

<b>Document Title</b>	Requirements on FlexRay
<b>Document Owner</b>	AUTOSAR
<b>Document Responsibility</b>	AUTOSAR
<b>Document Identification No</b>	005
<b>Document Classification</b>	Auxiliary

<b>Document Version</b>	3.1.0
<b>Document Status</b>	Final
<b>Part of Release</b>	4.0
<b>Revision</b>	3

<b>Document Change History</b>			
<b>Date</b>	<b>Version</b>	<b>Changed by</b>	<b>Change Description</b>
09.12.2011	3.1.0	AUTOSAR Administration	<ul style="list-style-type: none"> <li>Added "Wake-pin" as wake-up Reason</li> <li>Update of ISO 15765-2 and ISO 15765-4 support</li> </ul>
30.11.2009	3.0.0	AUTOSAR Administration	<ul style="list-style-type: none"> <li>Added support for FlexRay 3.0 hardware (CCs and transceivers)</li> <li>Added functionalities to get detailed (error) information of the communications bus</li> <li>Bus Guardian support removed</li> <li>Support for cluster related APIs removed</li> <li>Legal disclaimer revised</li> </ul>
23.06.2008	2.0.4	AUTOSAR Administration	Legal disclaimer revised
31.10.2007	2.0.3	AUTOSAR Administration	<ul style="list-style-type: none"> <li>Document meta information extended</li> <li>Small layout adaptations made</li> </ul>
24.01.2007	2.0.2	AUTOSAR Administration	<ul style="list-style-type: none"> <li>"Advice for users" revised</li> <li>"Revision Information" added</li> </ul>
05.12.2006	2.0.1	AUTOSAR Administration	Legal disclaimer revised
12.04.2006	2.0.0	AUTOSAR Administration	<ul style="list-style-type: none"> <li>Major rework of most requirements to enhance clarity (e.g. ensured expressive requirement "descriptions" and only brief "short descriptions")</li> <li>Closed all open requirements, rejected some requirements, and created some new requirements</li> <li>Chapter "4.1.1 Functional Overview": "Support of the FlexRay Protocol 2.1 specification" instead of "Support of the FlexRay Protocol 2.0 specification"</li> <li>Added transceiver requirements chapter 4.7</li> <li>Changed the allowed retry mechanism in <a href="#">[BSW05083]</a></li> </ul>

**Document Change History**

<b>Date</b>	<b>Version</b>	<b>Changed by</b>	<b>Change Description</b>
23.06.2005	1.0.0	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

Known Limitations .....	9
1 Scope of document.....	10
2 How to read this document.....	11
2.1 Conventions used .....	11
2.2 Requirement structure.....	11
3 Acronyms and abbreviations .....	14
4 Functional Overview .....	15
4.1 Common Requirements (FlexRay Interface and FlexRay Driver) .....	15
4.2 FlexRay Interface .....	15
4.3 FlexRay Driver .....	16
4.4 Transport Layer FlexRay.....	16
4.5 FlexRay Time Service .....	16
4.6 FlexRay Transceiver Driver.....	16
4.6.1 Transceiver Operation Modes .....	17
4.6.2 Transceiver Error Status.....	17
4.6.3 Transceiver Wake-up Reason .....	17
4.6.4 Remarks to the FlexRay Transceiver Driver.....	17
4.6.5 Explicitly uncovered FlexRay Transceiver Functionality .....	18
4.6.6 System Basis Chip and FlexRay Transceiver Driver .....	18
5 Requirement Specification.....	19
5.1 Functional Requirements .....	19
5.1.1 Compatibility overview on supported FlexRay CC features.....	19
5.1.1.1 General requirements.....	19
5.1.1.1.1 [BSW05000] Support of Synchronous SW Modules .....	19
5.1.1.1.2 [BSW05001] Support of Asynchronous SW Modules .....	20
5.1.1.1.3 [BSW05002] FlexRay Interface and FlexRay Driver as Only Necessarily Synchronous SW Modules .....	20
5.1.1.1.4 [BSW05003] Support of Slot/Cycle Multiplexing .....	21
5.1.1.1.5 [BSW05169] Avoid Timer Interrupts during Start-up.....	22
5.1.1.1.6 [BSW05055] Avoid Timer Interrupts during Shutdown.....	22
5.1.1.1.7 [BSW005216] API to query the FlexRay Rate and Offset Correction .....	22
5.1.1.1.8 [BSW005217] API Report of FlexRay Status Data for number of Sync Frames.....	23
5.1.1.1.9 [BSW005218] Report of FlexRay Status Data for list of Sync Frame IDs .....	23
5.1.1.1.10 [BSW005219] Support for FlexRay Key Slot Mode.....	24
5.1.1.1.11 [BSW005223] Get number of Startup Frames .....	24
5.1.1.1.12 [BSW005220] Immediate Buffer Read Access .....	25
5.1.1.1.13 [BSW005225] Support of CHI Command AllowColdstart.....	25
5.1.1.1.14 [BSW005224] Support of Payload Preamble Indicator Bit .....	25
5.1.1.1.15 [BSW005221] Synchronization of FlexRay stack to the FlexRay cycle.....	26
5.1.2 FlexRay Interface .....	27
5.1.2.1 General Requirement .....	27
5.1.2.1.1 [BSW05004] PDU-Based Data API.....	27
5.1.2.1.2 [BSW05010] Unique PDU-ID .....	28
5.1.2.1.3 [BSW05126] PDU Update/Valid Information.....	29
5.1.2.1.4 [BSW05097] Number of FlexRay Drivers per FlexRay Interface .....	29

5.1.2.1.5	[BSW05007] Number of FlexRay CCs per Interface .....	30
5.1.2.1.6	[BSW05130] Transmit Request Queuing .....	30
5.1.2.1.7	[BSW005215] Transmit Request Cancellation .....	30
5.1.2.1.8	Configuration .....	31
5.1.2.1.8.1	[BSW05060] Scheduling of Copy Operation into/from FlexRay CC	31
5.1.2.1.8.2	[BSW005060] FlexRay Network Management Scheduling Timing Window Relief .....	31
5.1.2.1.8.3	[BSW005061] NM Vector Support .....	32
5.1.2.1.8.4	[BSW05096] Assignment of Drivers to Controllers .....	32
5.1.2.2	Initialization .....	33
5.1.2.2.1	[BSW05013] Initialize Local Memory Space .....	33
5.1.2.2.2	[BSW05031] Initialization of a FlexRay CC .....	33
5.1.2.2.3	[BSW05034] Configuration Modifiable by a Flashing Process .....	33
5.1.2.2.4	[BSW05042] Switch Configuration in Normal Active Mode .....	34
5.1.2.3	Start-up Operation .....	35
5.1.2.3.1	[BSW05015] Start-up of a FlexRay CC .....	35
5.1.2.3.2	[BSW05018] Sending of a Wake-Up Pattern .....	35
5.1.2.3.3	[BSW05158] Get FlexRay Transceiver Wake-up Reason .....	36
5.1.2.3.4	[BSW05159] Enable FlexRay Transceiver Wake-up Indication .....	36
5.1.2.3.5	[BSW05160] Disable FlexRay Transceiver Wake-up Indication .....	36
5.1.2.3.6	[BSW05161] Clear FlexRay Transceiver Wake-up Events .....	36
5.1.2.4	FlexRay Specific Information .....	38
5.1.2.4.1	[BSW05022] Get FlexRay CC POC Status .....	38
5.1.2.4.2	[BSW05039] Set FlexRay Transceiver Operation Mode .....	38
5.1.2.4.3	[BSW05157] Get FlexRay Transceiver Operation Mode .....	38
5.1.2.4.4	[BSW05174] Interrupt Handling .....	39
5.1.2.5	Normal Operation .....	39
5.1.2.5.1	[BSW05170] Receive PDU .....	39
5.1.2.5.2	[BSW05027] Transmit PDU .....	40
5.1.2.5.3	[BSW05171] Provide PDU Transmit Confirmation .....	40
5.1.2.6	Shutdown Operation .....	41
5.1.2.6.1	[BSW05016] Abortion of a FlexRay CC Communication .....	41
5.1.2.6.2	[BSW05063] Halt of a FlexRay CC Communication .....	41
5.1.2.7	Fault Operation .....	42
5.1.2.7.1	[BSW05175] Provide Error Information .....	42
5.1.2.7.2	[BSW005200] Error Information in Transmitted Frame .....	42
5.1.2.7.3	[BSW005201] Error Information in Received Frame .....	42
5.1.2.7.4	[BSW005202] Error Information in NIT .....	43
5.1.2.7.5	[BSW005203] Error Information in Bus Driver .....	43
5.1.2.7.6	[BSW005204] Customizable Checking Function of L-PDU Length .....	43
5.1.3	FlexRay Driver .....	44
5.1.3.1	General Requirements .....	44
5.1.3.1.1	[BSW05064] Abstraction of FlexRay CC-specific Implementation .....	44
5.1.3.1.2	[BSW05065] Number of FlexRay CCs per Driver .....	44
5.1.3.1.3	[BSW05005] Support of Hardware FIFO Mechanism .....	45
5.1.3.1.4	[BSW005006] Support of Message ID Filter Mechanism .....	45
5.1.3.1.5	[BSW005007] Support of MTM Mode .....	46
5.1.3.1.6	[BSW005008] FlexRay 2.1 and 3.0 Support .....	46
5.1.3.1.7	[BSW005009] Support of dynamic LPdu payload length .....	46
5.1.3.1.8	[BSW05024] Abstraction from CC Buffer Configuration .....	47

5.1.3.1.9	[BSW05066] L-SDU-Based API .....	47
5.1.3.1.10	[BSW005205] Transmit Request Cancellation .....	47
5.1.3.1.11	[BSW005206] Hardware Resetting Function on CC .....	48
5.1.3.2	Configuration .....	48
5.1.3.2.1	[BSW05058] Configuration of FlexRay Driver at System Configuration Time.....	48
5.1.3.2.2	[BSW05059] Transmit/Receive Buffer Configuration .....	48
5.1.3.3	Initialization.....	49
5.1.3.3.1	[BSW05116] Initialization of FlexRay CC.....	49
5.1.3.3.2	[BSW05011] Initialize Low-Level Parameters .....	49
5.1.3.3.3	[BSW05012] Initialize FlexRay CC Transmit/Receive Buffers .....	50
5.1.3.4	Start-up Operation .....	50
5.1.3.4.1	[BSW05109] Start-up of a FlexRay CC.....	50
5.1.3.4.2	[BSW05117] Sending of Wake-Up Pattern .....	50
5.1.3.5	FlexRay Specific Information.....	51
5.1.3.5.1	[BSW05120] Get FlexRay CC POC Status.....	51
5.1.3.5.2	[BSW05121] Get FlexRay CC Sync State .....	51
5.1.3.6	Normal Operation .....	52
5.1.3.6.1	[BSW05106] Buffer Reconfiguration in Normal Active Mode .....	52
5.1.3.6.2	[BSW005106] Trigger Configuration of FR CC Buffer with non- static Configuration .....	53
5.1.3.6.3	[BSW05125] Interrupt Handling .....	53
5.1.3.7	Shutdown Operation.....	54
5.1.3.7.1	[BSW05114] Abortion of FlexRay CC Communication .....	54
5.1.3.7.2	[BSW05115] Halt of FlexRay CC Communication .....	54
5.1.3.7.3	[BSW005207] API to Detect Error Information in Transmitted Frame .....	54
5.1.3.7.4	[BSW005208] API to Detect Error Information in Received Frame.....	55
5.1.3.7.5	[BSW005209] API to Detect Error Information in NIT .....	55
5.1.3.7.6	[BSW005210] Criteria on Validness of Received Frame.....	56
5.1.3.8	Fault Operation.....	56
5.1.3.8.1	[BSW005211] Hardware Checking Function on CC.....	56
5.1.3.8.2	[BSW005222] Support of Transceiver Rx-Only Mode via FR State Manager .....	56
5.1.4	Transport Layer FlexRay .....	57
5.1.4.1	Configuration .....	57
5.1.4.1.1	[BSW05077] Unique Identifier of N-SDU .....	57
5.1.4.1.2	[BSW05079] Transport Connection Properties .....	57
5.1.4.2	Initialization.....	58
5.1.4.2.1	[BSW05088] FlexRay Transport Layer Initialization.....	58
5.1.4.2.2	[BSW05089] FlexRay Transport Layer Availability.....	58
5.1.4.3	Normal Operation .....	59
5.1.4.3.1	[BSW05090] Support of Optional ISO10681-2 Service.....	59
5.1.4.3.2	[BSW05093] Transmit Cancellation .....	59
5.1.4.3.3	[BSW05095] Bandwidth Control .....	59
5.1.4.4	Fault Operation.....	60
5.1.4.4.1	[BSW05129] Mismatch of Service Call and Connection Properties .....	60
5.1.5	FlexRay Time Service .....	60
5.1.5.1	General Requirements .....	60
5.1.5.1.1	[BSW05033] Tick Conversion .....	60

5.1.5.1.2	[BSW05053] Cluster External Clock Synchronization .....	61
5.1.5.1.3	[BSW05156] Controller External Clock Synchronization.....	61
5.1.5.2	Configuration .....	61
5.1.5.2.1	[BSW05044] Set Absolute Timer .....	61
5.1.5.3	Normal Operation .....	62
5.1.5.3.1	[BSW05046] Enable Absolute Alarms.....	62
5.1.5.3.2	[BSW05047] Disable Absolute Alarms.....	62
5.1.5.3.3	[BSW05048] Acknowledge Absolute Alarms .....	63
5.1.5.3.4	[BSW05049] Enable Relative Alarms.....	63
5.1.5.3.5	[BSW05050] Disable Relative Alarms.....	63
5.1.5.3.6	[BSW05051] Acknowledge Relative Alarms .....	64
5.1.5.3.7	[BSW05052] Get Cycle Length in Macroticks .....	64
5.1.5.3.8	[BSW05019] Get FlexRay Global Time.....	64
5.1.5.4	Fault Operation.....	65
5.1.5.4.1	[BSW05071] FlexRay Time Services Access if CC is in Halt.....	65
5.1.5.4.2	[BSW05072] FlexRay Time Services Access if CC is Out of Sync	65
5.1.6	FlexRay Transceiver Driver .....	66
5.1.6.1	Configuration .....	66
5.1.6.1.1	[BSW05131] Configuration Data for FlexRay Transceiver .....	66
5.1.6.1.2	[BSW05132] Support for More than One FlexRay Transceiver ...	66
5.1.6.1.3	[BSW05133] Configuration of Bus Operation Mode after Initialization for each FlexRay Transceiver .....	67
5.1.6.1.4	[BSW05134] Initialization Sequence for FlexRay Transceiver Driver .....	67
5.1.6.1.5	[BSW05136] Configuration "Notification for Wake-up by Bus" .....	67
5.1.6.2	Initialization.....	68
5.1.6.2.1	[BSW05137] Initialize the FlexRay Transceiver Driver.....	68
5.1.6.3	Normal Operation .....	69
5.1.6.3.1	[BSW05138] FlexRay Transceiver Driver API shall be Synchronous.....	69
5.1.6.3.2	[BSW05166] Set FlexRay Transceiver Operation Mode .....	69
5.1.6.3.3	[BSW05167] Get FlexRay Transceiver Operation Mode.....	69
5.1.6.3.4	[BSW05144] Get FlexRay Transceiver Wake-up Reason.....	70
5.1.6.3.5	[BSW05147] Notification for Wake-up by Bus.....	71
5.1.6.3.6	[BSW05148] Support for Wake-up During Sleep Transition .....	71
5.1.6.3.7	[BSW05149] Support API to Enable/Disable and Clear Wake-up Events.....	72
5.1.6.3.8	[BSW005135] Support for active FlexRay Stars .....	72
5.1.6.4	Shutdown Operation.....	73
5.1.6.4.1	[BSW05150] Safe System Shutdown for FlexRay Transceiver Driver .....	73
5.1.6.5	Fault Operation.....	73
5.1.6.5.1	[BSW05151] FlexRay Transceiver Driver Must Check Transceiver Control.....	73
5.1.6.5.2	[BSW05168] Indicate FlexRay Transceiver Error State (modify according to Monitoring Concept and Concept Reliability).....	74
5.1.6.5.3	[BSW005212] Detect Error Information in Bus Driver .....	74
5.1.6.5.4	[BSW005213] Hardware Checking Function on Bus Driver .....	74
5.1.6.5.5	[BSW005214] Hardware Resetting Function on Bus Driver.....	75
5.2	Non-Functional Requirements.....	75
5.2.1	FlexRay Interface .....	75

5.2.1.1	Abstraction Requirements .....	75
5.2.1.1.1	[BSW05006] Abstraction of FlexRay-Specific Features .....	75
5.2.1.2	Resource Usage.....	75
5.2.1.2.1	[BSW05009] Local Memory Space Usage.....	75
5.2.1.3	Configuration .....	76
5.2.1.3.1	[BSW05056] Configuration of the FlexRay Interface at System Configuration Time.....	76
5.2.2	Transport Layer FlexRay .....	76
5.2.2.1	[BSW05098] Usage of ISO 15765-2 and ISO 15765-4 Specifications.....	76
5.2.2.2	[BSW05073] Usage of ISO 10681-2 Specifications.....	77
5.2.2.3	[BSW05074] FlexRay Transport Layer Interfaces .....	77
5.2.2.4	[BSW05075] Independence of the Network Configuration .....	77
5.2.2.5	[BSW05123] Configuration Modifiable by a Flashing Process.....	78
5.2.2.6	[BSW05076] Multiple Logical FlexRay Transport Layer Channels ...	78
5.2.3	FlexRay Transceiver Driver .....	78
5.2.3.1.1	Timing Requirements.....	79
5.2.3.1.1.1	[BSW05152] Transceiver-specific Timing Requirements....	79
5.3	[BSW999999999] Not Applicable Requirements.....	79
6	References .....	80
6.1	Deliverables of AUTOSAR .....	80
6.2	Related Standards and Norms .....	80
6.2.1	FlexRay .....	80
6.2.2	ISO .....	80



## Known Limitations

- No requirements on FlexRay StateManager listed in this document.

