalid, if the Parameter is not TP_BS, or if the Value is larger than 255. ⌋()

## 8.4 Callback Notifications

This is a list of functions provided for other modules. The function prototypes of the callback functions shall be provided in the file J1939Tp_Cbk.h.

### 8.4.1 J1939Tp_RxIndication

**[SWS_J1939Tp_00108]** ⌈

| | |
|---|---|
| *Service name:* | J1939Tp_RxIndication |
| *Syntax:* | ```void J1939Tp_RxIndication(     PduIdType RxPduId,     const PduInfoType* PduInfoPtr )``` |
| *Service ID[hex]:* | 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. |
| *Parameters (inout):* | None |
| *Parameters (out):* | None |
| *Return value:* | None |
| *Description:* | Indication of a received I-PDU from a lower layer communication interface module. |

⌋ (SRS_BSW_00359, SRS_BSW_00360)

The J1939Tp module provides the J1939Tp_RxIndication() API to allow the CanIf to notify that a new N-PDU has been received.

**[SWS_J1939Tp_00110]** ⌈
The function J1939Tp_RxIndication shall be callable in interrupt context (it could be called from the CAN receive interrupt). ⌋ (SRS_BSW_00333)

### 8.4.2 J1939Tp_TxConfirmation

**[SWS_J1939Tp_00112]** ⌈

| | |
|---|---|
| *Service name:* | J1939Tp_TxConfirmation |
| *Syntax:* | ```void J1939Tp_TxConfirmation(     PduIdType TxPduId )``` |
| *Service ID[hex]:* | 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. |
| *Parameters (inout):* | None |
| *Parameters (out):* | None |
| *Return value:* | None |
| *Description:* | The lower layer communication interface module confirms the transmission of an I-PDU. |

⌋ (SRS_BSW_00359, SRS_BSW_00360)

The J1939Tp module implements the J1939Tp_TxConfirmation API to allow the CanIf module to confirm that a TP related frame (TP.CM, TP.DT) or direct frame has been successfully transmitted to the J1939 network.

**[SWS_J1939Tp_00114]** ⌈
The function J1939Tp_TxConfirmation shall be callable in interrupt context (it could be called from the CAN transmit interrupt). ⌋ (SRS_BSW_00333)

## 8.5  Scheduled Functions

The Basic Software Scheduler directly calls the functions listed in this section. Scheduled functions shall have no return value and no parameter, and need not be reentrant.

### 8.5.1  J1939Tp_MainFunction

**[SWS_J1939Tp_00104]** ⌈

| Service name: | J1939Tp_MainFunction |
|---|---|
| Syntax: | `void J1939Tp_MainFunction(`<br>`    void`<br>`)` |
| Service ID[hex]: | 0x04 |
| Description: | Main function of the J1939Tp. Used for scheduling purposes and timeout supervision. |

⌋ (SRS_BSW_00373)

**[SWS_J1939Tp_00106]** ⌈
The calling frequency of the function J1939Tp_MainFunction is determined by the parameter J1939TpMainFunctionPeriod (see **ECUC_J1939Tp_00044 :** ). ⌋ ( )

## 8.6  Expected Interfaces

In this section, all interfaces required from other modules are listed.

### 8.6.1  Mandatory Interfaces

This section defines all interfaces, which are required to fulfill the core functionality of the module.

**[SWS_J1939Tp_00116]** ⌈

| API function | Description |
|---|---|
| CanIf_Transmit | This service initiates a request for transmission of the CAN L-PDU specified by the CanTxSduId and CAN related data in the L-SDU |

| | |
|---|---|
| | structure. |
| PduR_J1939TpCopyRxData | This function is called to provide the received data of an I-PDU segment (N-PDU) to the upper layer.<br>Each call to this function provides the next part of the I-PDU data.<br>The size of the remaining data is written to the position indicated by bufferSizePtr. |
| PduR_J1939TpCopyTxData | This function is called to acquire the transmit data of an I-PDU segment (N-PDU).<br>Each call to this function provides the next part of the I-PDU data unless retry->TpDataState is TP_DATARETRY. In this case the function restarts to copy the data beginning at the offset from the current position indicated by retry->TxTpDataCnt. The size of the remaining data is written to the position indicated by availableDataPtr. |
| PduR_J1939TpRxIndication | Called after an I-PDU has been received via the TP API, the result indicates whether the transmission was successful or not. |
| PduR_J1939TpStartOfReception | This function is called at the start of receiving an N-SDU. The N-SDU might be fragmented into multiple N-PDUs (FF with one or more following CFs) or might consist of a single N-PDU (SF). |
| PduR_J1939TpTxConfirmation | This function is called after the I-PDU has been transmitted on its network, the result indicates whether the transmission was successful or not. |

⌋ ( )


### 8.6.2 Optional Interfaces

This section defines all interfaces that are required to fulfill an optional functionality of the module.


**[SWS_J1939Tp_00060]** ⌈

| *API function* | *Description* |
|---|---|
| Det_ReportError | Service to report development errors. |

⌋ ( )

# 9 Sequence Diagrams

The following sequence diagrams shall give an impression of the way the J1939 Transport Layer module shall behave and interoperate with other BSW modules. They are not complete and not binding for the implementation.

## 9.1 Reception of Direct PG

The following diagram shows the interaction of the J1939 Transport Layer module with the CAN Interface and the PDU Router during reception of a direct PG, i.e. an N-SDU with dynamic length that is not larger than 8 bytes.
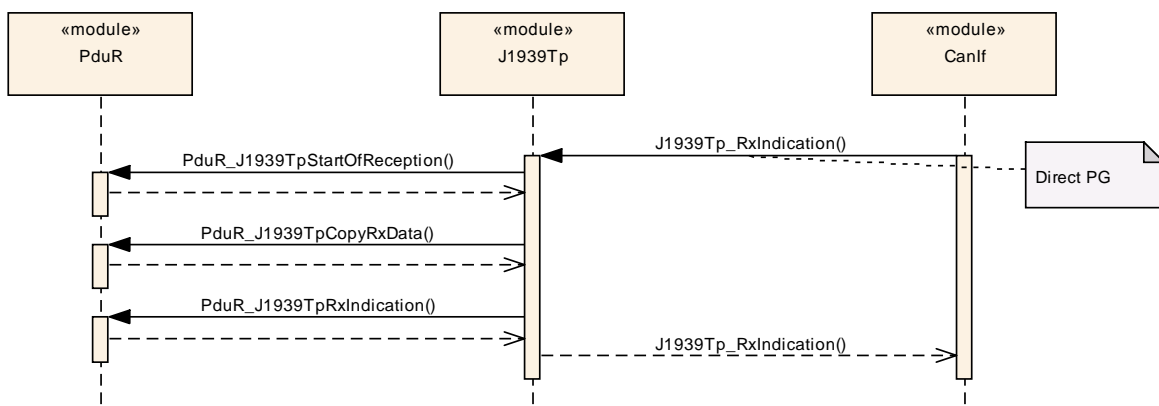


**Figure 9-1: Reception of Direct PG**

## 9.2 Reception via BAM

The following diagram shows the interaction of the J1939 Transport Layer module with the CAN Interface and the PDU Router during reception of a PG via BAM, i.e. an N-SDU that is larger than 8 bytes and is sent to the whole network.
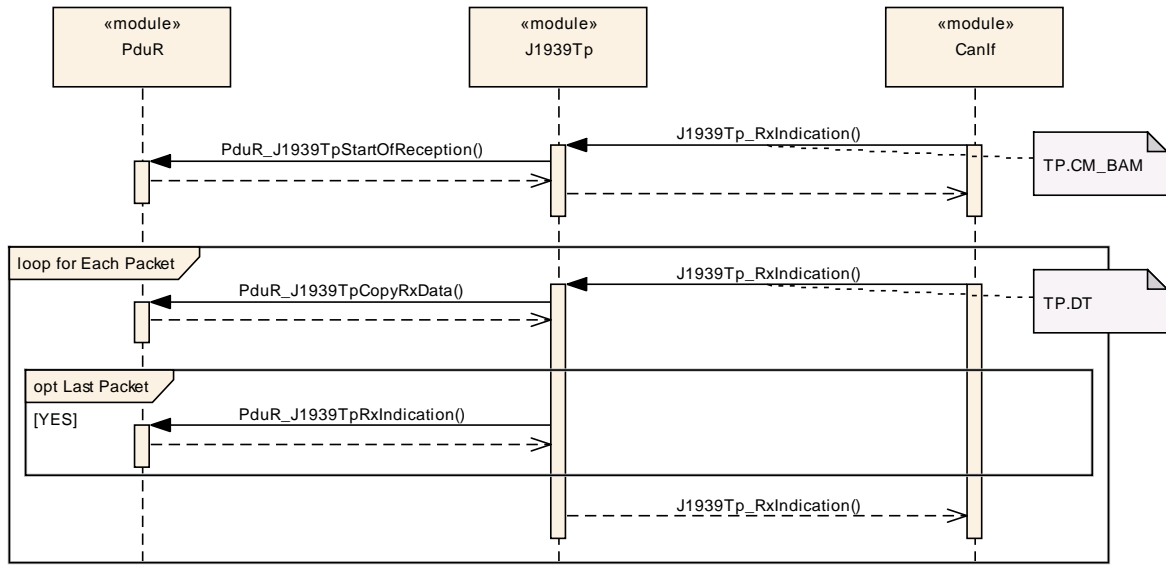
**Figure 9-2: Reception via BAM**