## 1 Scope of document

The goal of this document is to define to what extent elements of the basic software have to be configurable and what preliminaries they have to comply with to meet the tailoring requirements.

As far as possible, the set of basic software elements should consist of already existing elements of modules of automotive software. Only in case of "good reasons" new elements of basic software should be part of the set.

If such the definition of these new elements is not part of this work package. Nevertheless the information about basic software elements additionally required has to be given to related work groups.

### Constraints and restrictions

The first scope for specification of requirements on basic software modules is systems which are not safety relevant. For this reason safety requirements are assigned to medium priority.

Data transmission accesses to the FlexRay Interface while FlexRay Communication Controllers are out of sync or in shutdown state shall not return an error.

The FlexRay Transport Layer shall only support half-duplex communication sessions (If both half-duplex and full-duplex session were possible in one implementation, the need for resources and runtime would be increased significantly).

Redundancy (in the time or spatial domain) is not supported by the FlexRay Interface; redundancy has to be explicitly modeled on a higher BSW layer.

There are functional restrictions imposed by the FlexRay Protocol Specification [[FR\\_PROT\\_SPEC](#)] concerning Wake-Up and Start-up. In both cases, it is not guaranteed by the FlexRay Protocol Specification that the intended result will be reached, i.e. in case of:

- Start-up: It is not guaranteed that the start-up process will be successful, i.e. after initiating a start-up, it is not guaranteed that the FlexRay Communication Controller will successfully establish a communication on the Cluster, or integrate into an ongoing communication, resp.
- Wake-up: It is not guaranteed that the wake-up process will be successful, i.e. after the FlexRay Communication Controller has sent a wake-up pattern on a specific FlexRay Channel, it is not guaranteed that all other FlexRay Communication Controllers connected to this Channel actually do wake up.

A higher AUTOSAR BSW module has to provide a supervision mechanism to ensure that the intended result of these operations will be reached. This mechanism is supported by features requested by the FlexRay Software Requirements Specification documents in hand.

## 2 How to read this document

Each requirement has its unique identifier starting with the prefix "BSW" (for "Basic Software"). For any review annotations, remarks or questions, please refer to this unique ID rather than chapter or page numbers!

### 2.1 Conventions used

In requirements, the following specific semantics are used (taken from Request for Comment RFC 2119 from the Internet Engineering Task Force IETF)

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119. Note that the requirement level of the document in which they are used modifies the force of these words.

- **MUST:** This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.
- **MUST NOT:** This phrase, or the phrase „SHALL NOT“, means that the definition is an absolute prohibition of the specification.
- **SHOULD:** This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- **SHOULD NOT:** This phrase, or the phrase "NOT RECOMMENDED" mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
- **MAY:** This word, or the adjective „OPTIONAL“, means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation, which does not include a particular option, **MUST** be prepared to interoperate with another implementation, which does include the option, though perhaps with reduced functionality. In the same vein an implementation, which does include a particular option, **MUST** be prepared to interoperate with another implementation, which does not include the option (except, of course, for the feature the option provides.)

### 2.2 Requirement structure

Each module-specific chapter contains a short functional description of the Basic Software Module. Requirements of the same kind within each chapter are grouped under the following headlines (where applicable):

## Functional Requirements:

- Configuration (Which elements of the module need to be configurable?)
- Initialization
- Normal Operation
- Shutdown Operation
- Fault Operation
- ...

## Non-Functional Requirements:

- Timing Requirements
- Resource Usage
- Usability
- Dependencies on other SW modules (e.g. Description Templates, Tooling, ...)
- ...

Not each requirement in this SRS does necessarily map to exactly one API call of the respective AUTOSAR FlexRay SW module. It might be possible that one API call fulfills multiple requirements.

It might also be possible that some requirements address internal features of the AUTOSAR FlexRay SW modules and do not require an API call at all in order to be fulfilled.

Implementations of the AUTOSAR FlexRay SW modules used in safety-critical environments might only be allowed to fulfill a subset of the requirements described herein.

Apart from this safety-critical case, every requirement listed in this document is mandatory, i.e. each implementation of an AUTOSAR FlexRay SW module **MUST** fulfill all respective requirements in this document so if that SW module is being used with a hardware supporting the full feature set, the full functionality of the AUTOSAR FlexRay SW module is available.

However, if the underlying hardware does NOT support a specific feature, the AUTOSAR FlexRay SW modules do NOT have to emulate this functionality if that requirement is explicitly marked as "**if supported by the FlexRay CC**".

If a higher layer BSW module calls an API of the AUTOSAR FlexRay SW modules that operates on an optional feature of the FlexRay Specification, and this feature is not supported in the system in question, the respective FlexRay SW modules shall raise an error in debug mode.

In general, the FlexRay Interface has no knowledge of the origin of a PDU passed to it in an API call.

Therefore, throughout this document, unless stated otherwise, the term "PDU" is being used for PDUs originating or sent to:

- AUTOSAR COM (I-PDU), or

- FlexRay TP (N-PDU), or
- FlexRay NM

Throughout this document the term "**configuration**" is being used in several requirements. Unless otherwise noted, this term refers to the configuration of the respective software module that affects the **runtime behavior** but ***not*** the generated **executable code** of this software module. The most prominent case of such a configuration is the FlexRay communication schedule. Enabling the development error handling in the executable code is explicitly not meant.

Accordingly, the term "configuration data" in general refers to pure data (representing a certain configuration) and not to any executable code.

Throughout this document, several scenarios for **changing configuration data** are mentioned. They are being used as follows:

- "**pre compile time**" = carried out *before* compiling the code of the FlexRay Interface/Driver, since the code generation might or will depend on this setting.
- "**at system configuration time**" = static configuration parameters stored in the FlexRay Interface/Driver; may be defined *after* compilation of the code of the FlexRay Interface/Driver, but have to be defined *before* the first execution of the FlexRay Interface/Driver code.
- "**by a flashing process**" = data (not code!) manipulation carried out in a *flashing process* of a flashable memory (in general a Flash-EEPROM) e.g. in a garage, but *not* while the car is being driven. Usually used to *replace* a static configuration *already stored* in the ECU, or a part thereof.
- "**during runtime**" = *dynamically switching* (in normal active mode of the FlexRay CC, if supported by the FlexRay CC) between different configuration parameter sets *statically stored* in the FlexRay Interface.

### 3 Acronyms and abbreviations

The following acronyms and abbreviations are being used throughout this document

<b>Acronym:</b>	<b>Description:</b>
SW	Software
BSW	Basic Software
DEM	Diagnostic Event Manager BSW module
ComM	CommunicationManager BSW module
CC	(FlexRay) Communication Controller

<b>Abbreviation:</b>	<b>Description:</b>
i.e.	[lat.] id est = [eng.] that is
e.g.	[lat.] exempli gratia = [eng.] for example

## 4 Functional Overview

### 4.1 Common Requirements (FlexRay Interface and FlexRay Driver)

- Support of the FlexRay Protocol 2.1 specification
- Support of the FlexRay Protocol 3.0 specification, for details see chapter 5.2
- Abstraction of FlexRay specific features
- Abstraction of FlexRay Controller type specific features
- Support of at least 4 (possibly different) FlexRay Controllers per ECU with up to 2 Channels each and data rates up to 10 MBit/s
- Connect the communication interface to the specific FlexRay Controller
- Transmit and receive FlexRay Frames in the static and in the dynamic segment
- Support of multiple FlexRay Clusters
- Support of (with respect to the FlexRay global time) asynchronous and synchronous operation of the communication stack
- Packing and unpacking of multiple PDU into/out of one or more FlexRay Frames
- Interface: PDU-based API to upper software layer
- Single/multiple sender Cycle Multiplexing for scalable efficient bandwidth exploitation
- Configuration of the complete FlexRay system [at system configuration time](#)
- Run time reconfiguration of FlexRay Controllers
- Provision of basic network management functions: Transmit/detect wake-up patterns; start-up, shutdown, sleep mode, start and stop communication
- Provision of the FlexRay Controller state e.g. for usage by FlexRay NM
- Provision of FlexRay bus errors

For an overview of the structure of AUTOSAR software modules please see the current AUTOSAR Layered Software Architecture Specification [[LSA](#)].

### 4.2 FlexRay Interface

The FlexRay Interface shall provide a standardized interface to access the FlexRay Communication system/hardware. The FlexRay Interface shall be independent of the specific FlexRay CC(s) being used and its/their access through the FlexRay Driver(s). The FlexRay Interface shall give access to one or several FlexRay Drivers via one uniform interface.

### 4.3 FlexRay Driver

The FlexRay Driver module shall offer to the FlexRay Interface Module, the user of the API, a uniform interface to access a number of FlexRay Communication Controllers, usually of the same type. The FlexRay Driver is a software layer which maps abstract functional requests to sequences of CC specific hardware accesses. The hardware implementation of a CC will be hidden from the FlexRay Interface; e. g. the number of transmit/receive buffers will not be visible outside a FlexRay Driver.

### 4.4 Transport Layer FlexRay

The FlexRay Transport Layer supports the protocol handling according ISO 10681-2. Hence the FlexRay Transport Layer provides support for

- a) unsegmented unacknowledged 1:1 communication
- b) segmented unacknowledged 1:1 communication
- c) unsegmented acknowledged 1:1 communication
- d) segmented acknowledged 1:1 communication
- e) unsegmented unacknowledged 1:n communication

### 4.5 FlexRay Time Service

Provides FlexRay time services, synchronization to FlexRay global time, interrupt services based on FlexRay global time, and synchronization of FlexRay Clusters to external time references

### 4.6 FlexRay Transceiver Driver

The FlexRay Transceiver Driver is responsible to handle the FlexRay transceivers on an ECU according to the expected state of the bus specific NM.

The FlexRay Transceiver is a hardware device, which mainly transforms the logical I/O signals of the FlexRay Communication Controller's ports to the bus compliant electrical levels, currents and timings. Within an automotive environment, there is currently only one physical layer specified for FlexRay.

In addition, the transceivers are often able to detect electrical malfunctions like wiring issues, ground offsets or collisions. Depending on the interface, they are capable of flagging the detected errors by a single port pin or very detailed via SPI.

Some transceivers also support power supply control and wake-up via the bus. A lot of different wake-up/sleep and power supply concepts are available on the market with focus on the best cost-optimized solution for a given task.



Latest developments are so called System Basis Chips (SBC) where not only the FlexRay transceivers but also power-supply control and advanced watchdogs are implemented in one housing and are controlled via one interface (e.g. via SPI).

A typical FlexRay Transceiver device is the TJA1080 device for a 10 MBit FlexRay bus with support for wake-up via the bus.

#### 4.6.1 Transceiver Operation Modes

Important transceiver operation modes are:

- **Un-powered:** Neither communication nor wake-up is possible
- **Normal:** Full communication is possible
- **Read Only:** Transmission is inhibited, reception is possible
- **Standby:** No communication is possible but ECU is still powered. In addition, a transition to Sleep is only valid from this mode. Wake-up by bus is possible, too.
- **Sleep:** No communication is possible and ECU may be un-powered. Wake-up by bus is possible, too.

#### 4.6.2 Transceiver Error Status

In parallel to the transceiver operation modes, the transceiver provides status information for the bus. The status is only relevant for operation mode Normal:

- **Good:** No error, bus physics work correctly without any drawbacks.
- **Error:** The transceiver has detected a serious error on the bus which does not allow further communication.

#### 4.6.3 Transceiver Wake-up Reason

The transceiver driver is able to store the local view on who has requested the wake-up:

- **Remote wake-up event:** A remote wake-up event is the reception of at least two consecutive wake-up symbols via the bus.
- **Local wake-up event:** A pulse on the FlexRay Transceiver device's WAKE pin (if present) has caused the wake-up.
- **Power on wake-up:** The FlexRay Transceiver device has become sufficiently supplied after being not powered.
- **No wake-up:** No wake-up has been detected.

#### 4.6.4 Remarks to the FlexRay Transceiver Driver

FlexRay Transceivers are very different in their behavior and supported features. The range starts with very simple FlexRay Transceivers, which are "always on", includes transceivers with support for advanced limp-home-handling and error detection and ends with so called system basis chips (SBC) which internally contain multiple FlexRay Transceivers, watchdog, voltage regulators and more.

The target of this document is to specify interfaces and behavior, which are applicable to most current and future FlexRay Transceivers on the market for nearly all use cases. The target is that the "user" of the transceiver functionality, typically the AUTOSAR CommunicationManager (ComM) and the AUTOSAR ECU State Manager (EcuM), are bus independent and therefore reusable.

It will not be possible to cover all possible combinations of FlexRay Transceivers with all conceivable power concepts within one AUTOSAR implementation.

#### **4.6.5 Explicitly uncovered FlexRay Transceiver Functionality**

Some FlexRay Transceivers offer additional functionality to improve e.g. ECU self-test or enhanced error detection capability for diagnostics.

ECU self-test and enhanced error detection are not defined within AUTOSAR and requiring such functionality in general will lock out most currently used (and cheap) transceiver devices. Therefore, features like "ground shift detection", "selective wake-up", "slope control" and others are not supported within these requirements. A general and "open" API like IOControl() is not applicable (and accepted) within AUTOSAR due to portability and reuse.

#### **4.6.6 System Basis Chip and FlexRay Transceiver Driver**

A system basis chip (SBC) contains, beside the FlexRay Transceivers, additional hardware related to power control and safety (e.g. multiple voltage regulators and a watchdog) and even more features (e.g. persistent memory).

In the AUTOSAR concept, a separate manager/driver/handler (in AUTOSAR called: Interface) is responsible for each identified hardware device. Therefore, the additional manager/driver/handler covers the functionality inside a SBC beside the transceiver driver (e.g. Watchdog Manager, non-volatile memory manager, power control driver, etc.). Due to the shared communication access and the (security-related) restrictions within this communication, independent handling of each SBC-sub-functionality will not be possible.

This will lead to the situation that either a SBC could not be used within an AUTOSAR compliant ECU or (the better solution) a specialized manager/driver/handler for the SBC functionality with all APIs of each single domain has to be used.

## 5 Requirement Specification

### 5.1 Functional Requirements

#### 5.1.1 Compatibility overview on supported FlexRay CC features

Feature	FlexRay 2.1	FlexRay 3.0
NM Vector update	supported	added: early update
WUDOP in SymbolWindow	N/A	not supported
POC: deferred ready	N/A	not supported
POC: deferred halt	N/A	supported
FIFO support	supported	supported
2 <sup>nd</sup> absolute timer	N/A	supported
Message ID filtering	supported	supported
TT-E, TT-M	N/A	supported
repetition cycle filtering values	1,2,4,8,16,32,64	1,2,4,5,8,10,16,20,32,40,50,64
Static slot multiplexing	N/A	supported
Baud rate	10	2.5, 5, 10
number of valid start-up frames	N/A	supported

#### 5.1.1.1 General requirements

##### 5.1.1.1.1 [BSW05000] Support of Synchronous SW Modules

<b>ID:</b>	BSW05000
<b>Initiator:</b>	BMW
<b>Date:</b>	23.07.2004
<b>Short Description:</b>	Support of applications (or software modules) running synchronously to the FlexRay global time.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay-related BSW modules of the AUTOSAR communication stack shall support applications (or software modules) running synchronously to the FlexRay global time.
<b>Rationale:</b>	<p>An application (or software module) running synchronously to the FlexRay global time (i.e. it is driven by the FlexRay global time) shall be able to efficiently communicate via FlexRay, since an application (or software module) running synchronously to the FlexRay global time allows an optimal adaptation of the data processing schedule to the schedule of the data transmission via FlexRay.</p> <p>The synchronisation point of a sw module with FR can be set on each Macrotick within the flexraycycle</p>
<b>Use Case:</b>	<p>For some applications (e.g. chassis function as regulator) it is advantageous to run the application synchronously to sensors and actuators connected to the FlexRay bus, hence to the FlexRay global time.</p> <p>It shall be possible to start an SW Module at a specific Flexray time within the Flexray cycle</p>
<b>Dependencies:</b>	<a href="#">[BSW05001]</a> Support of Asynchronous SW Modules
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--

<b>Contributes to:</b>	--
------------------------	----

#### 5.1.1.1.2 [BSW05001] Support of Asynchronous SW Modules

<b>ID:</b>	BSW05001
<b>Initiator:</b>	BMW
<b>Date:</b>	23.07.2004
<b>Short Description:</b>	Support of applications (or software modules) running asynchronously to the FlexRay global time.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay-related BSW modules of the AUTOSAR communication stack shall support applications (or software modules) running asynchronously to the FlexRay global time.
<b>Rationale:</b>	An application (or software module) running asynchronously to the FlexRay global time (i.e. it has no "knowledge" about the FlexRay global time) shall be able to communicate via FlexRay, since an application (or software module) running asynchronously to the FlexRay global time allows a flexible software design.
<b>Use Case:</b>	For some applications (e.g. body or gateway) it is advantageous to run the application event-triggered, hence asynchronously to the FlexRay global time.
<b>Dependencies:</b>	[BSW05000] Support of Synchronous SW Modules
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.1.1.3 [BSW05002] FlexRay Interface and FlexRay Driver as Only Necessarily Synchronous SW Modules