## 9.3 Reception via CMDT

The following diagram shows the interaction of the J1939 Transport Layer module with the CAN Interface and the PDU Router during reception of a PG via CMDT, i.e. an N-SDU that is larger than 8 bytes and is sent directly to the receiving node.
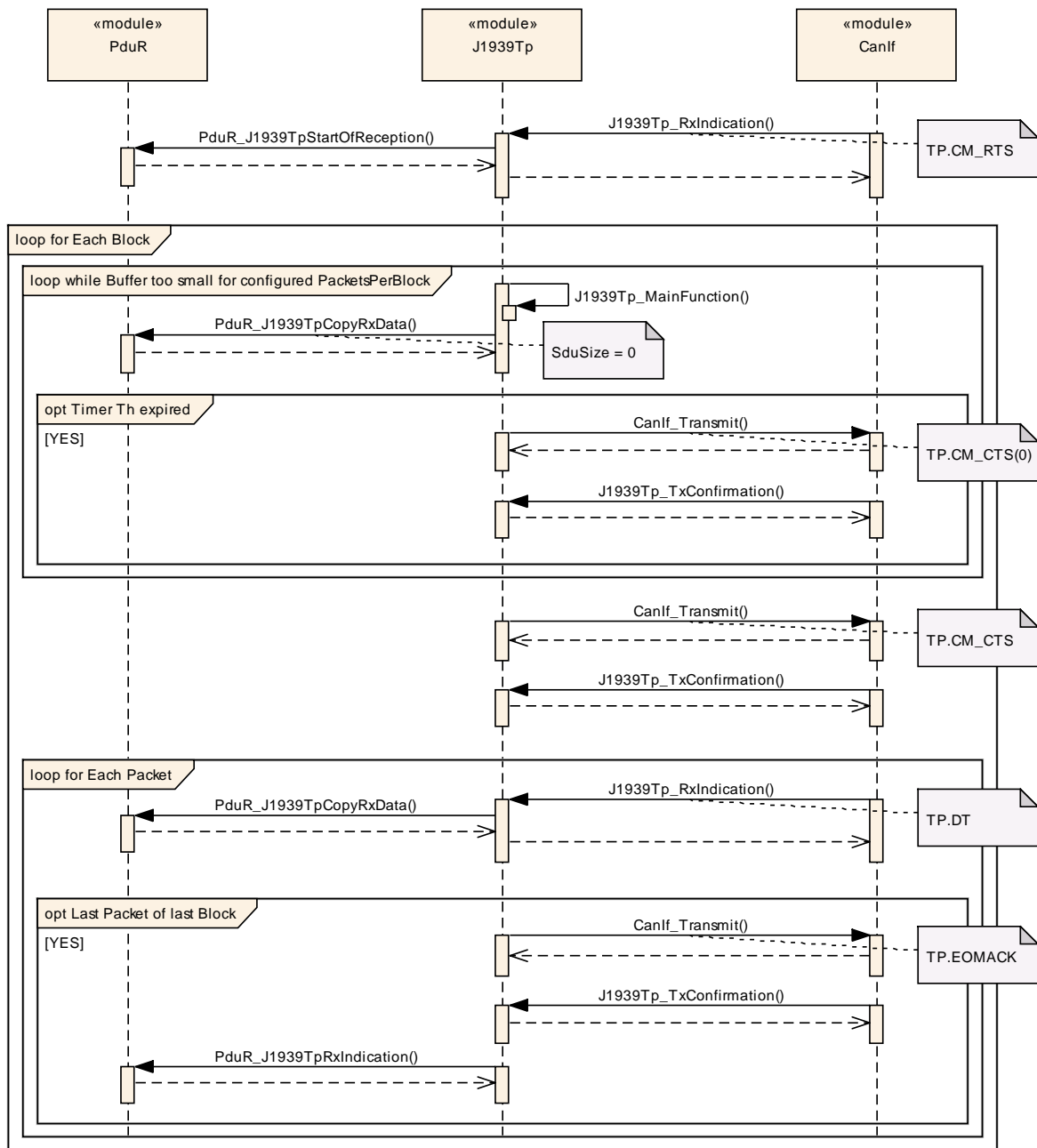
**Figure 9-3: Reception via CMDT**

## 9.4 Transmission of Direct PG

The following diagram shows the interaction of the J1939 Transport Layer module with the PDU Router and the CAN Interface during transmission of a direct PG, i.e. an N-SDU with dynamic length that is not larger than 8 bytes.
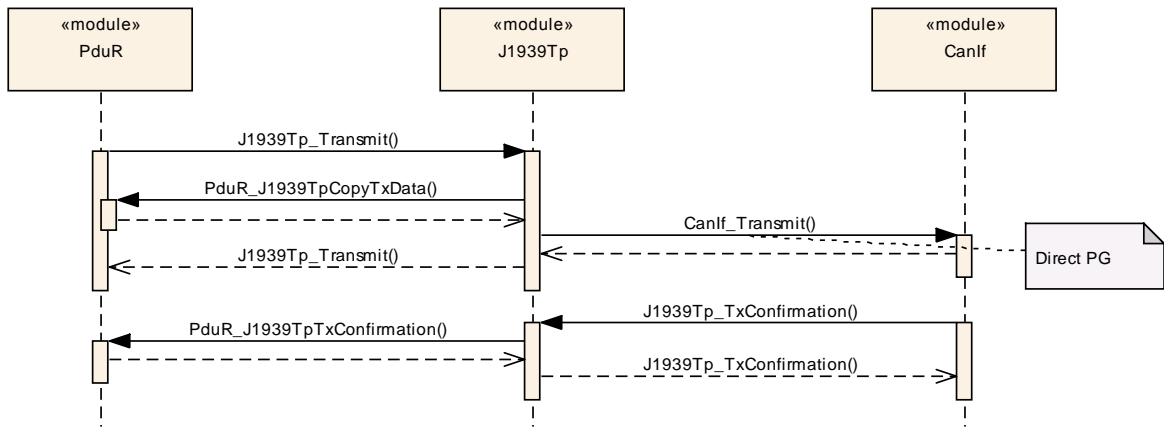
**Figure 9-4: Transmission of Direct PG**

## 9.5 Transmission via BAM

The following diagram shows the interaction of the J1939 Transport Layer module with the PDU Router and the CAN Interface during transmission of a PG via BAM, i.e. an N-SDU that is larger than 8 bytes and is sent to the whole network.
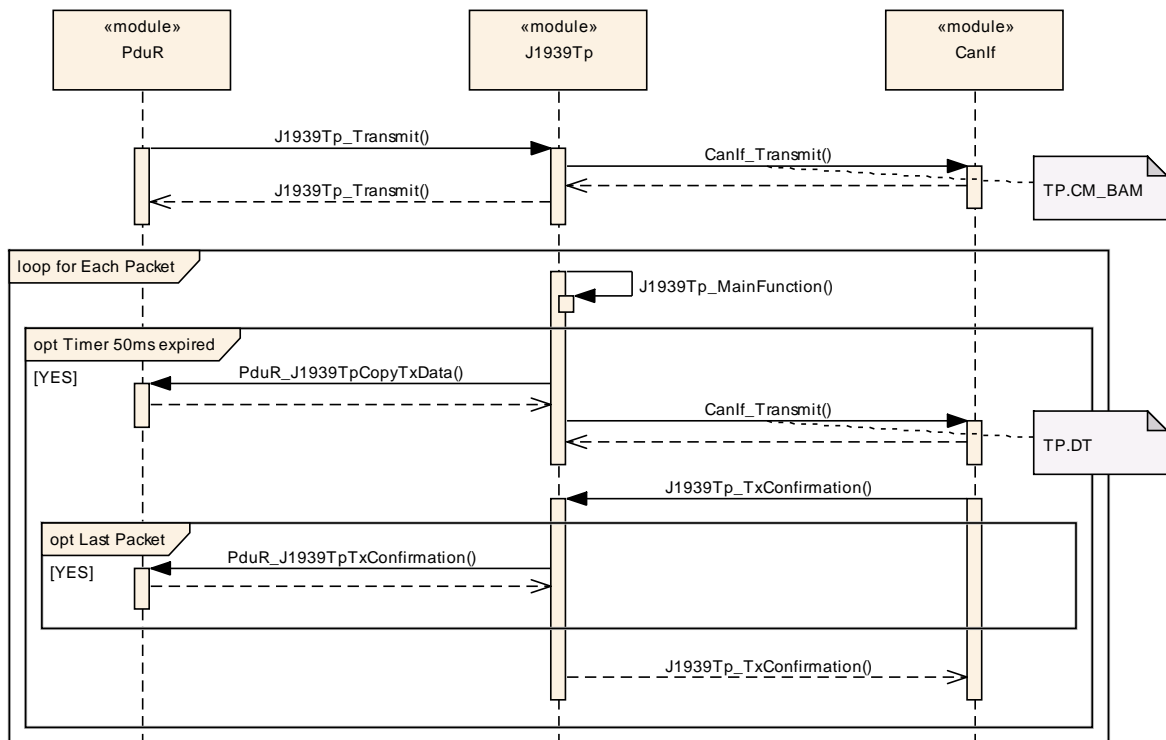
**Figure 9-5: Transmission via BAM**