<b>ID:</b>	BSW05002
<b>Initiator:</b>	BMW
<b>Date:</b>	23.07.2004
<b>Short Description:</b>	Support of completely asynchronous usage of the synchronously running FlexRay-related BSW modules by all upper-layer SW modules.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay-related BSW modules of the AUTOSAR communication stack shall encapsulate all synchronous services to enable a completely asynchronous usage of the FlexRay-related BSW modules by all upper-layer SW modules. Therefore, the FlexRay Interface shall operate synchronized to the FlexRay global time(s) of all involved FlexRay Communication Controllers by using services of the respective FlexRay Driver.
<b>Rationale:</b>	<p>The FlexRay Interface and the FlexRay Driver shall be synchronous to the FlexRay bus to support synchronous communication and to avoid buffering and queuing. Also, each upper-layer SW module shall have the possibility to run asynchronously to the FlexRay global time.</p> <p>If [BSW05001] applies to all other SW modules (i.e. all upper-layer software modules run completely asynchronously to any FlexRay global time), the FlexRay Interface and the FlexRay Driver(s) it controls are the only synchronous SW modules.</p> <p>Note that (for example in case of asynchronous gateways) the global time of multiple FlexRay Controllers in a single ECU might be different, since they are connected to different FlexRay Clusters. In this case, the respective operations for each single Communication Controller are to be performed synchronously to this Communication Controller's global time.</p>
<b>Use Case:</b>	An ECU with all SW modules running asynchronously to the FlexRay global

	time.
<b>Dependencies:</b>	[BSW05000] Support of Synchronous SW Modules [BSW05001] Support of Asynchronous SW Modules
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

**5.1.1.1.4 [BSW05003] Support of Slot/Cycle Multiplexing**

<b>ID:</b>	BSW05003
<b>Initiator:</b>	BMW
<b>Date:</b>	23.07.2004
<b>Short Description:</b>	Support of the FlexRay Slot/Cycle Multiplexing mechanism.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	<p>The FlexRay-related BSW modules of the AUTOSAR communication stack shall support the FlexRay Slot/Cycle Multiplexing mechanism which is used to use a FlexRay Slot on a Channel more efficiently by alternating the Frames being sent in this Slot from Cycle to Cycle.</p> <p>In the static segment, the FlexRay Slot/Cycle Multiplexing mechanism allows an ECU to transmit different Frame contents in the same Slot in different Cycles. This is called "Single Sender Slot Multiplexing".</p> <p>In the dynamic segment, the FlexRay Cycle Multiplexing mechanism allows several ECUs to share the same Slot by uniquely assigning certain Cycles (to be accurate: FlexRay Cells of the same Slot) to each ECU. This is called "Multiple Sender Slot Multiplexing".</p> <p>Both kinds of Slot/Cycle Multiplexing mechanisms shall be supported by the FlexRay-related BSW modules.</p>
<b>Rationale:</b>	Optimal uses of the FlexRay bandwidth can only be reached with an extensive and effective use of the Cycle Multiplexing mechanism.
<b>Use Case:</b>	<p>In the following example, an ECU sends two different Frame contents in the same Slot and thus effectively uses this resource.</p> <p style="text-align: center;"><b>FlexRay Communication Matrix with Slot Multiplexing</b></p> <p style="text-align: center;">Communication Cycle</p> <p> <span style="display: inline-block; width: 10px; height: 10px; background-color: #FFC0CB; border: 1px solid black; margin-right: 5px;"></span> L-PDU-Identifier: Slot: 5 Channel: A Base Cycle: 0 Cycle Repetition: 2<sup>1</sup> </p> <p> <span style="display: inline-block; width: 10px; height: 10px; background-color: #ADD8E6; border: 1px solid black; margin-right: 5px;"></span> L-PDU-Identifier: Slot: 5 Channel: A&amp;B Base Cycle: 3 Cycle Repetition: 2<sup>2</sup> </p>
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--

<b>Supporting Material:</b>	<a href="#">[FR_PROT_SPEC]</a>
<b>Contributes to:</b>	--

#### 5.1.1.1.5 [BSW05169] Avoid Timer Interrupts during Start-up

<b>ID:</b>	BSW05169
<b>Initiator:</b>	BMW /
<b>Date:</b>	13.09.2004
<b>Short Description:</b>	Avoid FlexRay Communication Controller timer interrupts during the start-up phase of the ECU.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay-related BSW modules of the AUTOSAR communication stack shall ensure that no timer interrupt raised from a FlexRay Communication Controller will be forwarded to other software modules during the start-up phase of the ECU.
<b>Rationale:</b>	Interrupts occurring before the entire ECU software has been initialized may cause problems.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05055]</a> Avoid Timer Interrupts during Shutdown
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.1.1.6 [BSW05055] Avoid Timer Interrupts during Shutdown

<b>ID:</b>	BSW05055
<b>Initiator:</b>	BMW /
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Avoid FlexRay Communication Controller timer interrupts during the shutdown phase of the ECU.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay-related BSW modules of the AUTOSAR communication stack shall ensure that no timer interrupt raised from a FlexRay Communication Controller will be forwarded to other software modules during the shutdown phase of the ECU.
<b>Rationale:</b>	Interrupts occurring while the ECU software is being shut down may cause problems.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05169]</a> Avoid Timer Interrupts during Start-up
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.1.1.7 [BSW005216] API to query the FlexRay Rate and Offset Correction

<b>ID:</b>	BSW005216
<b>Initiator:</b>	DAIMLER
<b>Date:</b>	29.05.2007
<b>Short Description:</b>	API to query the FlexRay rate and offset correction
<b>Importance:</b>	high   medium   <b>low</b>
<b>Description:</b>	<ol style="list-style-type: none"> <li>1) It shall be possible to query rate and offset correction of a specific FlexRay Communication Controller calculated by the clock synchronization algorithm</li> <li>2) It shall be possible to send rate and offset correction on the bus</li> </ol>
<b>Rationale:</b>	It could be useful to know the current drift rate and the offset of every

	<p>FlexRay controller.</p> <p>The new feature is needed to fulfill the safety requirement OSR107 AUTOSAR shall provide ECU hardware monitoring and testing to detect faults in the following components:</p> <ul style="list-style-type: none"> <li>• CPU</li> <li>• memory</li> <li>• peripheral devices</li> <li>• communication components</li> <li>• address, data and control buses</li> </ul>
<b>Use Case:</b>	<p>The monitoring of 'drift' of dedicated FlexRay controllers could shorten the error analysis. The monitoring of 'drift' of dedicated FlexRay controllers could allow the prediction of hardware errors.</p>
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00087 BRF00265

#### 5.1.1.1.8 [BSW005217] API Report of FlexRay Status Data for number of Sync Frames

<b>ID:</b>	BSW005217
<b>Initiator:</b>	GM
<b>Date:</b>	31.01.2008
<b>Short Description:</b>	Report of status data number of Sync Frames
<b>Importance:</b>	high   medium   low
<b>Description:</b>	Norm
<b>Rationale:</b>	Status data contains number of sync frames received or transmitted for each channel.
<b>Use Case:</b>	Necessary for Diagnostic purposes to detect persistent asymmetry between applied clock correction terms of different nodes. Useful to set DTC. Allows to service the nodes that are missing.
<b>Dependencies:</b>	Error Handling, DET, DCM, FlexRay Driver, FlexRay Interface.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00267 BRF00273

#### 5.1.1.1.9 [BSW005218] Report of FlexRay Status Data for list of Sync Frame IDs

<b>ID:</b>	BSW005218
<b>Initiator:</b>	GM
<b>Date:</b>	31.01.2008
<b>Short Description:</b>	Report of status data for list of Sync Frame IDs
<b>Importance:</b>	high   medium   low
<b>Description:</b>	Norm
<b>Rationale:</b>	Status data contains list of Sync Frame IDs received or transmitted for each channel.
<b>Use Case:</b>	Necessary for Diagnostic purposes to detect persistent asymmetry between applied clock correction terms of different nodes. Useful to set DTC. Allows to service the nodes that are missing.
<b>Dependencies:</b>	Error Handling, DET, DCM, FlexRay Driver, FlexRay Interface.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00299 BRF00273

#### 5.1.1.1.10 [BSW005219] Support for FlexRay Key Slot Mode

<b>ID:</b>	BSW005219
<b>Initiator:</b>	GM
<b>Date:</b>	31.01.2008
<b>Short Description:</b>	Support for FlexRay Key Slot Mode
<b>Importance:</b>	high   medium   low
<b>Description:</b>	<p>FlexRay includes a "Key Slot Mode" mode that can be configured for use as an extension of the startup process. During integration all FlexRay nodes only transmit one frame in the static segment (identified as the "key slot"). They do this to minimize the number of frames they might corrupt if their timing is incorrect and they transmit in the slots of other nodes. Once they integrate there are two possibilities.</p> <p>Nodes that do NOT use the Key Slot Mode allow their other frames (static and dynamic) to be automatically enabled as soon as they integrate. The underlying assumption is that the mere fact that they integrated is sufficient confirmation of their conformance to the schedule to allow these transmissions to be enabled.</p> <p>Nodes that use the Key Slot Mode must enable the transmission of their other frames with an explicit host command. The underlying assumption in this case is that the host will first perform some sort of explicit confirmation that its transmitted frames are properly timed before it executes the command. This covers the scenario where only the transmission timing is flawed - in a way that does not impact the ability to integrate. The likely way to perform this confirmation is to configure another node to explicitly confirm the timing of the Key Slot frame(s) with a flag in a frame that it transmits. It simply sends a specified frame with the confirmation flag set if it receives the single frame from the node whose timing it is supervising. If the host sees receives this confirmation flag, it issues the command to the controller to disable Key Slot Mode and this causes all frames to be enabled.</p>
<b>Rationale:</b>	FlexRay provides support for Key Slot Mode.
<b>Use Case:</b>	<p>Active NM only after controller transitions from Key Slot Mode to normal communication (i.e., all slots are activated).</p> <p>Key Slot Mode is used for Diagnostics to limit transmission of all slots until node is sure that it is not interfering with transmission of normal communication messages by other nodes, i.e., the node transitions from Key Slot Mode when it is sure that it is in a compatible system mode.</p>
<b>Dependencies:</b>	Mode management, COM, Vehicle Mode Management, FlexRay State Manager
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00268

#### 5.1.1.1.11 [BSW005223] Get number of Startup Frames

<b>ID:</b>	BSW005223
<b>Initiator:</b>	DAIMLER
<b>Date:</b>	29.05.2007
<b>Short Description:</b>	API to get number of available startup frames
<b>Importance:</b>	high   medium   low
<b>Description:</b>	It shall be possible to detect the situation that no startup frames are present but the FlexRay is still operating because of still available sync frames.
<b>Rationale:</b>	<p>Every network consists of at least 2 Coldstart nodes and these coldstart nodes send startup frames. All Coldstart nodes are shut down if no startup frame is present thus it is impossible for any node to integrate himself into the network without a previous shutdown of all nodes.</p> <p>The indication could be used to "force a shutdown of all nodes" to allow for a</p>



	restart with all nodes.
<b>Use Case:</b>	A network with 3 coldstart nodes and 2 sync nodes. If the 3 coldstart nodes fall asleep/perform a restart because of an error (e.g. reset or low voltage) it is necessary to restart the FlexRay to allow the reintegration of all FlexRay nodes.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.1.1.12 [BSW005220] Immediate Buffer Read Access

<b>ID:</b>	BSW005220
<b>Initiator:</b>	Bosch
<b>Date:</b>	17.12.2008
<b>Short Description:</b>	The FlexRay related BSW modules of the AUTOSAR communication stack shall provide a method to trigger the Job List Execution by a synchronized OS-task directly, instead of using an interrupt.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay related BSW modules of the AUTOSAR communication stack shall provide a method to allow immediate buffer access for read operations.
<b>Rationale:</b>	If a PDU is received by the communication controller in every cycle, but where the task is only acting on it occasionally, it does not make sense, that the FlexRay Driver and Interface transfers this PDU every time, but only on request.
<b>Use Case:</b>	To introduce functionality for direct read access to the CC is especially useful, if the FlexRay is solely used as a high-bandwidth replacement for an event-driven protocol, e.g. CAN.
<b>Dependencies:</b>	Unknown
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.1.1.13 [BSW005225] Support of CHI Command AllowColdstart

<b>ID:</b>	BSW005225
<b>Initiator:</b>	BMW
<b>Date:</b>	17.12.2008
<b>Short Description:</b>	Support of CHI Command "AllowColdstart"
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	It shall be possible to call the CHI Command at an arbitrary point in time
<b>Rationale:</b>	Influencing factor of the start-up behavior of a FlexRay Cluster
<b>Use Case:</b>	The start-up time of an ECU can be taken into consideration to ensure that start-up attempts are not missed on receiver side.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.1.1.14 [BSW005224] Support of Payload Preamble Indicator Bit

<b>ID:</b>	BSW005224
<b>Initiator:</b>	DAIMLER
<b>Date:</b>	17.12.2008
<b>Short Description:</b>	Support of Payload Preamble Indicator Bit

<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	It shall be possible to switch the Payload Preamble Indicator Bit
<b>Rationale:</b>	The payload preamble indicator indicates whether or not an optional vector is contained within the payload segment of the frame transmitted
<b>Use Case:</b>	<p>If the frame is transmitted in the static segment the payload preamble indicator indicates the presence of a network management vector at the beginning of the payload.</p> <p>If the frame is transmitted in the dynamic segment the payload preamble indicator indicates the presence of a message ID at the beginning of the payload. If the frame is transmitted in the static segment the payload preamble indicator indicates the presence of a network management vector at the beginning of the payload.</p>
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.1.1.15 [BSW005221] Synchronization of FlexRay stack to the FlexRay cycle

<b>ID:</b>	BSW005221
<b>Initiator:</b>	Bosch
<b>Date:</b>	17.12.2008
<b>Short Description:</b>	The FlexRay related BSW modules of the AUTOSAR communication stack shall provide a method to trigger the Job List Execution by a synchronized OS-task directly, instead of using an interrupt.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	<p>The FlexRay related BSW modules of the AUTOSAR communication stack shall provide a method to trigger the functionality, usually implemented in the interrupt service routine by an OS-task, instead of using the FlexRay interrupt.</p> <p>Therefore, if the OS provides a method to synchronize a task to the FlexRay cycle, e.g. by using a Synchronized Rate Monotonous Scheduler, it can trigger the FlexRay protocol stack without wasting crucial interrupt resources.</p>
<b>Rationale:</b>	<p>AUTOSAR allows letting event-driven tasks act on the time-triggered FlexRay system. This is done by synchronizing the FlexRay stack to the FlexRay cycle (FlexRay Interrupt Service Routine, JobList, ...).</p> <p>In future systems, not only the FlexRay stack, but also the applications will be synchronized to use all advantages of time-triggered systems. When this happens, all transmit and receive operations by the task will already be synchronized to the FlexRay cycle. The internal synchronization mechanism of the FlexRay stack is then not necessary any more. The resource consumption (Memory, Processing Power, Runtime, Application jitter through interrupt load), currently a significant factor, can be dramatically reduced, when the Job List Execution can be triggered by a synchronized task directly.</p> <p>Therefore the synchronization using an interrupt service routine, as currently required by the AUTOSAR FlexRay documents (Requirements, Interface, Driver, ...), shall not remain the only solution. There shall be an option to provide the synchronization using a synchronized OS-task.</p>
<b>Use Case:</b>	Systems, where not only the FlexRay stack, but also the applications are synchronized. The method will reduce interrupt resource consumption.
<b>Dependencies:</b>	Unknown
<b>Conflicts:</b>	--

<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

## 5.1.2 FlexRay Interface

### 5.1.2.1 General Requirement

#### 5.1.2.1.1 [BSW05004] PDU-Based Data API

<b>ID:</b>	BSW05004
<b>Initiator:</b>	BMW
<b>Date:</b>	23.07.2004
<b>Short Description:</b>	PDU-based data API to all upper layers.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a PDU-based data API to all upper layers.
<b>Rationale:</b>	<p>A PDU-based data API is needed in order to fulfill <a href="#">[BSW05003]</a> in an optimal way.</p> <p>In order to fulfill <a href="#">[BSW05002]</a>, the FlexRay Interface needs to abstract the synchronous features of the FlexRay protocol.</p> <p>In order to fulfill <a href="#">[BSW05003]</a>, a FlexRay Slot-based API is not possible. On the other hand, a FlexRay Cell-based API needs a lot of resources (one local memory space per FlexRay Cell) and performance (e.g. for the identification of the most recently received value).</p> <p>Moreover, in the static segment of the FlexRay Cycle, all FlexRay Cells always have the same length, independent of the Cell contents. To obtain an optimal use of the FlexRay bandwidth, the FlexRay Interface shall allow transferring PDUs with different transmission periods in the same FlexRay Cell.</p>
<b>Use Case:</b>	<p>The following series-relevant example: An ECU sends the following PDUs using a 5ms Cycle:</p> <ul style="list-style-type: none"> <li>• Acceleration (A) with a 2,5ms period</li> <li>• Status (S) with a 5ms period</li> <li>• Controller Status (C) with a 20ms period</li> <li>• Fault Status (F) with a 80ms period</li> </ul>

	<p style="text-align: center;">FlexRay bus schedule with only one PDU per FlexRay Cell</p>	
	<p style="text-align: center;">FlexRay bus schedule with multiple PDUs per FlexRay Cell</p>	
<p>Conclusion: Only half the bandwidth is required if multiple PDUs are scheduled per FlexRay Cell.</p>		
<b>Dependencies:</b>	<p><a href="#">[BSW05002]</a> FlexRay Modules as Only Necessarily Synchronous SW Modules  <a href="#">[BSW05003]</a> Support of Slot/Cycle Multiplexing</p>	
<b>Conflicts:</b>	--	
<b>Supporting Material:</b>	--	
<b>Contributes to:</b>	--	

**5.1.2.1.2 [BSW05010] Unique PDU-ID**

<b>ID:</b>	BSW05010
<b>Initiator:</b>	BMW
<b>Date:</b>	23.07.2004
<b>Short Description:</b>	Identify each PDU with a unique PDU-ID.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall identify each PDU with a unique PDU-ID, based on the operation to be carried out with this PDU.
<b>Rationale:</b>	Unique PDU-IDs are being used in each BSW module throughout the AUTOSAR Com stack to identify each single PDU and to decide on the handling of the respective PDU. Within the FlexRay Interface, independent sets of PDU-IDs may be used for independent types of operation (i.e.

	sending & receiving).
	Unlike with other communication systems – as for example CAN – the Frame-ID is not sufficient for the identification of an PDU
<b>Use Case:</b>	Interaction of the FlexRay Interface with higher layer BSW modules, e.g. the PDU-Router.
<b>Dependencies:</b>	<a href="#">[BSW05004]</a> PDU-Based API
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.2.1.3 [BSW05126] PDU Update/Valid Information

<b>ID:</b>	BSW05126
<b>Initiator:</b>	BMW
<b>Date:</b>	16.02.2005
<b>Short Description:</b>	Generate, transmit, and (upon reception) handle PDU-based update/valid information where necessary.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall be configurable <a href="#">at system configuration time</a> to generate, transmit, and (upon reception) handle PDU-based update/valid information where necessary. Per PDU, this information shall not consume more than one bit in the FlexRay Frame. This feature shall be configurable <a href="#">at system configuration time</a> independently for each FlexRay Frame.
<b>Rationale:</b>	In certain configurations of the FlexRay Interface and the FlexRay CC, this information is necessary in order for the receiving FlexRay Interface to be able to decide whether a specific PDU contained in a received FlexRay Frame is up-to-date or outdated.
<b>Use Case:</b>	Assume two PDUs to be packed into the same FlexRay Frame. Assume further that only one of the two PDUs has been updated before the scheduled sending of this Frame, and the other PDU has not. The receiver needs to be able to recognize which of the PDUs is valid (i.e. contains updated data) and which is not (i.e. contains invalid or old data). Therefore, this update information has to be generated by the sending FlexRay Interface and transmitted via the FlexRay bus to the receiving FlexRay Interface which evaluates it and acts according to this evaluation.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	Signal update information of COM
<b>Contributes to:</b>	--

### 5.1.2.1.4 [BSW05097] Number of FlexRay Drivers per FlexRay Interface

<b>ID:</b>	BSW05097
<b>Initiator:</b>	DECOMSYS
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Communicate with at least four FlexRay Drivers.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall be able to communicate with at least four FlexRay Drivers. The actual number of FlexRay Drivers serviced by the FlexRay Interface shall be defined <a href="#">at system configuration time</a> and shall be less or equal to the number of FlexRay Drivers this FlexRay Interface supports.
<b>Rationale:</b>	Required in order to fulfill <a href="#">[BSW05007]</a> for four different types of FlexRay Communication Controllers, each of them requiring their own FlexRay

	Driver.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05007]</a> Number of FlexRay CCs per Interface <a href="#">[BSW05065]</a> Number of FlexRay CCs per Driver
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.1.5 [BSW05007] Number of FlexRay CCs per Interface

<b>ID:</b>	BSW05007
<b>Initiator:</b>	BMW
<b>Date:</b>	23.07.2004
<b>Short Description:</b>	Communicate with at least four FlexRay CCs.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall be able to communicate with at least four FlexRay CCs via the appropriate FlexRay Driver(s). The actual number of CCs operated by the FlexRay Interface shall be defined <a href="#">at system configuration time</a> and shall be less or equal to the number of CCs this FlexRay Interface supports.
<b>Rationale:</b>	Future network topologies might demand more than one FlexRay CC on one ECU.
<b>Use Case:</b>	An ECU connected to two FlexRay Clusters.
<b>Dependencies:</b>	<a href="#">[BSW05065]</a> Number of FlexRay CCs per Driver <a href="#">[BSW05097]</a> Number of FlexRay Drivers per FlexRay Interface
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.1.6 [BSW05130] Transmit Request Queuing

<b>ID:</b>	BSW05130
<b>Initiator:</b>	BMW
<b>Date:</b>	20.07.2005
<b>Short Description:</b>	Support of PDU transmission buffer queues located in a higher BSW module.
<b>Type:</b>	General
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall support PDU transmission buffer queues (FIFOs) of configurable size located in a higher BSW module by queuing the transmit requests originating from this BSW module.
<b>Rationale:</b>	A higher BSW module might use a FIFO to queue PDUs to be transmitted, but might forward the transmission request to the FlexRay Interface faster (within a limited time interval) than this PDU can be transmitted via FlexRay.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.1.7 [BSW005215] Transmit Request Cancellation

<b>ID:</b>	BSW005215
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Transmit Request Cancellation

<b>Importance:</b>	high   medium   low
<b>Description:</b>	The FlexRay Interface shall provide an API that cancels transmit request.
<b>Rationale:</b>	Cancellation API is a functionality to remove transmit request from CC's buffer which is realized by calling an API provided by FlexRay Driver.
<b>Use Case:</b>	When transmission of L-PDU is suspended due to the congested condition in dynamic segment and its containing data becomes outdated, such a request should be cancelled by this API.
<b>Dependencies:</b>	<a href="#">[BSW05205]</a> Transmit Request Cancellation (in FlexRay Driver)
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00303

### 5.1.2.1.8 Configuration

#### 5.1.2.1.8.1 [BSW05060] Scheduling of Copy Operation into/from FlexRay CC

<b>ID:</b>	BSW05060
<b>Initiator:</b>	BMW
<b>Date:</b>	15.09.2004
<b>Short Description:</b>	Copy the data into/from the FlexRay CC's transmit/receive buffers at the desired point in the FlexRay global time.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall be configured <a href="#">at system configuration time</a> to copy, by means of the corresponding FlexRay Driver, the data into/from the FlexRay CC's transmit/receive buffers at the desired point in the FlexRay global time. This copy operation shall be carried out in a task configured (i.e. scheduled) <a href="#">at system configuration time</a> . This task has to be synchronous to the communication schedule of the involved FlexRay CC. Together with the copy operation, a reconfiguration of a transmit/receive buffer during normal active mode may be carried out, if supported by the FlexRay CC. The configuration of the copy operation (i.e. the schedule) shall be changeable <a href="#">by a flashing process</a> .
<b>Rationale:</b>	FlexRay Controllers have to be serviced synchronously to the FlexRay communication schedule.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05058]</a> Configuration of FlexRay Driver at System Configuration Time <a href="#">[BSW05034]</a> Configuration Modifiable by a Flashing Process
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.1.8.2 [BSW005060] FlexRay Network Management Scheduling Timing Window Relief

<b>ID:</b>	BSW005060
<b>Initiator:</b>	GM
<b>Date:</b>	31.01.2008
<b>Short Description:</b>	FlexRay Network Management Scheduling Timing Window Relief
<b>Importance:</b>	high   medium   low
<b>Description:</b>	The timing window to schedule the NM task when using the Static segment NM Vector hardware service is too constrained and a timing window relief is desired. Particularly, there is a dependence between when a NM task can be scheduled and when the NM message is transmitted in a slot in the <u>same cycle</u> . A timing window relief could aim at the following two windows:

	<ul style="list-style-type: none"> <li>a) The NM Task could be scheduled anywhere in the Communication Cycle.</li> <li>b) The NM Task could be scheduled anywhere in the Static segment of the Communication Cycle.</li> </ul>
<b>Rationale:</b>	
<b>Use Case:</b>	<ul style="list-style-type: none"> <li>1. Network Management need not run every FlexRay communication cycle saving processing time and resources.</li> <li>2. Less stringent constraints on scheduling Network Management main function and transmission slots.</li> </ul>
<b>Dependencies:</b>	Error Handling, DET, DCM, FlexRay Driver, FlexRay Interface.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00274

#### 5.1.2.1.8.3 [BSW005061] NM Vector Support

<b>ID:</b>	BSW005061
<b>Initiator:</b>	DAIMLER
<b>Date:</b>	31.01.2007
<b>Short Description:</b>	Possibility to read the NM Vector
<b>Importance:</b>	high   medium   low
<b>Description:</b>	The NM Vector feature provided by the FlexRay Communication Controller shall be supported
<b>Rationale:</b>	The NM Vector can be used by the FlexRay Network Management
<b>Use Case:</b>	Shutting down a FlexRay cluster synchronously
<b>Dependencies:</b>	FlexRay Network Management
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.1.8.4 [BSW05096] Assignment of Drivers to Controllers

<b>ID:</b>	BSW05096
<b>Initiator:</b>	DECOMSYS
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Assignment of FlexRay Drivers to FlexRay Controllers.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The configuration of the FlexRay Interface shall specify which FlexRay Communication Controller is to be served by which FlexRay Driver. Thus a mapping between used FlexRay Drivers and available FlexRay Controllers is made though the interface configuration <a href="#">at system configuration time</a> .
<b>Rationale:</b>	This information has to be available to the FlexRay Interface in order to invoke the correct Driver for a given FlexRay Controller on an ECU.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05056]</a> Configuration of the FlexRay Interface at System Configuration Time
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--



## 5.1.2.2 Initialization

### 5.1.2.2.1 [BSW05013] Initialize Local Memory Space

<b>ID:</b>	BSW05013
<b>Initiator:</b>	BMW
<b>Date:</b>	26.07.2004
<b>Short Description:</b>	Initialize the FlexRay Interface's local memory space.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to initialize the local memory space used to store the PDU data and their corresponding properties.
<b>Rationale:</b>	The initialization of the local memory space is necessary to store the transferred data from the FlexRay Controller. Usually, variables (e.g. pointers) have to be initialized before they are being used during runtime.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.2.2.2 [BSW05031] Initialization of a FlexRay CC

<b>ID:</b>	BSW05031
<b>Initiator:</b>	BMW
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Initialization and configuration of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to initialize and configure a specific FlexRay CC. Thereby, the transmit/receive buffers and the low level parameters of the FlexRay CC shall be configured.
<b>Rationale:</b>	--
<b>Use Case:</b>	Initial configuration.
<b>Dependencies:</b>	<a href="#">[BSW05116]</a> Initialization of FlexRay CC <a href="#">[BSW05011]</a> Initialize Low-Level Parameters <a href="#">[BSW05012]</a> Initialize FlexRay CC Transmit/Receive Buffers
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.2.2.3 [BSW05034] Configuration Modifiable by a Flashing Process

<b>ID:</b>	BSW05034
<b>Initiator:</b>	BMW
<b>Date:</b>	16.08.2004
<b>Short Description:</b>	Modification of configuration data <a href="#">by a flashing process</a> .
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	All configuration data of the FlexRay Interface defined <a href="#">at system configuration time</a> shall be modifiable <a href="#">by a flashing process</a> .  These modifiable configuration data encompass (amongst other configuration items) the scheduling of the copy operation into/from the FlexRay CCs' transmit/receive buffers as well as the configuration of those

	<p>buffers and the FlexRay Interface's local memory space.</p> <p>The modification shall be carried out in a specific mode (e.g. in the garage) and not during run-time (i.e. while the car is being driven).</p>
<b>Rationale:</b>	Re-flashing of a changed communication schedule.
<b>Use Case:</b>	If e.g. the only modification in a configuration is the Frame-ID of a specific PDU (i.e. the FlexRay Frame in which this PDU is being sent), this shall not automatically implicate a new compilation of the complete ECU code! Only the altered parameters need to be modified.
<b>Dependencies:</b>	<a href="#">[BSW05060]</a> Scheduling of Copy Operation into/from FlexRay CC <a href="#">[BSW05012]</a> Initialize FlexRay CC Transmit/Receive Buffers <a href="#">[BSW05013]</a> Initialize Local Memory Space
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.2.4 [BSW05042] Switch Configuration in Normal Active Mode

<b>ID:</b>	BSW05042
<b>Initiator:</b>	BMW
<b>Date:</b>	30.08.2004
<b>Short Description:</b>	Switching the configuration of a specific FlexRay Communication Controller's transmit/receive buffers and the local memory space <a href="#">during runtime</a> (i.e. in normal active mode), if supported by this FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	<p>The FlexRay Interface shall allow switching from one configuration of a specific FlexRay Communication Controller's transmit/receive buffers and the local memory space to another one, both stored in the ECU's memory space. This switching shall be possible <a href="#">during runtime</a>, i.e. while the CC is in normal active mode, if supported by this FlexRay CC.</p> <p>It shall be ensured that none of the selectable configurations violate the rules for Slot Multiplexing set forth in <a href="#">[FR_PROT_SPEC]</a>.</p> <p>The switching of the configuration includes a reconfiguration of the FlexRay Communication Controller's transmit/receive buffers. Therefore the following restrictions apply:</p> <ul style="list-style-type: none"> <li>In the dynamic segment any FlexRay Communication Controller's transmit/receive buffer shall be used maximally once per Cycle. This limits the reconfiguration possibilities and thus restricts the number of transmittable (sent and received) Frames per dynamic segment to the accumulated number (over all CCs on one ECU) of transmit/receive buffers connected to one Cluster.</li> </ul> <p>The FlexRay Interface shall support at least 4 different configurations. The actual number of configurations stored in the ECU's memory space shall be defined <a href="#">at system configuration time</a> and shall be less or equal to the number of configurations this FlexRay Interface supports.</p> <p>For safety reasons it shall be possible to completely deactivate this feature <a href="#">pre compile time</a>.</p> <p>For safety reason such a switching operation shall not be carried out while the car is being driven, but only in the garage in a specific mode.</p>
<b>Rationale:</b>	--
<b>Use Case:</b>	<p>Specific flash mode for the gateway:</p> <p>In order to increase the throughput of the gateway, and thus speed up the flashing process, transmit/receive buffers normally used for application data transmission could temporarily be used for the flashing process.</p>

<b>Dependencies:</b>	<a href="#">[BSW05012]</a> Initialize FlexRay CC Transmit/Receive Buffers <a href="#">[BSW05013]</a> Initialize Local Memory Space
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.2.3 Start-up Operation

#### 5.1.2.3.1 [BSW05015] Start-up of a FlexRay CC

<b>ID:</b>	BSW05015
<b>Initiator:</b>	BMW
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Start-up of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to start-up a specific FlexRay CC.
<b>Rationale:</b>	Start-up of a specific FlexRay Controller after proper configuration.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05109]</a> Start-up of a FlexRay CC <a href="#">[BSW05031]</a> Initialization of a FlexRay CC
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.3.2 [BSW05018] Sending of a Wake-Up Pattern

<b>ID:</b>	BSW05018
<b>Initiator:</b>	BMW
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Sending of a wake-up pattern on a specific FlexRay Channel of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to send a wake-up pattern on a specific FlexRay Channel of a specific FlexRay CC. The FlexRay Specification allows only a wake-up on one Channel at a time. However, the FlexRay Interface shall support the sending of a wake-up pattern on any one of the two FlexRay Channels. The FlexRay Interface shall observe the restrictions set forth in the FlexRay Protocol Specification concerning reception of a wake-up pattern or a Frame header during own wake-up pattern sending attempts.
<b>Rationale:</b>	Enable waking up a FlexRay Cluster.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05117]</a> Sending of Wake-Up Pattern
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_PROT_SPEC]</a>
<b>Contributes to:</b>	BRF00273

### 5.1.2.3.3 [BSW05158] Get FlexRay Transceiver Wake-up Reason

<b>ID:</b>	BSW05158
<b>Initiator:</b>	VW
<b>Date:</b>	24.01.2006
<b>Short Description:</b>	Get the wake-up reason of a specific FlexRay Transceiver device.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to get the wake-up reason of a specific FlexRay Transceiver device.
<b>Rationale:</b>	Provide EcuM with a possibility to find out the wake-up reason.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05144]</a> Get FlexRay Transceiver Wake-up Reason
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_EPL_SPEC]</a>
<b>Contributes to:</b>	--

### 5.1.2.3.4 [BSW05159] Enable FlexRay Transceiver Wake-up Indication

<b>ID:</b>	BSW05159
<b>Initiator:</b>	VW
<b>Date:</b>	24.01.2006
<b>Short Description:</b>	Enable the wake-up indication of a specific FlexRay Transceiver device.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to enable the wake-up indication of a specific FlexRay Transceiver device.
<b>Rationale:</b>	EcuM might be interested in a wake-up event via FlexRay.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05160]</a> Disable FlexRay Transceiver Wake-up Indication
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_EPL_SPEC]</a>
<b>Contributes to:</b>	--

### 5.1.2.3.5 [BSW05160] Disable FlexRay Transceiver Wake-up Indication

<b>ID:</b>	BSW05160
<b>Initiator:</b>	VW
<b>Date:</b>	24.01.2006
<b>Short Description:</b>	Disable the wake-up indication of a specific FlexRay Transceiver device.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to disable the wake-up indication of a specific FlexRay Transceiver device.
<b>Rationale:</b>	EcuM might not be interested in a wake-up event via FlexRay.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05159]</a> Enable FlexRay Transceiver Wake-up Indication
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_EPL_SPEC]</a>
<b>Contributes to:</b>	--

### 5.1.2.3.6 [BSW05161] Clear FlexRay Transceiver Wake-up Events

<b>ID:</b>	BSW05161
<b>Initiator:</b>	VW
<b>Date:</b>	24.01.2006
<b>Short Description:</b>	Clear all pending wake-up events of a specific FlexRay Transceiver device.