## 9.6 Transmission via CMDT

The following diagram shows the interaction of the J1939 Transport Layer module with the PDU Router and the CAN Interface during transmission of a PG via CMDT, i.e. an N-SDU that is larger than 8 bytes and is sent directly to the receiving node.
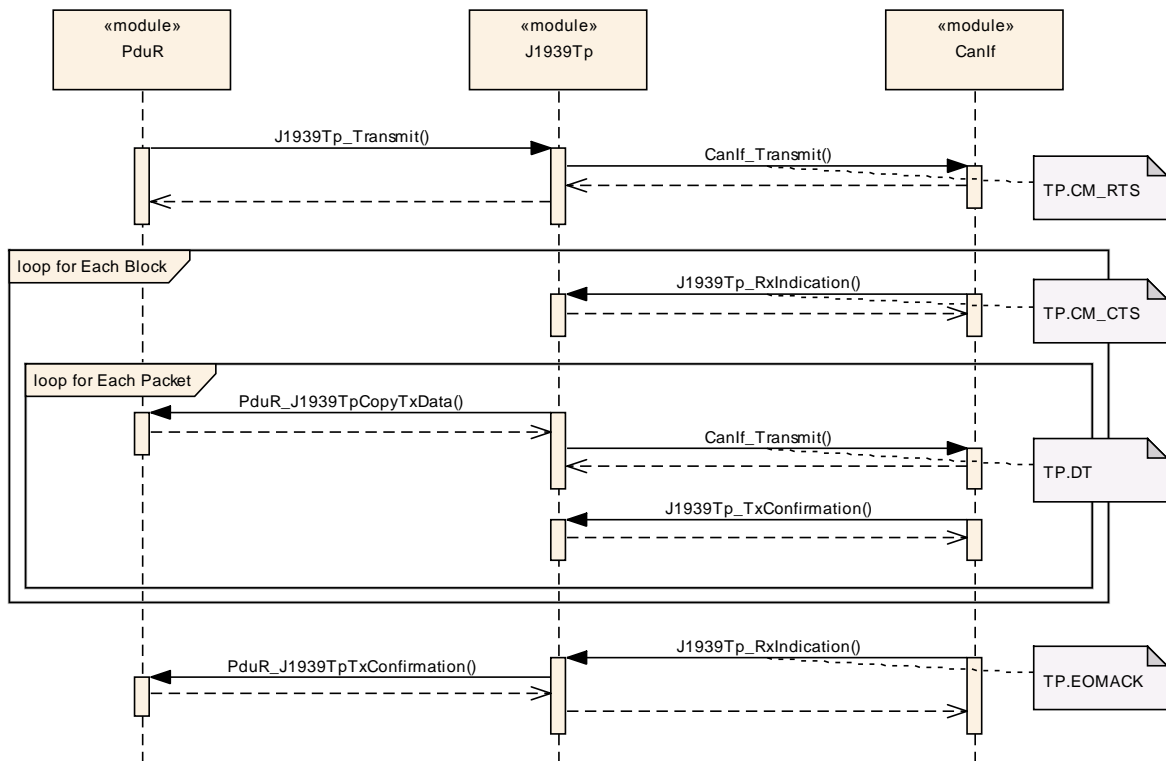
**Figure 9-6: Transmission via CMDT**

## 9.7 Handling of Retry during CMDT Transmission

The following diagram shows the interaction of the J1939 Transport Layer module with the PDU Router in the sender and the receiver node during transmission of a PG via CMDT when a retry is performed because some data is lost.
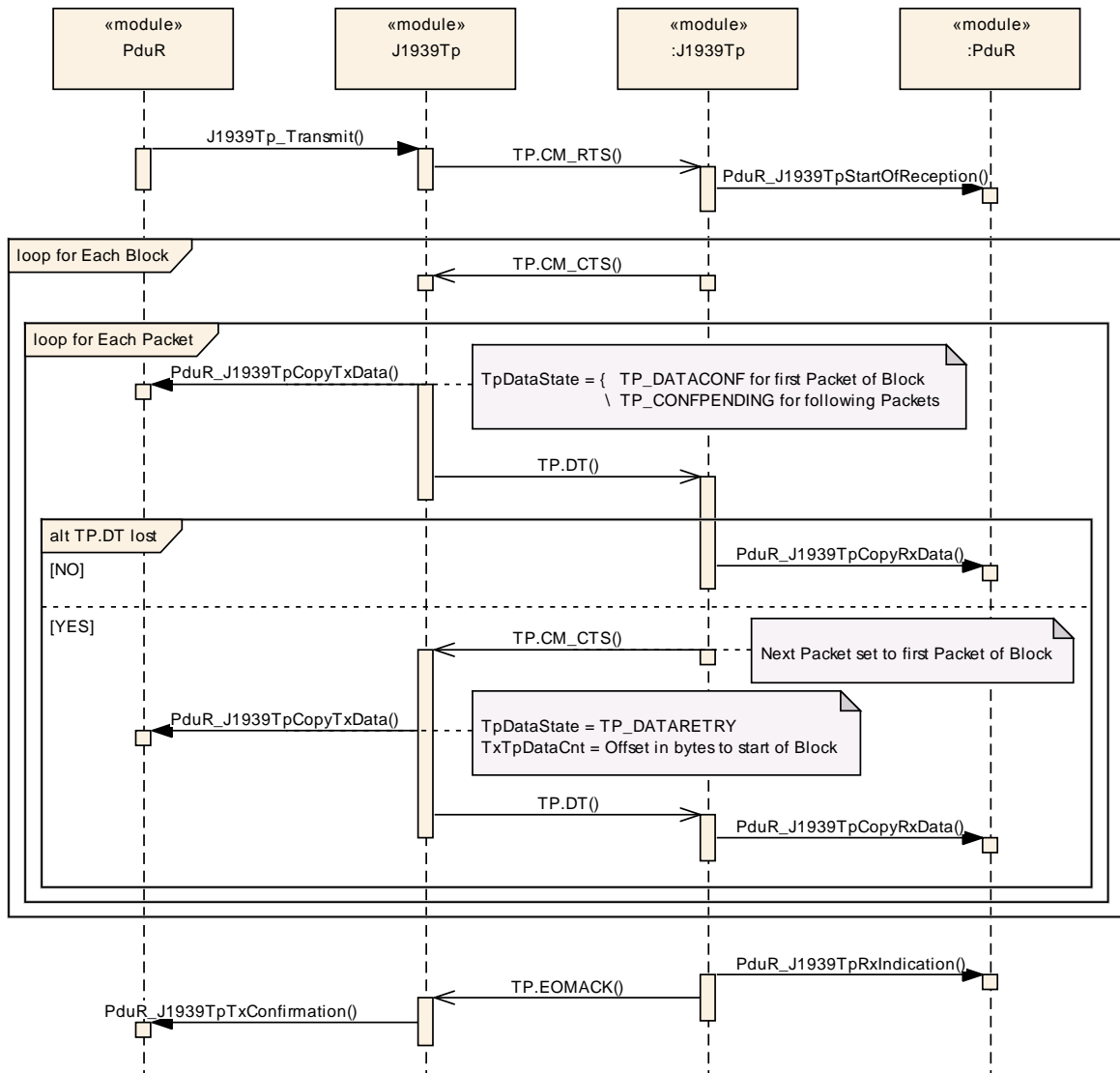
**Figure 9-7: Retry Handling**

# 10 Configuration Specification

In general, this chapter defines configuration parameters and their clustering into containers. Section 10.1 describes fundamentals and the template used for the tables in section 10.2. Section 10.2 specifies the configuration structure (containers) and the configuration parameters of J1939Tp. Section 10.3 specifies published information of J1939Tp.

The configuration parameters are derived from a network description database, which is based on the System Template. The configuration tool will extract all relevant information to configure the J1939 Transport Protocol.

**[SWS_J1939Tp_00084]** ⌈
The consistency of the configuration must be checked by the configuration tool at configuration time. Configuration rules and constraints for plausibility checks will be performed where possible, during configuration time. ⌋ (SRS_BSW_00167)

## 10.1 How to Read this Chapter

For details, refer to the chapter 10.1 "Introduction to configuration specification" in the SWS BSW General [15].

## 10.2 Containers and Configuration Parameters

The following sections summarize all configuration parameters. Additional information on the usage of these parameters can be found in chapter 7 and 8.
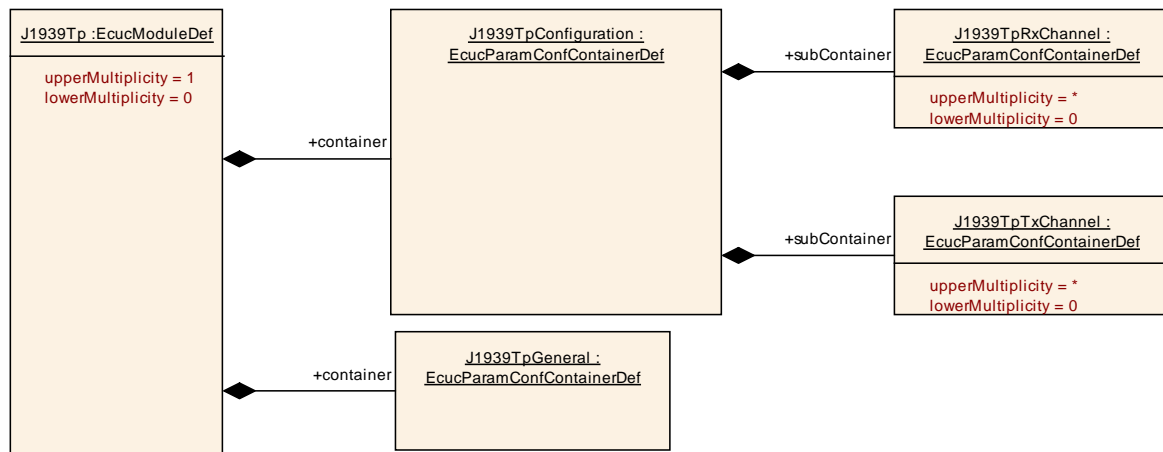