<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to clear all pending wake-up events of a specific FlexRay Transceiver device.
<b>Rationale:</b>	EcuM needs to reset the wake-up events of FlexRay Transceiver devices.
<b>Use Case:</b>	Wake-Up via FlexRay Communications Bus
<b>Dependencies:</b>	<a href="#">[BSW05165]</a> Cluster-wide Clear FlexRay Transceiver Wake-up Events
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_EPL_SPEC]</a>
<b>Contributes to:</b>	--

### 5.1.2.4 FlexRay Specific Information

This section includes all the FlexRay-specific requirements.

#### 5.1.2.4.1 [BSW05022] Get FlexRay CC POC Status

<b>ID:</b>	BSW05022
<b>Initiator:</b>	BMW
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Get the CC POC status of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to get the CC POC status of a specific FlexRay CC. All information contained in the vPOC structure as defined in chapter "2.2.1.3 POC status" of the <a href="#">[FR_PROT_SPEC]</a> shall be returned.
<b>Rationale:</b>	Software modules like Mode Manager, Network Management, or DCM might be interested in the FlexRay CC POC status.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05120]</a> Get FlexRay CC POC Status
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.4.2 [BSW05039] Set FlexRay Transceiver Operation Mode

<b>ID:</b>	BSW05039
<b>Initiator:</b>	BMW
<b>Date:</b>	20.08.2004
<b>Short Description:</b>	Set the operation mode of a specific FlexRay Transceiver device.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to set the operation mode of a specific FlexRay Transceiver device. According to <a href="#">[FR_EPL_SPEC]</a> , at least these operation modes shall be supported: <ul style="list-style-type: none"> <li>• BD_Normal</li> <li>• BD_Standby</li> <li>• BD_Receive_only</li> <li>• BD_Sleep</li> </ul>
<b>Rationale:</b>	Provide ComM with a possibility to set the operation mode of a specific FlexRay Transceiver device.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05166]</a> Set FlexRay Transceiver Operation Mode <a href="#">[BSW05157]</a> Get FlexRay Transceiver Operation Mode
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_EPL_SPEC]</a>
<b>Contributes to:</b>	--

#### 5.1.2.4.3 [BSW05157] Get FlexRay Transceiver Operation Mode

<b>ID:</b>	BSW05157
<b>Initiator:</b>	VW
<b>Date:</b>	24.01.2006
<b>Short Description:</b>	Get the operation mode of a specific FlexRay Transceiver device.
<b>Type:</b>	New

<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to get the operation mode of a specific FlexRay Transceiver device. According to <a href="#">[FR_EPL_SPEC]</a> , at least these operation modes shall be supported: <ul style="list-style-type: none"> <li>• BD_Normal</li> <li>• BD_Standby</li> <li>• BD_Receive_only</li> <li>• BD_Sleep</li> </ul>
<b>Rationale:</b>	Provide FlexRay StateManager with a possibility to get the operation mode of a specific FlexRay Transceiver device.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05167]</a> Get FlexRay Transceiver Operation Mode <a href="#">[BSW05039]</a> Set FlexRay Transceiver Operation Mode
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_EPL_SPEC]</a>
<b>Contributes to:</b>	--

#### 5.1.2.4.4 [BSW05174] Interrupt Handling

<b>ID:</b>	BSW05174
<b>Initiator:</b>	BMW
<b>Date:</b>	01.02.2006
<b>Short Description:</b>	The FlexRay Interface shall provide services to handle interrupts of a FlexRay Communication Controller.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide services to handle interrupts of a FlexRay Communication Controller. At least the following functionality shall be provided: <ul style="list-style-type: none"> <li>• get interrupt source</li> <li>• service interrupt</li> <li>• enable interrupt</li> <li>• disable interrupt</li> <li>• acknowledge interrupt.</li> </ul>
<b>Rationale:</b>	The FlexRay Interface might use an alarm (absolute timer interrupt) of a FlexRay Communication Controller to run synchronously to the FlexRay global time. This interrupt has to be serviced.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05125]</a> Interrupt Handling
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.5 Normal Operation

##### 5.1.2.5.1 [BSW05170] Receive PDU

<b>ID:</b>	BSW05170
<b>Initiator:</b>	BMW
<b>Date:</b>	29.01.2006
<b>Short Description:</b>	Reception of PDUs via the FlexRay communication system.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall retrieve PDUs received via the FlexRay communication system and indicate the reception to the appropriate higher-

	<p>layer BSW module, allowing this module to retrieve the PDU data.</p> <p>Depending on static/dynamic configuration of the FlexRay Interface, it might be possible that one FlexRay Frame contains more than one PDU. In this case, the FlexRay Interface shall disassemble the FlexRay Frame and indicate the reception of each PDU to the appropriate higher BSW module.</p> <p>If the use of PDU-based update/valid information has been configured for a FlexRay Frame, the reception indication shall consider this information.</p>
<b>Rationale:</b>	As explained in the general requirement, the FlexRay Interface API shall be PDU-based. The access via the PDU is the only possibility to access the FlexRay data.
<b>Use Case:</b>	An application needs to receive information from the FlexRay network.
<b>Dependencies:</b>	<a href="#">[BSW05004]</a> PDU-Based API <a href="#">[BSW05010]</a> Unique PDU-ID <a href="#">[BSW05126]</a> PDU Update/Valid Information
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.5.2 [BSW05027] Transmit PDU

<b>ID:</b>	BSW05027
<b>Initiator:</b>	BMW
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Transmission of PDUs via the FlexRay communication system.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	<p>The FlexRay Interface shall provide a software interface to transmit a PDU via the FlexRay communication system</p> <p>Depending on static configuration of the FlexRay Interface, it might be possible that one FlexRay Frame contains more than one PDU. In this case, the FlexRay Interface shall assemble the FlexRay Frame of the corresponding PDUs before the Frame is transmitted.</p> <p>If the use of PDU-based update/valid information has been configured for a FlexRay Frame, the FlexRay Interface shall correctly generate this information.</p>
<b>Rationale:</b>	As explained in the general requirement the FlexRay Interface API shall be PDU-based. The access via the PDU is the only possibility to access the FlexRay data.
<b>Use Case:</b>	An applications needs to send information over the FlexRay network.
<b>Dependencies:</b>	<a href="#">[BSW05004]</a> PDU-Based API <a href="#">[BSW05010]</a> Unique PDU-ID <a href="#">[BSW05126]</a> PDU Update/Valid Information
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.2.5.3 [BSW05171] Provide PDU Transmit Confirmation

<b>ID:</b>	BSW05171
<b>Initiator:</b>	BMW
<b>Date:</b>	30.07.2004
<b>Short Description:</b>	Provide the transmit status of a PDU.
<b>Type:</b>	New
<b>Importance:</b>	High



<b>Description:</b>	The FlexRay Interface shall be configurable to provide the appropriate higher-layer BSW module with a transmit confirmation for a PDU this BSW module has requested to be sent. <a href="#">At system configuration time</a> it shall be configurable per PDU whether a transmit confirmation shall be provided.
<b>Rationale:</b>	In the dynamic segment, the transmission of a Frame is not guaranteed. Therefore, the application needs information whether a PDU has been sent.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05004]</a> PDU-Based API
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

## 5.1.2.6 Shutdown Operation

### 5.1.2.6.1 [BSW05016] Abortion of a FlexRay CC Communication

<b>ID:</b>	BSW05016
<b>Initiator:</b>	BMW
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Immediate abortion of the communication of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to immediately abort the communication of a specific FlexRay CC by setting this FlexRay CC into the "HALT" state. Thus, the FlexRay CC will stop participating in the communication and lose synchronicity with the FlexRay global time.
<b>Rationale:</b>	--
<b>Use Case:</b>	Error confinement
<b>Dependencies:</b>	<a href="#">[BSW05114]</a> Abortion of FlexRay CC Communication <a href="#">[BSW05113]</a> Abortion of a FlexRay Cluster Communication
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.2.6.2 [BSW05063] Halt of a FlexRay CC Communication

<b>ID:</b>	BSW05063
<b>Initiator:</b>	BMW
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Halting of the communication of a specific FlexRay CC at the end of the current Communication Cycle.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide a software interface to halt the communication of a specific FlexRay CC at the end of the current Communication Cycle by setting this FlexRay CC into the "HALT" state at the end of the current Communication Cycle. Thus, the FlexRay CC will stop participating in the communication and lose synchronicity with the FlexRay global time.
<b>Rationale:</b>	--
<b>Use Case:</b>	Error confinement
<b>Dependencies:</b>	<a href="#">[BSW05115]</a> Halt of FlexRay CC Communication <a href="#">[BSW05016]</a> Abortion of a FlexRay CC Communication
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_PROT_SPEC]</a>
<b>Contributes to:</b>	--

## 5.1.2.7 Fault Operation

### 5.1.2.7.1 [BSW05175] Provide Error Information

<b>ID:</b>	BSW05175
<b>Initiator:</b>	BMW
<b>Date:</b>	31.01.2006
<b>Short Description:</b>	Provide error-related information using the aggregated channel status
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall provide the DEM with error-related information using the aggregated channel status that is only known to the FlexRay Interface.
<b>Rationale:</b>	Some error-related information is only known to the FlexRay Interface.
<b>Use Case:</b>	The following error-related information shall be derived using the using the aggregated channel status: <ul style="list-style-type: none"> <li>- SyntaxError: Flag that reports a syntax error within a slot.</li> <li>- ContentError: Flag that reports a content error within a slot.</li> <li>- TxConflict: Flag that reports a transmission conflict within a slot.</li> <li>- Bviolation: Flag that reports a boundary violation within a slot.</li> </ul>
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00264 BRF00266

### 5.1.2.7.2 [BSW005200] Error Information in Transmitted Frame

<b>ID:</b>	BSW005200
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Error Information in Transmitted Frame
<b>Importance:</b>	<b>High</b>   medium   low
<b>Description:</b>	The FlexRay Interface shall call API of <a href="#">[BSW05207]</a> periodically that detects error information in transmitted frames. Pre-compile configuration parameter shall determine whether this functionality is activated.
<b>Rationale:</b>	The FlexRay modules should provide information that only the modules can detect.
<b>Use Case:</b>	Applications can use this information as an indicator of network quality.
<b>Dependencies:</b>	<a href="#">[BSW05207]</a> API to Detect Error Information in Transmitted Frame
<b>Conflicts:</b>	<a href="#">[BSW05175]</a> Provide Error Information
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00302

### 5.1.2.7.3 [BSW005201] Error Information in Received Frame

<b>ID:</b>	BSW005201
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Error Information in Received Frame
<b>Importance:</b>	<b>High</b>   medium   low
<b>Description:</b>	The FlexRay Interface shall call API of <a href="#">[BSW05208]</a> periodically that detects error information in received frames. Pre-compile configuration parameter shall determine whether this

	functionality is activated.
<b>Rationale:</b>	The FlexRay modules should provide information that only the modules can detect.
<b>Use Case:</b>	Applications can be implemented with an indirect-NM functionality when the error information mentioned above is provided.
<b>Dependencies:</b>	<a href="#">[BSW05208]</a> API to Detect Error Information in Received Frame <a href="#">[BSW05210]</a> Criteria on Validness of Received Frame
<b>Conflicts:</b>	<a href="#">[BSW05175]</a> Provide Error Information
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00304

#### 5.1.2.7.4 [BSW005202] Error Information in NIT

<b>ID:</b>	BSW005202
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Error Information in NIT
<b>Importance:</b>	<b>High</b>   medium   low
<b>Description:</b>	The FlexRay Interface shall call API of <a href="#">[BSW05209]</a> periodically that detects error information in NIT. Pre-compile configuration parameter shall determine whether this functionality is activated.
<b>Rationale:</b>	The FlexRay modules should provide information that only the modules can detect.
<b>Use Case:</b>	Application can use this information as an indicator of network quality.
<b>Dependencies:</b>	<a href="#">[BSW05209]</a> API to Detect Error Information in NIT
<b>Conflicts:</b>	<a href="#">[BSW05175]</a> Provide Error Information
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00304

#### 5.1.2.7.5 [BSW005203] Error Information in Bus Driver

<b>ID:</b>	BSW005203
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Error Information in Bus Driver
<b>Importance:</b>	<b>High</b>   medium   low
<b>Description:</b>	The FlexRay Interface shall call API of <a href="#">[BSW05212]</a> periodically that detects error information in BD. Pre-compile configuration parameter shall determine whether this functionality is activated.
<b>Rationale:</b>	The FlexRay modules should provide information that only the modules can detect.
<b>Use Case:</b>	Applications could take actions to recover the failure cause like resetting the modules when they receive this error information.
<b>Dependencies:</b>	<a href="#">[BSW05212]</a> Detect Error Information in Bus Driver
<b>Conflicts:</b>	<a href="#">[BSW05175]</a> Provide Error Information
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00264 BRF00304

#### 5.1.2.7.6 [BSW005204] Customizable Checking Function of L-PDU Length

<b>ID:</b>	BSW005204
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Customizable Checking Function of L-PDU Length
<b>Importance:</b>	<b>high</b>   medium   low

<b>Description:</b>	In the FlexRay Interface, pre-compile configuration parameter shall define a mode in checking function of L-PDU length. In a loose mode, L-PDU with expected length or longer length than expected should be accepted. In a strict mode, only L-PDU with expected length should be accepted.
<b>Rationale:</b>	When using a dynamic segment, receivers would likely get L-PDUs with unexpected length. There are two requirements like follows on the behavior for such a case. <ul style="list-style-type: none"> <li>• Longer L-PDU should be rejected because such an L-PDU would likely be an illegal one that has been mistakenly transmitted from other nodes.</li> <li>• Longer L-PDU should be accepted. This mode is necessary when a user assumes that the length of L-PDU will be expanded in the future version of ECU's software.</li> </ul> To satisfy these use-cases, there should be two modes in the functionality's behavior.
<b>Use Case:</b>	(See the description in Rationale)
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00305

### 5.1.3 FlexRay Driver

#### 5.1.3.1 General Requirements

##### 5.1.3.1.1 [BSW05064] Abstraction of FlexRay CC-specific Implementation

<b>ID:</b>	BSW05064
<b>Initiator:</b>	BMW
<b>Date:</b>	16.12.2004
<b>Short Description:</b>	Abstraction from the FlexRay CC-specific implementation.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall provide an abstraction from the FlexRay CC-specific implementation as well as a generic FlexRay Driver API to the upper software layer (FlexRay Interface).
<b>Rationale:</b>	All FlexRay Drivers should provide the same API semantics/signature. Upper layers should not depend on the concrete implementation of specific features (e.g. optional features) of the CC. They need to be hardware-independent.
<b>Use Case:</b>	Several (different) FlexRay CCs (e.g. from different manufacturers) handled by one FlexRay Interface.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

##### 5.1.3.1.2 [BSW05065] Number of FlexRay CCs per Driver

<b>ID:</b>	BSW05065
<b>Initiator:</b>	BMW
<b>Date:</b>	16.12.2004
<b>Short Description:</b>	Communicate with at least four FlexRay CCs of the same type.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall be able to communicate with at least four FlexRay

	CCs of the same type. The actual number of CCs operated by one FlexRay Driver shall be defined <a href="#">at system configuration time</a> and shall be less or equal to the number of CCs this FlexRay Driver supports.
<b>Rationale:</b>	Future network topologies might demand more than one FlexRay CC on one ECU.
<b>Use Case:</b>	An ECU connected to two FlexRay Clusters.
<b>Dependencies:</b>	<a href="#">[BSW05007]</a> Number of FlexRay CCs per Interface <a href="#">[BSW05097]</a> Number of FlexRay Drivers per FlexRay Interface
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.3.1.3 [BSW05005] Support of Hardware FIFO Mechanism

<b>ID:</b>	BSW05005
<b>Initiator:</b>	BMW
<b>Date:</b>	23.07.2004 / Update: 31.07.2008
<b>Short Description:</b>	Support the optional CC hardware FIFO mechanism, if supported by the FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall support the CC hardware FIFO mechanism, if supported by the FlexRay CC, and abstract its usage because this mechanism is optional.
<b>Rationale:</b>	In the future, a FlexRay CC could have a low number of transmit/receive buffers to reduce its cost. Then the high number of different Slot-IDs – with slow communication requirements – in the dynamic segment requires a FIFO on the reception side.
<b>Use Case:</b>	The hardware FIFO could be configured for the dynamic segment and / or the static segment of the FlexRay Communication Cycle.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	FlexRay Protocol Specification 2.1 FlexRay Protocol Specification 3.0
<b>Contributes to:</b>	--

#### 5.1.3.1.4 [BSW005006] Support of Message ID Filter Mechanism

<b>ID:</b>	BSW005006
<b>Initiator:</b>	BMW
<b>Date:</b>	31.07.2008
<b>Short Description:</b>	Hardware Message ID Filter Mechanism according to the FlexRay Protocol Specification 3.0 shall be supported
<b>Type:</b>	New
<b>Importance:</b>	Medium
<b>Description:</b>	Hardware filtering mechanism provided by the FlexRay Communication Controller that can be used for selecting receive buffers based on a message ID within the dynamic segment shall be supported
<b>Rationale:</b>	- Avoidance of unnecessary receive indications to upper layers - Dynamic bandwidth allocation
<b>Use Case:</b>	Optimized usage of FlexRay Transport Protocol
<b>Dependencies:</b>	- Hardware message ID filtering can only be used in combination RX-FIFO - Hardware message ID filtering is an optional feature for FlexRay 2.1 CCs
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	FlexRay Protocol Specification 3.0
<b>Contributes to:</b>	BRF00192

#### 5.1.3.1.5 [BSW005007] Support of MTM Mode

<b>ID:</b>	BSW005007
<b>Initiator:</b>	BMW
<b>Date:</b>	31.07.2008
<b>Short Description:</b>	Support of Time Triggered Master Mode
<b>Type:</b>	New
<b>Importance:</b>	Medium
<b>Description:</b>	It shall be possible to use the Time Triggered Master mode provided by FlexRay 3.0 Communication Controller. I. e. support of: - TT-L Time Triggered Local Master Sync - TT-E Time Triggered External Sync It is also possible to set up a TT-L/TT-E cluster with more than one master for increased robustness
<b>Rationale:</b>	Setup up a network with two or more time synchronized clusters (e.g. a Backbone is the timing source for multiple clusters, cascaded clusters)
<b>Use Case:</b>	Several Topologies
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	FlexRay Protocol Specification 3.0
<b>Contributes to:</b>	BRF00277

#### 5.1.3.1.6 [BSW005008] FlexRay 2.1 and 3.0 Support

<b>ID:</b>	BSW005008
<b>Initiator:</b>	BMW
<b>Date:</b>	31.07.2008
<b>Short Description:</b>	FlexRay 2.1 and 3.0 Hardware shall be supported
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	FlexRay 2.1 and 3.0 Communication Controllers and Transceivers shall be supported
<b>Rationale:</b>	New FlexRay Hardware feature Support is required by several OEMs
<b>Use Case:</b>	BSW05005, BSW005006, BSW005007
<b>Dependencies:</b>	Refer to Supporting Material
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	FlexRay Protocol Specification 3.0 FlexRay Electrical Physical Layer 3.0 FlexRay Protocol Specification 2.1 Rev. A FlexRay Electrical Physical Layer 2.1 Rev. A
<b>Contributes to:</b>	--

#### 5.1.3.1.7 [BSW005009] Support of dynamic LPdu payload length

<b>ID:</b>	BSW005009
<b>Initiator:</b>	BMW
<b>Date:</b>	31.07.2008
<b>Short Description:</b>	Dynamic LPdu payload length shall be supported
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	It shall be possible to transmit and receive dynamic LPdu payload length in the dynamic segment of a FlexRay Communication Cycle.
<b>Rationale:</b>	This feature is required by the FlexRay Transport Protocol and can also be used by the AUTOSAR Communication Module.
<b>Use Case:</b>	Usage of ISO FlexRay Transport Protocol and Support of dynamic Signal Length sent by a SW-C
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--

<b>Supporting Material:</b>	AUTOSAR FlexRay Transport Protocol
<b>Contributes to:</b>	--

#### 5.1.3.1.8 [BSW05024] Abstraction from CC Buffer Configuration

<b>ID:</b>	BSW05024
<b>Initiator:</b>	BMW
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Abstraction from the FlexRay CC buffer configuration.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The software interface of the FlexRay Driver shall be independent of the FlexRay CC transmit/receive buffers' configuration (transmit, receive, filtering ...) and their access.
<b>Rationale:</b>	The FlexRay transmit/receive buffer configuration and properties shall not have any influence on the upper-layer software.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.3.1.9 [BSW05066] L-SDU-Based API

<b>ID:</b>	BSW05066
<b>Initiator:</b>	BMW
<b>Date:</b>	16.12.2004
<b>Short Description:</b>	Provide an L-SDU-based data handling API to all upper layers.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:-</b>	The FlexRay Driver shall provide an L-SDU-based data handling to all upper layers. Based on the unique L-SDU identifier (this is NOT the Frame-ID according to <a href="#">[FR_PROT_SPEC]</a> ) and according to rules defined <a href="#">at system configuration time</a> , the FlexRay Interface assembles L-SDU for sending or extracts them for reception, resp. Therefore, the FlexRay Driver shall provide an L-SDU-based data handling to all upper layers.
<b>Rationale:</b>	Payload contained in L-SDU is usually transferred into/out of a FlexRay Cell via transmit/receive buffers allocated in the FlexRay Controller.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.3.1.10 [BSW005205] Transmit Request Cancellation

<b>ID:</b>	BSW005205
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Transmit Request Cancellation
<b>Importance:</b>	high   medium   low
<b>Description:</b>	The FlexRay Driver shall provide an API that cancels transmit request.
<b>Rationale:</b>	Cancellation API is a functionality to remove transmit request from CC's buffer.
<b>Use Case:</b>	When transmission of L-PDU is suspended due to the congested condition in dynamic segment and its containing data becomes outdated, such a request should be cancelled by this API.

<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00303

#### 5.1.3.1.11 [BSW005206] Hardware Resetting Function on CC

<b>ID:</b>	BSW005206
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Hardware Resetting Function on CC
<b>Importance:</b>	high   medium   low
<b>Description:</b>	FlexRay Driver shall provide a method that reinitializes CC's functionality
<b>Rationale:</b>	When trouble occurs in the hardware level, it's likely to fix the cause by resetting the hardware.
<b>Use Case:</b>	This function shall be executed when so many errors are detected in FlexRay modules.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00307

### 5.1.3.2 Configuration

#### 5.1.3.2.1 [BSW05058] Configuration of FlexRay Driver at System Configuration Time

<b>ID:</b>	BSW05058
<b>Initiator:</b>	BMW
<b>Date:</b>	15.09.2004
<b>Short Description:</b>	The configuration of the FlexRay Driver shall be defined <a href="#">at system configuration time</a> .
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The configuration of the FlexRay Driver shall be defined <a href="#">at system configuration time</a> , i.e. before the execution of the FlexRay Driver code.
<b>Rationale:</b>	The purpose of the FlexRay Driver is to carry out data processing that has been configured <a href="#">at system configuration time</a> , i.e. before the execution of the FlexRay Driver code. It shall not decide or calculate during runtime any communication parameters like the assignment of Slots to ECU or the packaging of PDUs into FlexRay Frames.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05034]</a> Configuration Modifiable by a Flashing Process
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.3.2.2 [BSW05059] Transmit/Receive Buffer Configuration

<b>ID:</b>	BSW05059
<b>Initiator:</b>	BMW
<b>Date:</b>	15.09.2004
<b>Short Description:</b>	Transmit/Receive Buffer Configuration.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall configure the FlexRay CC's transmit/receive buffers according to the configuration of the FlexRay Driver defined at



	system configuration time.
<b>Rationale:</b>	The FlexRay CC's transmit/receive buffers need to be configured before the FlexRay CC can be used for communication.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05058]</a> Configuration of FlexRay Driver at System Configuration Time
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.3.3 Initialization

#### 5.1.3.3.1 [BSW05116] Initialization of FlexRay CC

<b>ID:</b>	BSW05116
<b>Initiator:</b>	BMW
<b>Date:</b>	02.02.2005
<b>Short Description:</b>	Initialization and configuration of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall provide a software interface to initialize and configure a specific FlexRay CC. Thereby, the transmit/receive buffers and the low level parameters of the FlexRay CC shall be configured.
<b>Rationale:</b>	--
<b>Use Case:</b>	Initial configuration.
<b>Dependencies:</b>	<a href="#">[BSW05031]</a> Initialization of a FlexRay CC <a href="#">[BSW05011]</a> Initialize Low-Level Parameters <a href="#">[BSW05012]</a> Initialize FlexRay CC Transmit/Receive Buffers
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.3.3.2 [BSW05011] Initialize Low-Level Parameters

<b>ID:</b>	BSW05011
<b>Initiator:</b>	BMW
<b>Date:</b>	26.07.2004
<b>Short Description:</b>	Initialization of the FlexRay low-level parameters.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall provide a software interface to initialize the FlexRay low-level parameters. The low-level parameters cover all the configurable parameters defined in the <a href="#">[FR_PROT_SPEC]</a> and also the FlexRay CC-specific parameters or CC-specific services (e.g. concerning interrupts).
<b>Rationale:</b>	The initialization of the low-level parameters is necessary to communicate on the bus with a FlexRay Controller. The set of CC parameters is FlexRay CC-specific.
<b>Use Case:</b>	Initialization of the FlexRay communication system.
<b>Dependencies:</b>	<a href="#">[BSW05116]</a> Initialization of FlexRay CC
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.3.3.3 [BSW05012] Initialize FlexRay CC Transmit/Receive Buffers

<b>ID:</b>	BSW05012
<b>Initiator:</b>	BMW
<b>Date:</b>	26.07.2004
<b>Short Description:</b>	Initialize the transmit/receive buffers of a specific FlexRay CC with a default configuration.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall provide a software interface to initialize the transmit/receive buffers of a specific FlexRay CC with a default configuration.
<b>Rationale:</b>	The initialization of the transmit/receive buffers is necessary to transfer data with a FlexRay Controller. The configuration of the transmit/receive buffers and filtering mechanisms is FlexRay CC-specific.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05116]</a> Initialization of FlexRay CC
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.3.4 Start-up Operation

#### 5.1.3.4.1 [BSW05109] Start-up of a FlexRay CC

<b>ID:</b>	BSW05109
<b>Initiator:</b>	BMW
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Start-up of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall provide a software interface to start-up a specific FlexRay CC.
<b>Rationale:</b>	Start-up of a specific FlexRay Controller after proper configuration.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05015]</a> Start-up of a FlexRay CC <a href="#">[BSW05114]</a> Abortion of FlexRay CC Communication <a href="#">[BSW05115]</a> Halt of FlexRay CC Communication
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.3.4.2 [BSW05117] Sending of Wake-Up Pattern

<b>ID:</b>	BSW05117
<b>Initiator:</b>	DECOMSYS
<b>Date:</b>	02.02.2005
<b>Short Description:</b>	Sending of a wake-up pattern on a specific FlexRay Channel of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High

<b>Description:</b>	The FlexRay Driver shall provide a software interface to send a wake-up pattern on a specific FlexRay Channel of a specific FlexRay CC. The FlexRay Specification allows only a wake-up on one Channel at a time. However, the FlexRay Driver shall support the sending of a wake-up pattern on any one of the two FlexRay Channels. The FlexRay Driver shall observe the restrictions set forth in the FlexRay Protocol Specification concerning reception of a wake-up pattern or a Frame header during own wake-up pattern sending attempts.
<b>Rationale:</b>	Enable waking up a FlexRay Cluster.
<b>Use Case:</b>	--
<b>Dependencies:</b>	[BSW05018] Sending of a Wake-Up Pattern
<b>Conflicts:</b>	
<b>Supporting Material:</b>	[FR_PROT_SPEC]
<b>Contributes to:</b>	BRF00273

### 5.1.3.5 FlexRay Specific Information

This section includes all the FlexRay-specific status information needed by the upper-layer Software e.g. synchronization information, etc...

#### 5.1.3.5.1 [BSW05120] Get FlexRay CC POC Status

<b>ID:</b>	BSW05120
<b>Initiator:</b>	DECOMSYS
<b>Date:</b>	02.02.2005
<b>Short Description:</b>	Get the CC POC status of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall provide a software interface to get the CC POC status of a specific FlexRay CC. All information contained in the vPOC structure as defined in chapter "2.2.1.3 POC status" of the [FR_PROT_SPEC] shall be returned.
<b>Rationale:</b>	Software modules like Mode Manager, Network Management, or DCM might be interested in the FlexRay CC POC status.
<b>Use Case:</b>	--
<b>Dependencies:</b>	[BSW05022] Get FlexRay CC POC Status
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.3.5.2 [BSW05121] Get FlexRay CC Sync State

<b>ID:</b>	BSW05121
<b>Initiator:</b>	DECOMSYS
<b>Date:</b>	02.02.2005
<b>Short Description:</b>	Get the sync state of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall provide a software interface to get the sync state ("controller synchronous"/"controller asynchronous") of a specific FlexRay CC.
<b>Rationale:</b>	In order to be able to communicate via FlexRay, it is crucial that the FlexRay CC be synchronous to the FlexRay Cluster's global time. Software modules

	like Mode Manager, Network Management, or DCM might be interested in the sync state of a specific FlexRay CC.
<b>Use Case:</b>	A distributed control might only be started once the communication via the FlexRay bus is possible. Therefore, knowledge about the sync state of the FlexRay CC is crucial
<b>Dependencies:</b>	<a href="#">[BSW05023]</a> Get FlexRay CC Sync State
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.3.6 Normal Operation

#### 5.1.3.6.1 [BSW05106] Buffer Reconfiguration in Normal Active Mode

<b>ID:</b>	BSW05106
<b>Initiator:</b>	BMW
<b>Date:</b>	26.07.2004
<b>Short Description:</b>	Reconfiguration of a specific transmit/receive buffer of a specific FlexRay Communication Controller in normal active mode, if supported by the FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	<p>The FlexRay Driver shall be able to reconfigure a specific transmit/receive buffer of a specific FlexRay Communication Controller in normal active mode, if supported by the FlexRay CC.</p> <p>The reconfiguration of a transmit/receive buffer of a FlexRay CC in normal active mode allows providing to the FlexRay Interface a higher number of (virtual) transmit/receive buffers than this FlexRay CC actually has.</p> <p>The configuration tool of the FlexRay Driver shall schedule these reconfiguration actions whenever it is necessary before/after the FlexRay Interface accesses a transmit/receive buffer of a FlexRay CC.</p> <p>The reconfiguration of a transmit/receive buffer of a FlexRay CC in normal active mode means that the same buffer is being used for different buffer configurations (i.e. Frame-ID and filter configuration, transmitting/sending, Channel A/B) without leaving normal active mode (i.e. without setting the FlexRay CC into configuration mode).</p> <p>In the dynamic segment, each transmit/receive buffer of a FlexRay CC shall be used maximally once per Cycle. This limits the reconfiguration possibilities and thus restricts the number of transmittable (sent and received) Frames per dynamic segment to the accumulated number (over all CCs on one ECU) of transmit/receive buffers connected to one Cluster.</p> <p>For safety reasons it shall be possible to completely deactivate this feature <a href="#">pre compile time</a>.</p>
<b>Rationale:</b>	The number of transmit/receive buffers affects the HW cost, therefore, the supplier offers low-cost controllers with a small number of transmit/receive buffers. And by using only dedicate transmit/receive buffer configurations, each small application would need a lot of buffer. Therefore, the buffer reconfiguration during normal active mode shall be supported by the FlexRay Driver.
<b>Use Case:</b>	For example, this can be used to send many PDUs with slow communication requirements in the dynamic segment. Non-safety-critical use, in gateways, etc.

	Another example (in the static segment): PDU with a period of 1.25ms in a 2.5 ms Cycle length scheduled in the Static Slot 5 and 45. The same buffer should be use to send (or receive) the Slot with the Frame-ID 5 and 45 with the buffer reconfiguration in normal active mode.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00273

### 5.1.3.6.2 [BSW005106] Trigger Configuration of FR CC Buffer with non-static Configuration

<b>ID:</b>	BSW005106
<b>Initiator:</b>	BMW
<b>Date:</b>	31.01.2008
<b>Short Description:</b>	Trigger Configuration of FlexRay CC Buffer with non-static Configuration
<b>Importance:</b>	high   medium   low
<b>Description:</b>	<p>It shall be possible to perform a CC Buffer reconfiguration at runtime (i.e. it is not being defined during system configuration time)</p> <p>The set of hardware-independent configuration parameters consists of the following items:</p> <ul style="list-style-type: none"> <li>• Identifier of the CC Buffer</li> <li>• Direction: Transmission or Reception</li> <li>• FlexRay Slot Number</li> <li>• Cycle Counter Offset</li> <li>• Cycle Counter Repetition</li> <li>• FlexRay Channel</li> <li>• Payload Length of FlexRay Frame</li> <li>• FlexRay Header CRC</li> </ul>
<b>Rationale:</b>	- Dynamic bandwidth assignment to the XCP slaves by the XCP master.
<b>Use Case:</b>	- Manipulation of parameters for adjustment purpose (e.g. chassis suspension, motor management,..)
<b>Dependencies:</b>	FrIf, FrDrv
<b>Conflicts:</b>	-
<b>Supporting Material:</b>	<a href="http://www.asam.net/doc_int/getfile/getfile.php?id=376">http://www.asam.net/doc_int/getfile/getfile.php?id=376</a>
<b>Contributes to:</b>	BRF00273 BRF00279

### 5.1.3.6.3 [BSW05125] Interrupt Handling

<b>ID:</b>	BSW05125
<b>Initiator:</b>	BMW
<b>Date:</b>	16.02.2005
<b>Short Description:</b>	The FlexRay Driver shall provide services to handle interrupts of a FlexRay Communication Controller.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	<p>The FlexRay Driver shall provide services to handle interrupts of a FlexRay Communication Controller. At least the following functionality shall be provided:</p> <ul style="list-style-type: none"> <li>• get interrupt source</li> <li>• service interrupt</li> <li>• enable interrupt</li> </ul>

	<ul style="list-style-type: none"> <li>• disable interrupt</li> <li>• acknowledge interrupt.</li> </ul>
<b>Rationale:</b>	Only the FlexRay Driver has knowledge on how to handle (e.g. enable/disable) a CC-specific interrupt.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05174]</a> Interrupt Handling
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.3.7 Shutdown Operation