**Figure 10-1: Module Configuration**

Document ID 425: AUTOSAR_SWS_SAEJ1939TransportLayer
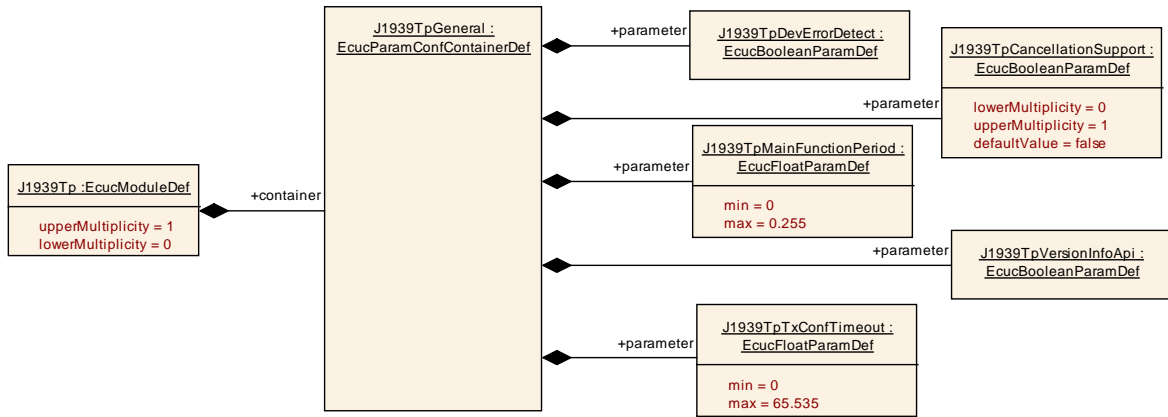
**Figure 10-2: General Parameters**
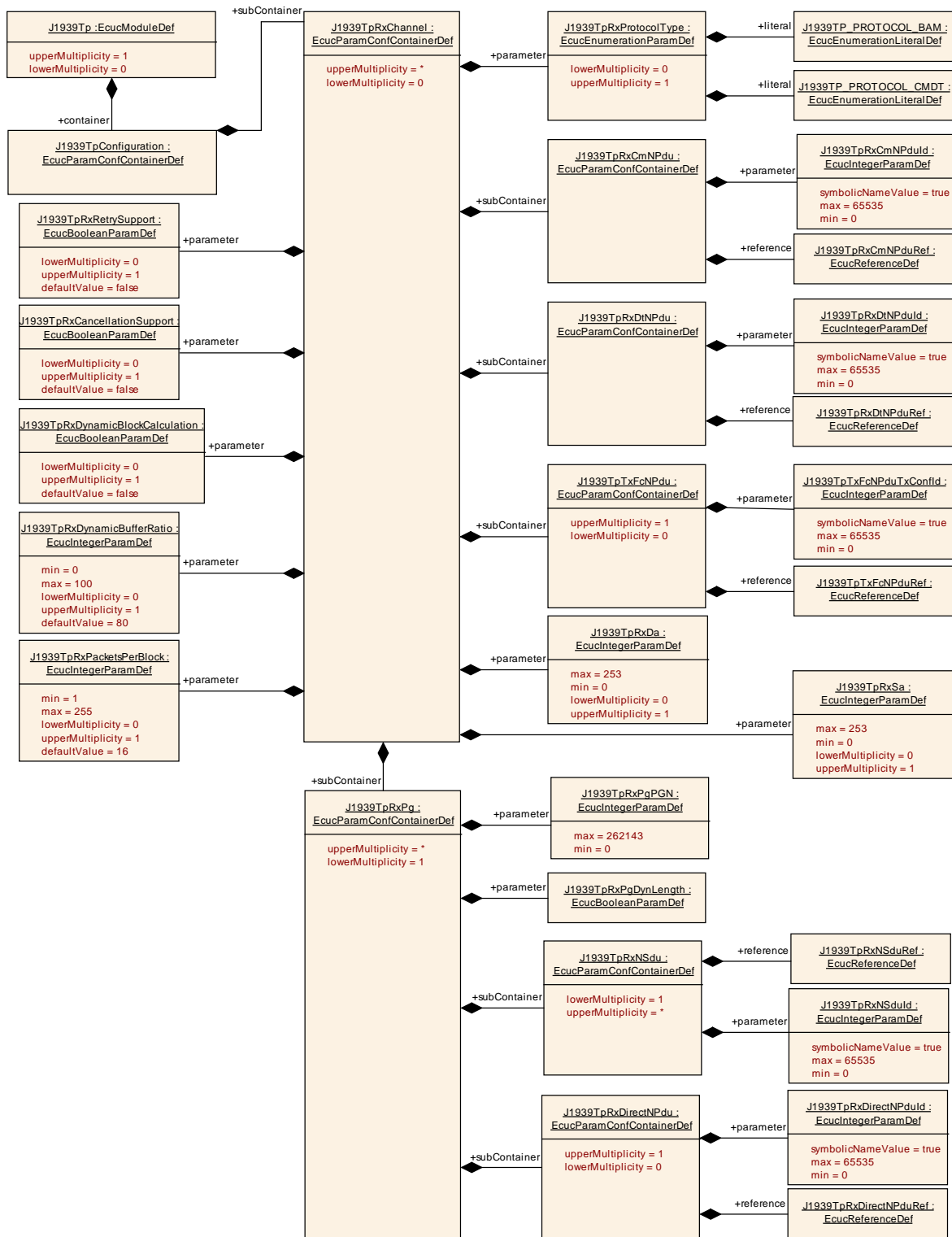
**Figure 10-3: Configuration of Rx Channel**
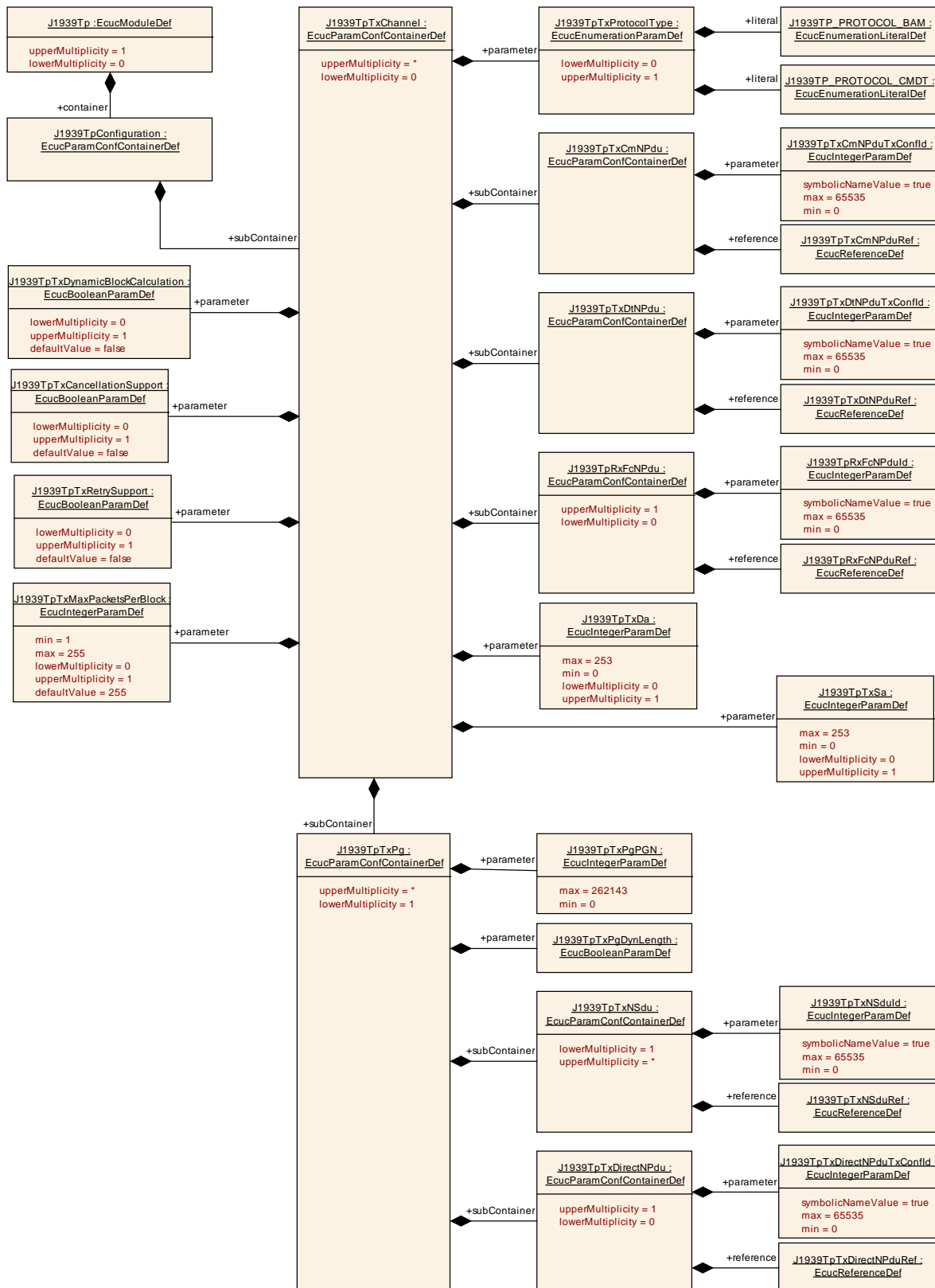
**Figure 10-4: Configuration of Tx Channel**
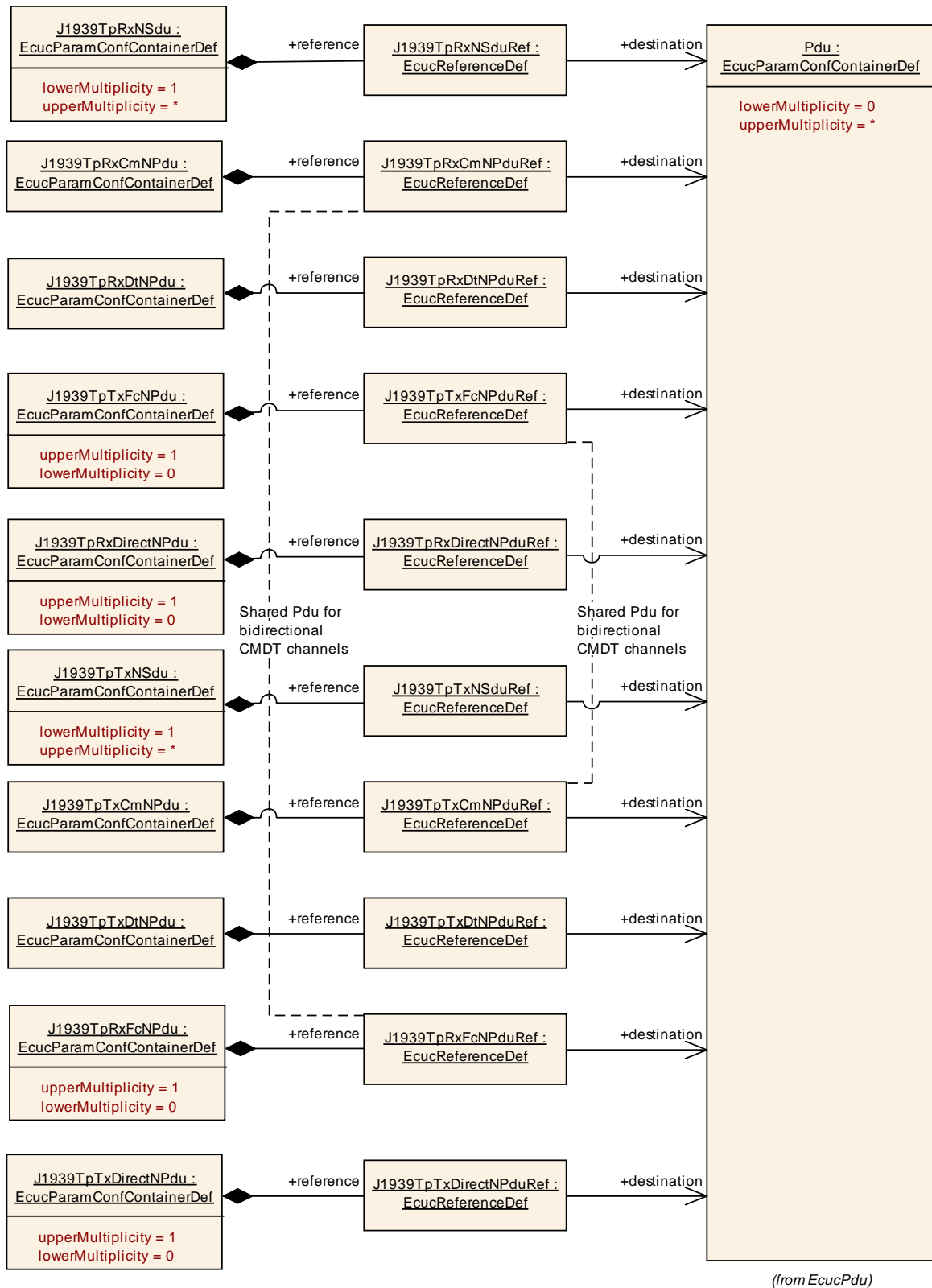
**Figure 10-5: References to PDUs**