#### 5.1.3.7.1 [BSW05114] Abortion of FlexRay CC Communication

<b>ID:</b>	BSW05114
<b>Initiator:</b>	BMW
<b>Date:</b>	02.02.2005
<b>Short Description:</b>	Immediate abortion of the communication of a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall provide a software interface to immediately abort of the communication of a specific FlexRay CC by setting this FlexRay CC into the "HALT" state. Thus, the FlexRay CC will stop participating in the communication and lose synchronicity with the FlexRay global time.
<b>Rationale:</b>	--
<b>Use Case:</b>	Error confinement
<b>Dependencies:</b>	<a href="#">[BSW05016]</a> Abortion of a FlexRay CC Communication
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.3.7.2 [BSW05115] Halt of FlexRay CC Communication

<b>ID:</b>	BSW05115
<b>Initiator:</b>	BMW
<b>Date:</b>	02.02.2005
<b>Short Description:</b>	Halting of the communication of a specific FlexRay CC at the end of the current Communication Cycle.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Driver shall provide a software interface to halt the communication of a specific FlexRay CC at the end of the current Communication Cycle by setting this FlexRay CC into the "HALT" state at the end of the current Communication Cycle. Thus, this FlexRay CC will stop participating in the communication and lose synchronicity with the FlexRay global time.
<b>Rationale:</b>	--
<b>Use Case:</b>	Error confinement
<b>Dependencies:</b>	<a href="#">[BSW05063]</a> Halt of a FlexRay CC Communication
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.3.7.3 [BSW005207] API to Detect Error Information in Transmitted Frame

<b>ID:</b>	BSW005207
------------	-----------

<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	API to Detect Error Information in Transmitted Frame
<b>Importance:</b>	<b>High</b>   medium   low
<b>Description:</b>	The FlexRay Driver shall provide an API that detects the following error status in transmitted frames and notifies them to application level. <ul style="list-style-type: none"> <li>● vSS!SyntaxError</li> <li>● vSS!ContentError</li> <li>● vSS!BViolation</li> </ul>
<b>Rationale:</b>	The FlexRay modules should provide information that only the modules can detect.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	<a href="#">[BSW05175]</a> Provide Error Information
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00266 BRF00302

#### 5.1.3.7.4 [BSW005208] API to Detect Error Information in Received Frame

<b>ID:</b>	BSW005208
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	API to Detect Error Information in Received Frame
<b>Importance:</b>	<b>High</b>   medium   low
<b>Description:</b>	The FlexRay Driver shall provide an API that detects the following error information in received frames and notifies them to application level. <ul style="list-style-type: none"> <li>● vSS!SyntaxError</li> <li>● vSS!ContentError</li> <li>● vSS!BViolation</li> </ul>
<b>Rationale:</b>	The FlexRay modules should provide information that only the modules can detect.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	<a href="#">[BSW05175]</a> Provide Error Information
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00266 BRF00304

#### 5.1.3.7.5 [BSW005209] API to Detect Error Information in NIT

<b>ID:</b>	BSW005209
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	API to Detect Error Information in NIT
<b>Importance:</b>	<b>High</b>   medium   low
<b>Description:</b>	The FlexRay Driver shall provide an API that detects the following error information in NIT and notifies them to application level. <ul style="list-style-type: none"> <li>● vSS!SyntaxError</li> <li>● vSS!BViolation</li> </ul>
<b>Rationale:</b>	The FlexRay modules should provide information that only the modules can detect.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	<a href="#">[BSW05175]</a> Provide Error Information
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00266 BRF00304

### 5.1.3.7.6 [BSW005210] Criteria on Validness of Received Frame

<b>ID:</b>	BSW005210
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Criteria on Validness of Received Frame
<b>Importance:</b>	<b>High</b>   medium   low
<b>Description:</b>	<p>The FlexRay Driver shall decide whether the received frame is valid in accordance with a criteria setting in pre-compiled parameters.</p> <p>When the criteria is set to a loose mode, the following state would be checked. The frame would be acknowledged as a valid one if ValidFrame is true.</p> <ul style="list-style-type: none"> <li>● vSS!ValidFrame</li> </ul> <p>When the criteria is set to a strict mode, the following states would be checked. The frame would be acknowledged as a valid one if ValidFrame is true and all the other error statuses are false.</p> <ul style="list-style-type: none"> <li>● vSS!ValidFrame</li> <li>● vSS!SyntaxError</li> <li>● vSS!ContentError</li> <li>● vSS!BViolation</li> </ul>
<b>Rationale:</b>	The FlexRay Interface should provide a method to strictly check the error information in frames.
<b>Use Case:</b>	This ensures the network reliability.
<b>Dependencies:</b>	<a href="#">[BSW05201]</a> Error Information in Received Frame
<b>Conflicts:</b>	<a href="#">[BSW05175]</a> Provide Error Information
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00266 BRF00304

### 5.1.3.8 Fault Operation

#### 5.1.3.8.1 [BSW005211] Hardware Checking Function on CC

<b>ID:</b>	BSW005211
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Hardware Checking Function on CC
<b>Importance:</b>	<b>high</b>   medium   low
<b>Description:</b>	FlexRay Driver shall check whether CC is working properly at its initialization phase and notify error to application when illegal status is detected.
<b>Rationale:</b>	This functionality ensures that the hardware is working as expected.
<b>Use Case:</b>	Improvement of hardware reliability.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00306

#### 5.1.3.8.2 [BSW005222] Support of Transceiver Rx-Only Mode via FR State Manager

<b>ID:</b>	BSW005222
<b>Initiator:</b>	DAI
<b>Date:</b>	27.01.2009
<b>Short Description:</b>	Support of diagnostic mode "silent" (switch off all Tx communication).
<b>Type:</b>	New
<b>Importance:</b>	high   <b>medium</b>   low
<b>Description:</b>	This mode change should be visible to the Mode Management, but not to the application. The application should still get FullCommunication indicated as if the ECU is able to send frames on the bus. TxConfirmation for send



	requests shall still be performed during ECU "silent" mode. The "silent" mode should persist even when the bus is shut down and woken up again. After a reset the ECU should be able to send frames on the bus again.
<b>Rationale:</b>	It should be possible to switch off the Tx path of an ECU to send in its slots without having any collision on the bus.
<b>Use Case:</b>	Set a faulty ECU to "silent" to have the possibility to simulate the Tx path of the ECU while the connected actuators and sensors are still working.
<b>Dependencies:</b>	FlexRay Transceiver hardware which offers the optional Rx-Only State
<b>Conflicts:</b>	FlexRay Transceiver hardware which does not offer the Rx-Only State
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

## 5.1.4 Transport Layer FlexRay

### 5.1.4.1 Configuration

#### 5.1.4.1.1 [BSW05077] Unique Identifier of N-SDU

<b>ID:</b>	BSW05077
<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Identification of each N-SDU with a unique identifier.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transport Layer shall identify each N-SDU with a unique identifier, so the upper layer can address an N-SDU without any assumption on the addressing mode configuration of the FlexRay Transport Layer. Furthermore, a symbolic name may be assigned for each N-SDU identifier value to simplify usage of the API.
<b>Rationale:</b>	Independence of upper layer with the FlexRay Transport Layer addressing particularities, optimization. The PDU-Router routes all N-SDUs (FlexRay, CAN and LIN) regardless of the addressing mode particularities of the underlying protocols.
<b>Use Case:</b>	FlexRay – CAN TP gateway
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[ISO 15765-2]</a> and <a href="#">[ISO 15765-4]</a> specification.
<b>Contributes to:</b>	--

#### 5.1.4.1.2 [BSW05079] Transport Connection Properties

<b>ID:</b>	BSW05079
<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Individual properties of each N-SDU.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transport Layer configuration shall be defined <a href="#">at system configuration time</a> and shall define the following properties individually for each N-SDU (i.e. for each connection): <ul style="list-style-type: none"> <li>• Its unique identifier</li> <li>• Associated N-PDU identifiers</li> </ul>

	<ul style="list-style-type: none"> <li>• Size of the associated N-PDU(s)</li> <li>• Acknowledge type</li> <li>• Transmit cancellation ON/OFF</li> <li>• Target address</li> <li>• Source address</li> <li>• Connection type (1:1 or 1:n)</li> <li>• Timeout parameters according to ISO 10681-2</li> <li>• Maximum PDU length</li> </ul>
<b>Rationale:</b>	At runtime the FlexRay Transport module must have all the information required to manage a Transport Layer connection.
<b>Use Case:</b>	This information can be used at generation time to check the network configuration with a FlexRay Transport Layer point of view.
<b>Dependencies:</b>	<a href="#">[BSW05129]</a> Mismatch of Service Call and Connection Properties
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	ISO 10681-2
<b>Contributes to:</b>	BRF00252

## 5.1.4.2 Initialization

### 5.1.4.2.1 [BSW05088] FlexRay Transport Layer Initialization

<b>ID:</b>	BSW05088
<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Interface for initialization.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transport Layer shall implement an interface for initialization in order to initialize all global variables of the FlexRay Transport Layer module and set all Transport Layer connections in a default state (Idle).
<b>Rationale:</b>	Basic functionality.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.4.2.2 [BSW05089] FlexRay Transport Layer Availability

<b>ID:</b>	BSW05089
<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	The FlexRay Transport Layer services shall not be operational before initializing the module.
<b>Type:</b>	New
<b>Importance:</b>	Medium
<b>Description:</b>	Before using the transmission capabilities of the Flex-Ray Transport Layer, it shall be initialized. If this is not the case, the services have to raise an error in development mode.
<b>Rationale:</b>	Basic functionality.
<b>Use Case:</b>	To avoid usage of the module without a complete initialization this could cause the transmission of corrupted Frames.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--

<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.4.3 Normal Operation

#### 5.1.4.3.1 [BSW05090] Support of Optional ISO10681-2 Service

<b>ID:</b>	BSW05090
<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Support of the Change Parameter Service according to ISO10681-2.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transport Layer shall support per connection (i.e. per N-SDU) the ISO 10681-2 service N_ChangeParameter. Thus, it is possible to adapt the BC to the receiver's needs.
<b>Rationale:</b>	In the case of Inter-ECU communication it could be necessary to adapt the BandwidthControl parameter to the individual needs of the receiver.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Caveats:</b>	--
<b>Supporting Material:</b>	ISO 10681-2 specification.
<b>Contributes to:</b>	BRF00252

#### 5.1.4.3.2 [BSW05093] Transmit Cancellation

<b>ID:</b>	BSW05093
<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Provide a transmit cancellation service to the sender and the receiver.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transport Layer shall provide to the sender a cancellation service of transmission at any point in time It shall be configurable <a href="#">at system configuration time</a> whether this feature is provided by the FlexRay Transport Layer.
<b>Rationale:</b>	Cancellation of pending transmission.
<b>Use Case:</b>	The higher layer can cancel a pending transmission by using this service.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.4.3.3 [BSW05095] Bandwidth Control

<b>ID:</b>	BSW05095
<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Dynamic control of used bandwidth.
<b>Type:</b>	Changed
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transport Layer shall support the dynamic bandwidth control mechanism as described in the ISO 10681-2
<b>Rationale</b>	Adapting the bandwidth control parameters as used in ISO-TP to FlexRay

	circumstances.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	ISO10681-2specification.
<b>Contributes to:</b>	BRF00252

#### 5.1.4.4 Fault Operation

##### 5.1.4.4.1 [BSW05129] Mismatch of Service Call and Connection Properties

<b>ID:</b>	BSW05129
<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	15.03.2005
<b>Short Description:</b>	Detection of mismatch of service call and connection properties.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transport Layer shall detect the mismatch of a service call by the upper layer and the corresponding connection properties (e.g. a transmit request which would cause segmentation for a configured 1:n connection,) and shall indicate this to the caller.
<b>Rationale:</b>	The connection specific configuration parameters have impact on the validity range of the parameters passed to the FlexRay Transport Layer API during runtime. Since these ranges cannot be checked <a href="#">at system configuration time</a> , this check has to take place during runtime.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05079]</a> Transport Connection Properties
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.5 FlexRay Time Service

##### 5.1.5.1 General Requirements

###### 5.1.5.1.1 [BSW05033] Tick Conversion

<b>ID:</b>	BSW05033
<b>Initiator:</b>	BMW
<b>Date:</b>	09.08.2004
<b>Short Description:</b>	Conversion between macroticks and nanoseconds.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to carry out a conversion between FlexRay macroticks and nanoseconds in both directions.
<b>Rationale:</b>	The upper-layer SW module does not know the duration of the FlexRay specific macrotick. Therefore, the FlexRay software modules shall provide an API to carry out the conversion.
<b>Use Case:</b>	--

<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.5.1.2 [BSW05053] Cluster External Clock Synchronization

<b>ID:</b>	BSW05053
<b>Initiator:</b>	BMW
<b>Date:</b>	02.06.2004
<b>Short Description:</b>	Apply rate and offset correction terms to a specific FlexRay Cluster.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to apply rate and offset correction terms (according to chapter 8.6.5 of <a href="#">[FR_PROT_SPEC]</a> ) to a specific FlexRay Cluster, i.e. to all FlexRay Communication Controllers connected to this Cluster.
<b>Rationale:</b>	--
<b>Use Case:</b>	An application may need to maintain the synchronicity between two FlexRay Clusters.
<b>Dependencies:</b>	<a href="#">[BSW05156]</a> Controller External Clock Synchronization
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.5.1.3 [BSW05156] Controller External Clock Synchronization

<b>ID:</b>	BSW05156
<b>Initiator:</b>	BMW
<b>Date:</b>	16.01.2006
<b>Short Description:</b>	Apply rate and offset correction terms to a specific FlexRay Communication Controller.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to apply rate and offset correction terms (according to chapter 8.6.5 of <a href="#">[FR_PROT_SPEC]</a> ) to a specific FlexRay Communication Controller.
<b>Rationale:</b>	--
<b>Use Case:</b>	An application may need to maintain the synchronicity between two FlexRay Clusters.
<b>Dependencies:</b>	<a href="#">[BSW05053]</a> Cluster External Clock Synchronization
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.5.2 Configuration

#### 5.1.5.2.1 [BSW05044] Set Absolute Timer

<b>ID:</b>	BSW05044
<b>Initiator:</b>	BMW
<b>Date:</b>	13.07.2004
<b>Short Description:</b>	Set the absolute timer of a FlexRay Communication Controller.

<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to set a FlexRay Communication Controller's absolute timer in order to program a timer interrupt ("alarm") at a defined point in (the FlexRay global) time specified by a pair of cycle time (in units of macroticks) and cycle count.
<b>Rationale:</b>	--
<b>Use Case:</b>	Absolute alarms can be used to synchronize software modules to the global time of a FlexRay node. The scheduler of a time driven operating system can be triggered by an absolute alarm, thereby synchronizing the operating system to the global time of a FlexRay node.
<b>Dependencies:</b>	The interrupt service routine for the alarm has to be registered with an operating system.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.5.3 Normal Operation

#### 5.1.5.3.1 [BSW05046] Enable Absolute Alarms

<b>ID:</b>	BSW05046
<b>Initiator:</b>	BMW
<b>Date:</b>	13.07.2004
<b>Short Description:</b>	Enable the absolute alarm(s) of a FlexRay Communication Controller.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to enable the absolute alarm(s) of a FlexRay Communication Controller.
<b>Rationale:</b>	--
<b>Use Case:</b>	Higher software layers may want to (re)-enable an absolute alarm.
<b>Dependencies:</b>	The operating system has to provide a service to enable interrupts.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.5.3.2 [BSW05047] Disable Absolute Alarms

<b>ID:</b>	BSW05047
<b>Initiator:</b>	BMW
<b>Date:</b>	13.07.2004
<b>Short Description:</b>	Disable the absolute alarm(s) of a FlexRay Communication Controller.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to disable the absolute alarm(s) of a FlexRay Communication Controller.
<b>Rationale:</b>	--
<b>Use Case:</b>	Higher software layers may want to disable an absolute alarm temporarily.
<b>Dependencies:</b>	The operating system has to provide a service to disable interrupts.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.5.3.3 [BSW05048] Acknowledge Absolute Alarms

<b>ID:</b>	BSW05048
<b>Initiator:</b>	BMW
<b>Date:</b>	13.07.2004
<b>Short Description:</b>	Acknowledge the absolute alarm(s) of a FlexRay Communication Controller.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to acknowledge the absolute alarm(s) of a FlexRay Communication Controller, i.e. to clear the interrupt condition.
<b>Rationale:</b>	If in case of level-sensitive interrupts, an alarm is not acknowledged within the ISR (i.e. the interrupt request flag is not being reset) an interrupt service will be requested immediately again after leaving the ISR.
<b>Use Case:</b>	Higher software layers may use this function to reset the absolute alarm condition.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.5.3.4 [BSW05049] Enable Relative Alarms

<b>ID:</b>	BSW05049
<b>Initiator:</b>	BMW
<b>Date:</b>	13.07.2004
<b>Short Description:</b>	Enable the relative alarm(s) of a FlexRay Communication Controller, if supported by the FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to enable the relative alarm(s) of a FlexRay Communication Controller, if supported by the FlexRay CC.
<b>Rationale:</b>	--
<b>Use Case:</b>	Higher software layers may want to (re)-enable a relative alarm.
<b>Dependencies:</b>	The operating system has to provide a service to enable interrupts.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.5.3.5 [BSW05050] Disable Relative Alarms

<b>ID:</b>	BSW05050
<b>Initiator:</b>	BMW
<b>Date:</b>	13.07.2004
<b>Short Description:</b>	Disable the relative alarm(s) of a FlexRay Communication Controller, if supported by the FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to disable the relative alarm(s) of a FlexRay Communication Controller, if supported by the FlexRay CC.
<b>Rationale:</b>	--
<b>Use Case:</b>	Higher software layers may want to disable a relative alarm temporarily.
<b>Dependencies:</b>	The operating system has to provide a service to disable interrupts.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--