## 10.2.1 J1939Tp

| SWS Item | ECUC_J1939Tp_00127 : |
|---|---|
| Module Name | J1939Tp |
| Module Description | Configuration of the J1939Tp (J1939 Transport Protocol) module. |
| Post-Build Variant Support | true |

| Included Containers | | |
|---|---|---|
| Container Name | Multiplicity | Scope / Dependency |
| J1939TpConfiguration | 1 | This container contains the configuration parameters and sub containers of the J1939Tp module that define the communication paths. |
| J1939TpGeneral | 1 | This container describes the general configuration parameters of the J1939Tp module. |

## 10.2.2 J1939TpGeneral

| SWS Item | ECUC_J1939Tp_00033 : |
|---|---|
| Container Name | J1939TpGeneral |
| Description | This container describes the general configuration parameters of the J1939Tp module. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00174 : | | |
|---|---|---|---|
| Name | J1939TpCancellationSupport | | |
| Description | Enable transmit and receive cancellation. The APIs J1939Tp_CancelTransmit() and J1939Tp_CancelReceive() will only be available when this parameter is enabled. | | |
| Multiplicity | 0..1 | | |
| Type | EcucBooleanParamDef | | |
| Default value | false | | |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00042 : |
|---|---|
| Name | J1939TpDevErrorDetect |
| Description | Switches the Default Error Tracer (Det) detection and notification ON or OFF.<br><br>• true: enabled (ON).<br>• false: disabled (OFF). |
| Multiplicity | 1 |
| Type | EcucBooleanParamDef |
| Default value | -- |

- AUTOSAR confidential -

| Post-Build Variant Value | false | | |
|---|---|---|---|
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00044 : | | |
|---|---|---|---|
| Name | J1939TpMainFunctionPeriod | | |
| Description | Allow to configure the time for the MainFunction (in seconds). Please note: This configuration value shall be equal to the value in the ScheduleManager module. | | |
| Multiplicity | 1 | | |
| Type | EcucFloatParamDef | | |
| Range | 0 .. 0.255 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: ECU | | |

| SWS Item | ECUC_J1939Tp_00006 : | | |
|---|---|---|---|
| Name | J1939TpTxConfTimeout | | |
| Description | Timeout in seconds for the CanIf Tx confirmation. After this time the J1939Tp assumes that an N-PDU could not be transmitted. Please note: The Tx confirmation timeout should be set to a value that enabled detection of a lost Tx confirmation in time, and that ensures that normal transmission delay caused by lower message priority does not lead to an error. | | |
| Multiplicity | 1 | | |
| Type | EcucFloatParamDef | | |
| Range | 0 .. 65.535 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00051 : | | |
|---|---|---|---|
| Name | J1939TpVersionInfoApi | | |
| Description | The function J1939Tp_GetVersionInfo is configurable (On/Off) by this configuration parameter. | | |
| Multiplicity | 1 | | |
| Type | EcucBooleanParamDef | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| No Included Containers |
|---|

## 10.2.3 J1939TpConfiguration

| SWS Item | ECUC_J1939Tp_00052 : |
|---|---|
| Container Name | J1939TpConfiguration |
| Description | This container contains the configuration parameters and sub containers of the J1939Tp module that define the communication paths. |
| Configuration Parameters | |

| Included Containers | | |
|---|---|---|
| Container Name | Multiplicity | Scope / Dependency |
| J1939TpRxChannel | 0..* | This container describes a reception channel of the J1939Tp module. A channel referencing N-PDUs without MetaData is used for all N-SDUs that share the same source address (SA) and the same destination address (BAM: DA = 0xFF, CMDT: DA != 0xFF). A channel with N-PDUs with MetaData is used for all possible source and destination addresses. |
| J1939TpTxChannel | 0..* | This container describes a transmission channel of the J1939Tp module. A channel referencing N-PDUs without MetaData is used for all N-SDUs that share the same source address (SA) and the same destination address (BAM: DA = 0xFF, CMDT: DA != 0xFF). A channel with N-PDUs with MetaData is used for all possible source and destination addresses. |

## 10.2.4 J1939TpRxChannel

| SWS Item | ECUC_J1939Tp_00053 : |
|---|---|
| Container Name | J1939TpRxChannel |
| Description | This container describes a reception channel of the J1939Tp module. A channel referencing N-PDUs without MetaData is used for all N-SDUs that share the same source address (SA) and the same destination address (BAM: DA = 0xFF, CMDT: DA != 0xFF). A channel with N-PDUs with MetaData is used for all possible source and destination addresses. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00186 : | | |
|---|---|---|---|
| Name | J1939TpRxCancellationSupport | | |
| Description | Enable receive cancellation using the API J1939Tp_CancelReceive() for this channel. | | |
| Multiplicity | 0..1 | | |
| Type | EcucBooleanParamDef | | |
| Default value | false | | |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00178 : |
|---|---|

| Name | J1939TpRxDa | | |
|---|---|---|---|
| Description | Destination address (DA) of this channel. This parameter is only required for channels with fixed DA which use N-PDUs with MetaData containing the DA. | | |
| Multiplicity | 0..1 | | |
| Type | EcucIntegerParamDef | | |
| Range | 0 .. 253 | | |
| Default value | -- | | |
| Post-Build Variant Multiplicity | true | | |
| Post-Build Variant Value | true | | |
| Multiplicity Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00187 : | | |
|---|---|---|---|
| Name | J1939TpRxDynamicBlockCalculation | | |
| Description | Enable dynamic calculation of "number of packets that can be sent" value in TP.CM_CTS, based on the size of buffers in upper layers reported via StartOfReception and PduR_J1939TpCopyRxData. | | |
| Multiplicity | 0..1 | | |
| Type | EcucBooleanParamDef | | |
| Default value | false | | |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00188 : | | |
|---|---|---|---|
| Name | J1939TpRxDynamicBufferRatio | | |
| Description | Percentage of available buffer that shall be used for retry. This parameter is only applicable when "J1939TpRxRetrySupport" and "J1939TpRxDynamicBlockCalculation" are enabled. | | |
| Multiplicity | 0..1 | | |
| Type | EcucIntegerParamDef | | |
| Range | 0 .. 100 | | |
| Default value | 80 | | |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Value Configuration Class | Pre-compile time | X | All Variants |

| | | | |
|---|---|---|---|
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local<br>dependency: Only applicable when "J1939TpRxRetrySupport" and "J1939TpRxDynamicBlockCalculation" are enabled | | |

| SWS Item | ECUC_J1939Tp_00189 : | | |
|---|---|---|---|
| Name | J1939TpRxPacketsPerBlock | | |
| Description | Number of TP.DT frames the receiving J1939Tp module allows the sender to send before waiting for another TP.CM_CTS. This parameter is transmitted in the TP.CM_CTS frame, and is thus only relevant for reception of messages via CMDT. When J1939TpRxDynamicBlockCalculation is enabled, this parameter specifies a maximum for the calculated value. For further details on this parameter value see SAE J1939/21. | | |
| Multiplicity | 0..1 | | |
| Type | EcucIntegerParamDef | | |
| Range | 1 .. 255 | | |
| Default value | 16 | | |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00029 : | | |
|---|---|---|---|
| Name | J1939TpRxProtocolType | | |
| Description | Protocol type used by this channel. This parameter is only required for channels with fixed destination address. | | |
| Multiplicity | 0..1 | | |
| Type | EcucEnumerationParamDef | | |
| Range | J1939TP_PROTOCOL_BAM | J1939 transport protocol type BAM (Broadcast Announce Message). This protocol uses two N-PDUs: The CmNPdu and the DtNPdu. | |
| | J1939TP_PROTOCOL_CMDT | J1939 transport protocol type CMDT (Connection Mode Data Transfer). This protocol uses three N-PDUs: The CmNPdu, the DtNPdu, and the FcNPdu. | |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00185 : | | |
|---|---|---|---|
| Name | J1939TpRxRetrySupport | | |
| Description | Enable support for triggering repetition of failed transmission using TP.CM_CTS with a packet number that has already been sent. Retransmission is triggered when a sequence number is missing or a timeout occurs during reception. | | |
| Multiplicity | 0..1 | | |
| Type | EcucBooleanParamDef | | |
| Default value | false | | |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00179 : | | |
|---|---|---|---|
| Name | J1939TpRxSa | | |
| Description | Source address (SA) of this channel. This parameter is only required for channels with fixed SA which use N-PDUs with MetaData. | | |
| Multiplicity | 0..1 | | |
| Type | EcucIntegerParamDef | | |
| Range | 0 .. 253 | | |
| Default value | -- | | |
| Post-Build Variant Multiplicity | true | | |
| Post-Build Variant Value | true | | |
| Multiplicity Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

| Included Containers | | |
|---|---|---|
| Container Name | Multiplicity | Scope / Dependency |
| J1939TpRxCmNPdu | 1 | This N-PDU represents the TP.CM frame of a J1939 transport protocol session. TP.CM is used both by BAM and CMDT to initialize the connection. For CMDT, it is also used to abort the connection. |
| J1939TpRxDtNPdu | 1 | This N-PDU represents the TP.DT frame of a J1939 transport protocol session. TP.DT is used both by BAM and CMDT to transfer the contents of an N-SDU. |
| J1939TpRxPg | 1..* | Parameter group received by the J1939 transport layer. |
| J1939TpTxFcNPdu | 0..1 | This N-PDU represents the TP.CM frame that is used in reverse direction for a J1939 transport protocol session using the CMDT protocol type. TP.CM in reverse direction is used for intermediate and final acknowledgement of received data and to abort the connection. |

| | |
|---|---|
| | Please note: This sub container is only required when J1939TpRxProtocolType is J1939TP_PROTOCOL_CMDT or when it is not configured at all. |

## 10.2.5 J1939TpRxCmNPdu

| SWS Item | ECUC_J1939Tp_00128 : |
|---|---|
| Container Name | J1939TpRxCmNPdu |
| Description | This N-PDU represents the TP.CM frame of a J1939 transport protocol session. TP.CM is used both by BAM and CMDT to initialize the connection. For CMDT, it is also used to abort the connection. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00129 : | | |
|---|---|---|---|
| Name | J1939TpRxCmNPduId | | |
| Description | The N-PDU identifier used for communication with CanIf. | | |
| Multiplicity | 1 | | |
| Type | EcucIntegerParamDef (Symbolic Name generated for this parameter) | | |
| Range | 0 .. 65535 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: ECU | | |

| SWS Item | ECUC_J1939Tp_00158 : | | |
|---|---|---|---|
| Name | J1939TpRxCmNPduRef | | |
| Description | Reference to the Pdu object representing the N-PDU. | | |
| Multiplicity | 1 | | |
| Type | Reference to [ Pdu ] | | |
| Post-Build Variant Value | true | | |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

**No Included Containers**

## 10.2.6 J1939TpRxDtNPdu

| SWS Item | ECUC_J1939Tp_00117 : |
|---|---|
| Container Name | J1939TpRxDtNPdu |
| Description | This N-PDU represents the TP.DT frame of a J1939 transport protocol session. TP.DT is used both by BAM and CMDT to transfer the contents of an N-SDU. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00133 : |
|---|---|

| Name | J1939TpRxDtNPduId | | |
|---|---|---|---|
| Description | The N-PDU identifier used for communication with CanIf. | | |
| Multiplicity | 1 | | |
| Type | EcucIntegerParamDef (Symbolic Name generated for this parameter) | | |
| Range | 0 .. 65535 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: ECU | | |

| SWS Item | ECUC_J1939Tp_00134 : | | |
|---|---|---|---|
| Name | J1939TpRxDtNPduRef | | |
| Description | Reference to the Pdu object representing the N-PDU. | | |
| Multiplicity | 1 | | |
| Type | Reference to [ Pdu ] | | |
| Post-Build Variant Value | true | | |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

**No Included Containers**

## 10.2.7 J1939TpRxPg

| SWS Item | ECUC_J1939Tp_00050 : |
|---|---|
| Container Name | J1939TpRxPg |
| Description | Parameter group received by the J1939 transport layer. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00066 : | | |
|---|---|---|---|
| Name | J1939TpRxPgDynLength | | |
| Description | This flag is set to TRUE when the N-SDU refers to a PGN with variable length.<br>Please note: When this attribute is TRUE, the sub container J1939TpRxDirectNPdu is required. | | |
| Multiplicity | 1 | | |
| Type | EcucBooleanParamDef | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | | | |

| SWS Item | ECUC_J1939Tp_00065 : |
|---|---|
| Name | J1939TpRxPgPGN |
| Description | PGN of the referenced N-SDUs. |
| Multiplicity | 1 |
| Type | EcucIntegerParamDef |

| Range | 0 .. 262143 | | |
|---|---|---|---|
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| Included Containers | | |
|---|---|---|
| Container Name | Multiplicity | Scope / Dependency |
| J1939TpRxDirectNPdu | 0..1 | This N-PDU represents the short frame that is used for a dynamic length PGN when it has a length of less that 8 bytes. Please note: This sub container is only necessary when J1939TpRxPgDynLength is TRUE. |
| J1939TpRxNSdu | 1..* | This container describes the parameters that are relevant for the reception of a specific N-SDU. |

## 10.2.8 J1939TpRxDirectNPdu

| SWS Item | ECUC_J1939Tp_00130 : |
|---|---|
| Container Name | J1939TpRxDirectNPdu |
| Description | This N-PDU represents the short frame that is used for a dynamic length PGN when it has a length of less that 8 bytes. Please note: This sub container is only necessary when J1939TpRxPgDynLength is TRUE. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00131 : | | |
|---|---|---|---|
| Name | J1939TpRxDirectNPduId | | |
| Description | The N-PDU identifier used for communication with CanIf. | | |
| Multiplicity | 1 | | |
| Type | EcucIntegerParamDef (Symbolic Name generated for this parameter) | | |
| Range | 0 .. 65535 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: ECU | | |

| SWS Item | ECUC_J1939Tp_00132 : | | |
|---|---|---|---|
| Name | J1939TpRxDirectNPduRef | | |
| Description | Reference to the Pdu object representing the N-PDU. | | |
| Multiplicity | 1 | | |
| Type | Reference to [ Pdu ] | | |
| Post-Build Variant Value | true | | |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

| No Included Containers |
|---|

### 10.2.9 J1939TpRxNSdu

| SWS Item | ECUC_J1939Tp_00063 : |
|---|---|
| Container Name | J1939TpRxNSdu |
| Description | This container describes the parameters that are relevant for the reception of a specific N-SDU. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00184 : | | |
|---|---|---|---|
| Name | J1939TpRxNSduId | | |
| Description | This is a unique identifier for a received N-SDU. This Id is used in the CancelReceive and ChangeParameter API call. | | |
| Multiplicity | 1 | | |
| Type | EcucIntegerParamDef (Symbolic Name generated for this parameter) | | |
| Range | 0 .. 65535 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: ECU | | |

| SWS Item | ECUC_J1939Tp_00069 : | | |
|---|---|---|---|
| Name | J1939TpRxNSduRef | | |
| Description | Reference to the Pdu object representing the N-SDU. | | |
| Multiplicity | 1 | | |
| Type | Reference to [ Pdu ] | | |
| Post-Build Variant Value | true | | |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

| No Included Containers |
|---|

### 10.2.10    J1939TpTxFcNPdu

| SWS Item | ECUC_J1939Tp_00135 : |
|---|---|
| Container Name | J1939TpTxFcNPdu |
| Description | This N-PDU represents the TP.CM frame that is used in reverse direction for a J1939 transport protocol session using the CMDT protocol type. TP.CM in reverse direction is used for intermediate and final acknowledgement of received data and to abort the connection. Please note: This sub container is only required when J1939TpRxProtocolType is J1939TP_PROTOCOL_CMDT or when it is not configured at all. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00168 : | | |
|---|---|---|---|
| Name | J1939TpTxFcNPduTxConfId | | |
| Description | The N-PDU identifier used for Tx confirmation from CanIf. | | |
| Multiplicity | 1 | | |
| Type | EcucIntegerParamDef (Symbolic Name generated for this parameter) | | |
| Range | 0 .. 65535 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: ECU | | |

| SWS Item | ECUC_J1939Tp_00136 : | | |
|---|---|---|---|
| Name | J1939TpTxFcNPduRef | | |
| Description | Reference to the Pdu object representing the N-PDU. Please note: When two channels have identical but exchanged source and destination addresses, the Pdu referenced by this parameter is shared with J1939TpTxCmNPduRef of the corresponding J1939TpTxChannel. | | |
| Multiplicity | 1 | | |
| Type | Reference to [ Pdu ] | | |
| Post-Build Variant Value | true | | |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

| No Included Containers |
|---|

## 10.2.11    J1939TpTxChannel

| SWS Item | ECUC_J1939Tp_00059 : |
|---|---|
| Container Name | J1939TpTxChannel |
| Description | This container describes a transmission channel of the J1939Tp module. A channel referencing N-PDUs without MetaData is used for all N-SDUs that share the same source address (SA) and the same destination address (BAM: DA = 0xFF, CMDT: DA != 0xFF). A channel with N-PDUs with MetaData is used for all possible source and destination addresses. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00192 : | | |
|---|---|---|---|
| Name | J1939TpTxCancellationSupport | | |
| Description | Enable transmit cancellation using the API J1939Tp_CancelTransmit() for this channel. | | |
| Multiplicity | 0..1 | | |
| Type | EcucBooleanParamDef | | |
| Default value | false | | |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |

| | Post-build time | -- | |
|---|---|---|---|
| **Value Configuration Class** | **Pre-compile time** | X | All Variants |
| | **Link time** | -- | |
| | **Post-build time** | -- | |
| **Scope / Dependency** | scope: local | | |

| **SWS Item** | **ECUC_J1939Tp_00180 :** | | |
|---|---|---|---|
| **Name** | J1939TpTxDa | | |
| **Description** | Destination address (DA) of this channel. This parameter is only required for channels with fixed DA which use N-PDUs with MetaData containing the DA. | | |
| **Multiplicity** | 0..1 | | |
| **Type** | EcucIntegerParamDef | | |
| **Range** | 0 .. 253 | | |
| **Default value** | -- | | |
| **Post-Build Variant Multiplicity** | true | | |
| **Post-Build Variant Value** | true | | |
| **Multiplicity Configuration Class** | **Pre-compile time** | X | VARIANT-PRE-COMPILE |
| | **Link time** | -- | |
| | **Post-build time** | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| **Value Configuration Class** | **Pre-compile time** | X | VARIANT-PRE-COMPILE |
| | **Link time** | -- | |
| | **Post-build time** | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| **Scope / Dependency** | scope: local | | |

| **SWS Item** | **ECUC_J1939Tp_00191 :** | | |
|---|---|---|---|
| **Name** | J1939TpTxDynamicBlockCalculation | | |
| **Description** | Enable dynamic calculation of "maximum number of packets that can be sent" value in TP.CM_RTS, based on the available amount of data in upper layers reported via PduR_J1939TpCopyTxData. | | |
| **Multiplicity** | 0..1 | | |
| **Type** | EcucBooleanParamDef | | |
| **Default value** | false | | |
| **Post-Build Variant Multiplicity** | false | | |
| **Post-Build Variant Value** | false | | |
| **Multiplicity Configuration Class** | **Pre-compile time** | X | All Variants |
| | **Link time** | -- | |
| | **Post-build time** | -- | |
| **Value Configuration Class** | **Pre-compile time** | X | All Variants |
| | **Link time** | -- | |
| | **Post-build time** | -- | |
| **Scope / Dependency** | scope: local | | |

| **SWS Item** | **ECUC_J1939Tp_00190 :** | |
|---|---|---|
| **Name** | J1939TpTxMaxPacketsPerBlock | |
| **Description** | Maximum number of TP.DT frames the transmitting J1939Tp module is ready to send before waiting for another TP.CM_CTS. This parameter is transmitted in the TP.CM_RTS frame, and is thus only relevant for transmission of messages via CMDT. When J1939TpTxDynamicBlockCalculation is enabled, this parameter specifies a maximum for the calculated value. For further details on this parameter value see SAE J1939/21. | |

| Multiplicity | 0..1 | | |
|---|---|---|---|
| Type | EcucIntegerParamDef | | |
| Range | 1 .. 255 | | |
| Default value | 255 | | |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00137 : | | |
|---|---|---|---|
| Name | J1939TpTxProtocolType | | |
| Description | Protocol type used by this channel. This parameter is only required for channels with fixed destination address. | | |
| Multiplicity | 0..1 | | |
| Type | EcucEnumerationParamDef | | |
| Range | J1939TP_PROTOCOL_BAM | | J1939 transport protocol type BAM (Broadcast Announce Message). This protocol uses two N-PDUs: The CmNPdu and the DtNPdu. |
| | J1939TP_PROTOCOL_CMDT | | J1939 transport protocol type CMDT (Connection Mode Data Transfer). This protocol uses three N-PDUs: The CmNPdu, the DtNPdu, and the FcNPdu. |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00193 : | | |
|---|---|---|---|
| Name | J1939TpTxRetrySupport | | |
| Description | Enable support for repetition of failed transmission using TP.CM_CTS with a packet number that has already been sent. Retransmission is handled via the retry feature of PduR_J1939TpCopyTxData. | | |
| Multiplicity | 0..1 | | |
| Type | EcucBooleanParamDef | | |
| Default value | false | | |
| Post-Build Variant Multiplicity | false | | |
| Post-Build Variant Value | false | | |
| Multiplicity Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |

- AUTOSAR confidential -

| Value Configuration Class | Pre-compile time | X | All Variants |
|---|---|---|---|
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00181 : | | |
|---|---|---|---|
| Name | J1939TpTxSa | | |
| Description | Source address (SA) of this channel. This parameter is only required for channels with fixed SA which use N-PDUs with MetaData. | | |
| Multiplicity | 0..1 | | |
| Type | EcucIntegerParamDef | | |
| Range | 0 .. 253 | | |
| Default value | -- | | |
| Post-Build Variant Multiplicity | true | | |
| Post-Build Variant Value | true | | |
| Multiplicity Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

| Included Containers | | |
|---|---|---|
| Container Name | Multiplicity | Scope / Dependency |
| J1939TpRxFcNPdu | 0..1 | This N-PDU represents the TP.CM frame that is used in reverse direction for a J1939 transport protocol session using the CMDT protocol type. TP.CM in reverse direction is used for intermediate and final acknowledgement of received data and to abort the connection. Please note: This sub container is only required when J1939TpRxProtocolType is J1939TP_PROTOCOL_CMDT or when it is not configured at all. |
| J1939TpTxCmNPdu | 1 | This N-PDU represents the TP.CM frame of a J1939 transport protocol session. TP.CM is used both by BAM and CMDT to initialize the connection. For CMDT, it is also used to abort the connection. |
| J1939TpTxDtNPdu | 1 | This N-PDU represents the TP.DT frame of a J1939 transport protocol session. TP.DT is used both by BAM and CMDT to transfer the contents of an N-SDU. |
| J1939TpTxPg | 1..* | Parameter group transmitted by the J1939 transport layer. |

## 10.2.12    J1939TpRxFcNPdu

| SWS Item | ECUC_J1939Tp_00144 : |
|---|---|
| Container Name | J1939TpRxFcNPdu |
| Description | This N-PDU represents the TP.CM frame that is used in reverse direction for a J1939 transport protocol session using the CMDT protocol type. TP.CM in reverse direction is used for intermediate and final acknowledgement of received data and to abort the connection. Please note: This sub container is only required when J1939TpRxProtocolType is |

| | J1939TP_PROTOCOL_CMDT or when it is not configured at all. |
|---|---|
| **Configuration Parameters** | |

| **SWS Item** | **ECUC_J1939Tp_00145 :** | | |
|---|---|---|---|
| *Name* | J1939TpRxFcNPduId | | |
| *Description* | The N-PDU identifier used for communication with CanIf. | | |
| *Multiplicity* | 1 | | |
| *Type* | EcucIntegerParamDef (Symbolic Name generated for this parameter) | | |
| *Range* | 0 .. 65535 | | |
| *Default value* | -- | | |
| *Post-Build Variant Value* | false | | |
| *Value Configuration Class* | *Pre-compile time* | X | All Variants |
| | *Link time* | -- | |
| | *Post-build time* | -- | |
| *Scope / Dependency* | scope: ECU | | |

| **SWS Item** | **ECUC_J1939Tp_00146 :** | | |
|---|---|---|---|
| *Name* | J1939TpRxFcNPduRef | | |
| *Description* | Reference to the Pdu object representing the N-PDU. Please note: When two channels have identical but exchanged source and destination addresses, the Pdu referenced by this parameter is shared with J1939TpRxCmNPduRef of the corresponding J1939TpRxChannel. | | |
| *Multiplicity* | 1 | | |
| *Type* | Reference to [ Pdu ] | | |
| *Post-Build Variant Value* | true | | |
| *Value Configuration Class* | *Pre-compile time* | X | VARIANT-PRE-COMPILE |
| | *Link time* | -- | |
| | *Post-build time* | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| *Scope / Dependency* | scope: local | | |

| **No Included Containers** |
|---|

### 10.2.13    J1939TpTxCmNPdu

| **SWS Item** | **ECUC_J1939Tp_00138 :** |
|---|---|
| *Container Name* | J1939TpTxCmNPdu |
| *Description* | This N-PDU represents the TP.CM frame of a J1939 transport protocol session. TP.CM is used both by BAM and CMDT to initialize the connection. For CMDT, it is also used to abort the connection. |
| **Configuration Parameters** | |

| **SWS Item** | **ECUC_J1939Tp_00170 :** | | |
|---|---|---|---|
| *Name* | J1939TpTxCmNPduTxConfId | | |
| *Description* | The N-PDU identifier used for Tx confirmation from CanIf. | | |
| *Multiplicity* | 1 | | |
| *Type* | EcucIntegerParamDef (Symbolic Name generated for this parameter) | | |
| *Range* | 0 .. 65535 | | |
| *Default value* | -- | | |
| *Post-Build Variant Value* | false | | |
| *Value Configuration Class* | *Pre-compile time* | X | All Variants |
| | *Link time* | -- | |
| | *Post-build time* | -- | |

| Scope / Dependency | scope: ECU | | |
|---|---|---|---|

| SWS Item | ECUC_J1939Tp_00139 : | | |
|---|---|---|---|
| Name | J1939TpTxCmNPduRef | | |
| Description | Reference to the Pdu object representing the N-PDU. | | |
| Multiplicity | 1 | | |
| Type | Reference to [ Pdu ] | | |
| Post-Build Variant Value | true | | |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

**No Included Containers**

## 10.2.14 J1939TpTxDtNPdu

| SWS Item | ECUC_J1939Tp_00142 : | |
|---|---|---|
| Container Name | J1939TpTxDtNPdu | |
| Description | This N-PDU represents the TP.DT frame of a J1939 transport protocol session. TP.DT is used both by BAM and CMDT to transfer the contents of an N-SDU. | |
| Configuration Parameters | | |

| SWS Item | ECUC_J1939Tp_00171 : | | |
|---|---|---|---|
| Name | J1939TpTxDtNPduTxConfId | | |
| Description | The N-PDU identifier used for Tx confirmation from CanIf. | | |
| Multiplicity | 1 | | |
| Type | EcucIntegerParamDef (Symbolic Name generated for this parameter) | | |
| Range | 0 .. 65535 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: ECU | | |

| SWS Item | ECUC_J1939Tp_00143 : | | |
|---|---|---|---|
| Name | J1939TpTxDtNPduRef | | |
| Description | Reference to the Pdu object representing the N-PDU. | | |
| Multiplicity | 1 | | |
| Type | Reference to [ Pdu ] | | |
| Post-Build Variant Value | true | | |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

**No Included Containers**

## 10.2.15 J1939TpTxPg

| SWS Item | ECUC_J1939Tp_00070 : |
|---|---|
| Container Name | J1939TpTxPg |
| Description | Parameter group transmitted by the J1939 transport layer. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00148 : | | |
|---|---|---|---|
| Name | J1939TpTxPgDynLength | | |
| Description | This flag is set to TRUE when the N-SDU refers to a PGN with variable length. Please note: When this attribute is TRUE, the sub container J1939TpTxDirectNPdu is required. | | |
| Multiplicity | 1 | | |
| Type | EcucBooleanParamDef | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| SWS Item | ECUC_J1939Tp_00150 : | | |
|---|---|---|---|
| Name | J1939TpTxPgPGN | | |
| Description | PGN of the referenced N-SDUs. | | |
| Multiplicity | 1 | | |
| Type | EcucIntegerParamDef | | |
| Range | 0 .. 262143 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: local | | |

| Included Containers | | |
|---|---|---|
| Container Name | Multiplicity | Scope / Dependency |
| J1939TpTxDirectNPdu | 0..1 | This N-PDU represents the short frame that is used for a dynamic length PGN when it has a length of less that 8 bytes. Please note: This sub container is only necessary when J1939TpTxPgDynLength is TRUE. |
| J1939TpTxNSdu | 1..* | This container describes the parameters that are relevant for the transmission of a specific N-SDU. |

## 10.2.16 J1939TpTxDirectNPdu

| SWS Item | ECUC_J1939Tp_00140 : |
|---|---|
| Container Name | J1939TpTxDirectNPdu |
| Description | This N-PDU represents the short frame that is used for a dynamic length PGN when it has a length of less that 8 bytes. Please note: This sub container is only necessary when J1939TpTxPgDynLength is TRUE. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00169 : | | |
|---|---|---|---|
| Name | J1939TpTxDirectNPduTxConfId | | |
| Description | The N-PDU identifier used for Tx confirmation from CanIf. | | |
| Multiplicity | 1 | | |
| Type | EcucIntegerParamDef (Symbolic Name generated for this parameter) | | |
| Range | 0 .. 65535 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: ECU | | |

| SWS Item | ECUC_J1939Tp_00141 : | | |
|---|---|---|---|
| Name | J1939TpTxDirectNPduRef | | |
| Description | Reference to the Pdu object representing the N-PDU. | | |
| Multiplicity | 1 | | |
| Type | Reference to [ Pdu ] | | |
| Post-Build Variant Value | true | | |
| Value Configuration Class | Pre-compile time | X | VARIANT-PRE-COMPILE |
| | Link time | -- | |
| | Post-build time | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| Scope / Dependency | scope: local | | |

| No Included Containers |
|---|

## 10.2.17 J1939TpTxNSdu

| SWS Item | ECUC_J1939Tp_00147 : |
|---|---|
| Container Name | J1939TpTxNSdu |
| Description | This container describes the parameters that are relevant for the transmission of a specific N-SDU. |
| Configuration Parameters | |

| SWS Item | ECUC_J1939Tp_00149 : | | |
|---|---|---|---|
| Name | J1939TpTxNSduId | | |
| Description | The N-SDU identifier used for communication with PduR. | | |
| Multiplicity | 1 | | |
| Type | EcucIntegerParamDef (Symbolic Name generated for this parameter) | | |
| Range | 0 .. 65535 | | |
| Default value | -- | | |
| Post-Build Variant Value | false | | |
| Value Configuration Class | Pre-compile time | X | All Variants |
| | Link time | -- | |
| | Post-build time | -- | |
| Scope / Dependency | scope: ECU | | |

| SWS Item | ECUC_J1939Tp_00151 : |
|---|---|
| Name | J1939TpTxNSduRef |
| Description | Reference to the Pdu object representing the N-SDU. |
| Multiplicity | 1 |

- AUTOSAR confidential -

| Type | Reference to [ Pdu ] | | |
|---|---|---|---|
| **Post-Build Variant Value** | true | | |
| **Value Configuration Class** | **Pre-compile time** | X | VARIANT-PRE-COMPILE |
| | **Link time** | -- | |
| | **Post-build time** | X | VARIANT-LINK-TIME, VARIANT-POST-BUILD |
| **Scope / Dependency** | scope: local | | |

| No Included Containers |
|---|

## 10.3 Published Information

For details, refer to the chapter 10.3 "Published Information" in the SWS BSW General [15].

# 11 Not applicable requirements

**[SWS_J1939Tp_99999]** ⌈These requirements are not applicable to this specification.⌋ (SRS_BSW_00005, SRS_BSW_00161, SRS_BSW_00162, SRS_BSW_00164, SRS_BSW_00168, SRS_BSW_00170, SRS_BSW_00314, SRS_BSW_00325, SRS_BSW_00341, SRS_BSW_00347, SRS_BSW_00375, SRS_BSW_00377, SRS_BSW_00413, SRS_BSW_00415, SRS_BSW_00416, SRS_BSW_00417, SRS_BSW_00419, SRS_BSW_00423, SRS_BSW_00427, SRS_BSW_00433, SRS_BSW_00437, SRS_BSW_00439, SRS_BSW_00440, SRS_BSW_00447, SRS_BSW_00449, SRS_BSW_00453)