<b>Contributes to:</b>	--
------------------------	----

#### 5.1.5.3.6 [BSW05051] Acknowledge Relative Alarms

<b>ID:</b>	BSW05051
<b>Initiator:</b>	BMW
<b>Date:</b>	13.07.2004
<b>Short Description:</b>	Acknowledge the relative alarm(s) of a FlexRay Communication Controller, if supported by the FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to acknowledge the relative alarm(s) of a FlexRay Communication Controller, if supported by the FlexRay CC, i.e. to clear the interrupt condition.
<b>Rationale:</b>	If in case of level-sensitive interrupts, an alarm is not acknowledged within the ISR (i.e. the interrupt request flag is not being reset) an interrupt service will be requested immediately again after leaving the ISR.
<b>Use Case:</b>	Higher software layers may use this function to reset the relative alarm condition.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.5.3.7 [BSW05052] Get Cycle Length in Macroticks

<b>ID:</b>	BSW05052
<b>Initiator:</b>	BMW
<b>Date:</b>	02.06.2004
<b>Short Description:</b>	Get the FlexRay Cycle length of a specific FlexRay Cluster.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to get the length of a Communication Cycle (in macroticks) of a specific FlexRay Cluster.
<b>Rationale:</b>	--
<b>Use Case:</b>	This is for example required to set up a (2 <sup>n</sup> Cycles) repetitive relative alarm with the repetition rate of one Cycle.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.5.3.8 [BSW05019] Get FlexRay Global Time

<b>ID:</b>	BSW05019
<b>Initiator:</b>	BMW
<b>Date:</b>	29.07.2004
<b>Short Description:</b>	Get the FlexRay global time from a specific FlexRay CC.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay software modules shall provide a software interface to get the FlexRay global time from a specific FlexRay CC, specified by a pair of cycle time (in units of macroticks) and cycle count.
<b>Rationale:</b>	The FlexRay global time may be used for the synchronization of the upper-layer software. It can also be read by an application running synchronously or asynchronously to the FlexRay global time.
<b>Use Case:</b>	--



<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

## 5.1.5.4 Fault Operation

### 5.1.5.4.1 [BSW05071] FlexRay Time Services Access if CC is in Halt

<b>ID:</b>	BSW05071
<b>Initiator:</b>	BMW
<b>Date:</b>	26.01.2005
<b>Short Description:</b>	Error on time services if CC is in "HALT" state.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	If a time services function is called for a specific FlexRay CC after the communication of this FlexRay CC has been halted or aborted (i.e. the FlexRay CC is in "HALT" state), the FlexRay Driver shall raise an error.
<b>Rationale:</b>	Time service functions only make sense when the FlexRay global time is available, thus when the FlexRay CC is not in "HALT" state.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.5.4.2 [BSW05072] FlexRay Time Services Access if CC is Out of Sync

<b>ID:</b>	BSW05072
<b>Initiator:</b>	BMW
<b>Date:</b>	26.01.2005
<b>Short Description:</b>	Error on time services if CC is not synchronous.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	If a time services function is called for a specific FlexRay CC which is not synchronous to its Cluster, the FlexRay Driver shall raise an error.
<b>Rationale:</b>	Time service functions only make sense when the FlexRay global time is available, thus when the FlexRay CC is synchronous.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

## 5.1.6 FlexRay Transceiver Driver

### 5.1.6.1 Configuration

#### 5.1.6.1.1 [BSW05131] Configuration Data for FlexRay Transceiver

<b>ID:</b>	BSW05131
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The transceiver driver package shall include a description file with the basic information needed to configure the driver for a given bus and the supported notifications.
<b>Type:</b>	New
<b>Importance:</b>	Medium
<b>Description:</b>	This file shall clarify whether the following features are supported: <ul style="list-style-type: none"> <li>• Single and/or multiple instances of a given transceiver device</li> <li>• Maximally supported baud rate of each bus to enable the detection of configuration errors</li> <li>• Wake-up by bus</li> <li>• Control of power supply possible by the transceiver</li> <li>• List of errors</li> <li>• Transceiver control via SPI or port pin</li> <li>• Which notifications are supported</li> <li>• Call context of the notification functions (ISR, polling) to enable detection of necessary data consistency mechanisms during configuration time</li> <li>• Transceiver supports bus error detection (if not, a state query will always return &lt;good&gt;)</li> </ul>
<b>Rationale:</b>	Configuration
<b>Use Case:</b>	Basic functionality for transceiver configuration.
<b>Dependencies:</b>	ECU resource template
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.6.1.2 [BSW05132] Support for More than One FlexRay Transceiver

<b>ID:</b>	BSW05132
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver shall support the configuration for more than one transceiver type as well as for more than one Cluster <a href="#">pre compile time</a> .
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The driver shall be able to support multiple FlexRay Clusters on the ECU. It must be possible to configure the used transceiver device independently for each Cluster. This includes also mixed systems with e.g. two FlexRay Clusters using different bus physics.
<b>Rationale:</b>	Systems with two FlexRay Clusters.
<b>Use Case:</b>	Basic functionality for transceiver configuration.
<b>Dependencies:</b>	ECU resource template
<b>Conflicts:</b>	Transceiver handling depends strongly on the used device. Therefore each transceiver may needs its own implementation within the driver and only known and supported devices could be selected. A general solution for the FlexRay Transceiver Driver for all use cases might not be possible.  FlexRay: By default each FlexRay controller is attached to an own Cluster

	and needs therefore an own transceiver. In some special cases more than one FlexRay controller is attached to the same Cluster to increase the number of buffers.
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.6.1.3 [BSW05133] Configuration of Bus Operation Mode after Initialization for each FlexRay Transceiver

<b>ID:</b>	BSW05133
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver shall support the independent configuration of the bus operation mode for each supported Cluster.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	Due to the different startup requirements on a multiple-FlexRay-Cluster-ECU, the FlexRay Transceiver Driver support the independent pre-selection of the bus operation mode to which each FlexRay Transceiver device is set during the driver initialization.
<b>Rationale:</b>	Multi-Cluster systems
<b>Use Case:</b>	Basic functionality for transceiver configuration.
<b>Dependencies:</b>	See needs described for FlexRay Interface, ECU state manager, ComM and NM.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.6.1.4 [BSW05134] Initialization Sequence for FlexRay Transceiver Driver

<b>ID:</b>	BSW05134
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver shall support the configuration sequence of the AUTOSAR stack.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	To start the ECU from power-up or reset, a fix sequence of driver and manager initialization is necessary to reach the required startup times and to set the FlexRay stack into working state. The sequence itself depends on a lot of requirements, partly dependent on the FlexRay controller and the power supply concept.
<b>Rationale:</b>	Correct ECU startup behavior
<b>Use Case:</b>	Basic functionality for transceiver configuration.
<b>Dependencies:</b>	NM, ECU state manager, wake-up concepts of ECU state manager and ComManager.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.6.1.5 [BSW05136] Configuration "Notification for Wake-up by Bus"

<b>ID:</b>	BSW05136
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005

<b>Short Description:</b>	The FlexRay Transceiver Driver shall support the compile time configuration of one notification to a higher layer for change notification for "wake-up by bus" events.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	One wake-up by bus event notification shall be supported to one higher layer.  If a transceiver device does not support "wake-up by bus", this notification is never called for this bus.
<b>Rationale:</b>	Efficient coupling between FlexRay Transceiver Driver and higher layer.
<b>Use Case:</b>	See <a href="#">[BSW05147]</a>
<b>Dependencies:</b>	Higher layer, i.e. one of (bus specific) NM or ECU state manager. <a href="#">[BSW05147]</a>
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

## 5.1.6.2 Initialization

### 5.1.6.2.1 [BSW05137] Initialize the FlexRay Transceiver Driver

<b>ID:</b>	BSW05137
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver shall provide an API to initialize the driver internally and set then all attached FlexRay Transceivers in their pre-selected operation modes.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transceiver Driver must be initialized during the power-up/reset sequence of the ECU. Depending on the used drivers to control the transceivers (e.g. DIO, SPI), they must be already available and working when the FlexRay Transceiver Driver is initialized. The wake-up reason has to be detected and stored during the execution of the driver initialization, too.
<b>Rationale:</b>	Set FlexRay Transceivers and FlexRay Transceiver Driver in a pre-defined and known state
<b>Use Case:</b>	Basic functionality for transceiver control.
<b>Dependencies:</b>	SPI and DIO driver initialization. <a href="#">[BSW05144]</a> The FlexRay Transceiver Driver setup information must provide the necessary configuration data to enable the generation tool to select the appropriate control mechanism (e.g. SPI, I/O ports) and to guarantee the correct allocation of the necessary communication resources and initialization sequences.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.6.3 Normal Operation

#### 5.1.6.3.1 [BSW05138] FlexRay Transceiver Driver API shall be Synchronous

<b>ID:</b>	BSW05138
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver API shall be synchronous.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transceiver Driver API shall execute the requested action immediately and shall deliver the result state immediately to the caller.  This will ease up the implementation of wake-up and sleep concepts within the AUTOSAR BSW stack.
<b>Rationale:</b>	Better usage of transceiver functionality in the complex AUTOSAR BSW environment.
<b>Use Case:</b>	Atomic transition to other operation mode; easier and better abstraction for higher layers like the ECU state manager or ComManager. Improved testability compared to asynchronous handling.
<b>Dependencies:</b>	ECU state manager, NM
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.6.3.2 [BSW05166] Set FlexRay Transceiver Operation Mode

<b>ID:</b>	BSW05166
<b>Initiator:</b>	BMW
<b>Date:</b>	02.02.2006
<b>Short Description:</b>	Set the operation mode of a specific FlexRay Transceiver device.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transceiver Driver shall provide a software interface to set the operation mode of a specific FlexRay Transceiver device. According to <a href="#">[FR_EPL_SPEC]</a> , at least these operation modes shall be supported: <ul style="list-style-type: none"> <li>• BD_Normal</li> <li>• BD_Standby</li> <li>• BD_Receive_only</li> <li>• BD_Sleep</li> </ul>
<b>Rationale:</b>	Provide ComM with a possibility to set the operation mode of a specific FlexRay Transceiver device.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05039]</a> Set FlexRay Transceiver Operation Mode <a href="#">[BSW05162]</a> Set Cluster-wide FlexRay Transceiver Operation Mode
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_EPL_SPEC]</a>
<b>Contributes to:</b>	--

#### 5.1.6.3.3 [BSW05167] Get FlexRay Transceiver Operation Mode

<b>ID:</b>	BSW05167
<b>Initiator:</b>	BMW
<b>Date:</b>	02.02.2006
<b>Short Description:</b>	Get the operation mode of a specific FlexRay Transceiver device.

<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	<p>The FlexRay Transceiver Driver shall provide a software interface to get the operation mode of a FlexRay Transceiver device.</p> <p>According to <a href="#">[FR_EPL_SPEC]</a>, at least these operation modes shall be supported:</p> <ul style="list-style-type: none"> <li>• BD_Normal</li> <li>• BD_Standby</li> <li>• BD_Receive_only</li> <li>• BD_Sleep</li> </ul>
<b>Rationale:</b>	Provide ComM with a possibility to get the operation mode of a specific FlexRay Transceiver device.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05157]</a> Get FlexRay Transceiver Operation Mode
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_EPL_SPEC]</a>
<b>Contributes to:</b>	--

#### 5.1.6.3.4 [\[BSW05144\]](#) Get FlexRay Transceiver Wake-up Reason

<b>ID:</b>	BSW05144
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	Get the wake-up reason of a specific FlexRay Transceiver device.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	<p>The FlexRay Transceiver Driver shall provide a software interface to get the wake-up reason of a specific FlexRay Transceiver device.</p> <p>The FlexRay Transceiver Driver shall be able to store the local view "who has requested the wake-up: bus or internally".</p> <ul style="list-style-type: none"> <li>• <b>Bus:</b> The bus has caused the wake-up.</li> <li>• <b>Internally:</b> The wake-up has been caused by an internal request to the driver.</li> <li>• <b>Sleep:</b> The transceiver is in operation mode sleep and no wake-up has been occurred.</li> <li>• <b>Wake pin:</b> An edge on the wake pin of the transceiver (if present) has caused the wakeup.</li> </ul> <p>The wake-up reason should be "sleep" when the operation mode is not Normal and no wake-up has been occurred.</p> <p>When a wake-up has occurred, the API must always return the first detected wake-up reason (e.g. if a wake-up by bus occurs and then nearly at the same time an internal wake-up, the wake-up reason is "bus".).</p> <p>After leaving the operation mode Normal, the wake-up reason shall be set to "sleep" again.</p>
<b>Rationale:</b>	Detection of wake-up reason during development and via diagnostic command. May also be used by the NM or ECU state manager.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05158]</a> Get FlexRay Transceiver Wake-up Reason
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.6.3.5 [BSW05147] Notification for Wake-up by Bus

<b>ID:</b>	BSW05147
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver shall support a notification to inform higher layers about the wake-up by bus. It must be possible to support more than one bus within the ECU with this notification.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transceiver Driver shall call this notification when the transceiver detects a wake-up by bus.  The FlexRay Transceiver Driver is notified by a notification from the underlying SPI or DIO driver in that case. The notification is executed in the context of the caller (may be interrupt context!). Due to the delay from wake-up detection till the start of the necessary actions have a large influence to the startup time of an ECU, this event shall be processed internally and transferred immediately via this notification to the next layer.  The call context and the reaction time depend on the call context of the lower layer DIO or SPI. In case of interrupt it is very fast but data consistency issues must be covered in all layers, in case of polling data consistency issues are reduced but reaction time may be to slow.
<b>Rationale:</b>	Support wake-up by FlexRay Transceiver devices.
<b>Use Case:</b>	The FlexRay Transceiver detects a wake-up condition on the bus and shows this to the $\mu$ C via e.g. a port pin. Further handling depends on current ECU state. Assumed the ECU is halted, the change on the port may terminate the "HALT" statement and let the processor continue its work. The assigned port interrupt will be executed and this handler is called. Now, the FlexRay Transceiver Driver will store the wake-up reason and give the call via this notification to e.g. the NM to let the NM decide how to handle the event.  See [BSW05136] Configuration "Notification for Wake-up by Bus" for details, too.
<b>Dependencies:</b>	DIO and SPI driver for notification, one of (bus specific) NM, diagnostics or ECU state manager as client. <a href="#">[BSW05136]</a>
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.1.6.3.6 [BSW05148] Support for Wake-up During Sleep Transition

<b>ID:</b>	BSW05148
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver shall support situations where a wake-up by bus occurs at the same moment the transition to standby/sleep is executed by the driver.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	Wake-up by bus is always asynchronous to the internal transition to sleep. In worst case, the wake-up occurs during the transition to sleep. This situation must be covered by the design and explicitly tested for each ECU.  The driver shall create a wake-up notification by bus immediately after the

	API to enter the standby/sleep mode has finished. The calling/controlling component (NM or ECU state manager) must be capable to handle the wake-up immediately after requesting the standby/sleep.
<b>Rationale:</b>	Safe wake-up and sleep handling.
<b>Use Case:</b>	All busses with a wake-up by bus are affected.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.6.3.7 [BSW05149] Support API to Enable/Disable and Clear Wake-up Events

<b>ID:</b>	BSW05149
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver shall support an API to enable and disable the wake-up notification for each bus separately.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	To enable higher layers to command the FlexRay Transceiver device safe into its standby and/or sleep state, an additional API to disable and enable the wake-up notification is necessary.  If the notification is disabled, driver shall not perform the notification but store the event internally until the notification is enabled again. The notification shall then be processed immediately. It shall be possible to clear a pending wake-up event. If no further wake-up event occurs, no notification shall be performed after enabling the notification again. If a further wake-up event occurs it shall be notified.
<b>Rationale:</b>	Safe wake-up and sleep handling.
<b>Use Case:</b>	All busses with a wake-up by bus are affected.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.6.3.8 [BSW005135] Support for active FlexRay Stars

<b>ID:</b>	BSW005135
<b>Initiator:</b>	GM
<b>Date:</b>	31.01.2008
<b>Short Description:</b>	Support for active FlexRay stars
<b>Importance:</b>	high   medium   low
<b>Description:</b>	Active Stars are part of valid FlexRay topologies.
<b>Rationale:</b>	
<b>Use Case:</b>	<ol style="list-style-type: none"> <li>1. Active stars are required for electrical isolation of Branches.</li> <li>2. Device drivers should read out status information to support diagnostics and selective control (and possibly enabling and disabling) of transceivers.</li> </ol>
<b>Dependencies:</b>	FlexRay Transceiver, FlexRay Interface, FlexRay State Manager.
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00272



## 5.1.6.4 Shutdown Operation

### 5.1.6.4.1 [BSW05150] Safe System Shutdown for FlexRay Transceiver Driver

<b>ID:</b>	BSW05150
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver shall support the AUTOSAR ECU state manager in a way that a safe system startup and shutdown is possible.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	<p>The ECU state manager is under development in parallel to this specification, therefore this requirement is added to check during implementation and system test for correct interaction between the FlexRay Transceiver Driver and the ECU state manager.</p> <p>In general, for startup the FlexRay Transceivers must not be enabled until the power supply is available and stable to prevent errors on the bus. Also the communication hardware and driver must not be enabled until the transceiver is configured into its normal operation mode.</p> <p>For shutdown, the communication must be stopped according to the AUTOSAR NM algorithm, the FlexRay drivers must be stopped and then the transceivers may be set to standby/sleep, too. The correct sequence depends on the wake-up/sleep concept of AUTOSAR.</p>
<b>Rationale:</b>	System startup and shutdown together with ECU state manager.
<b>Use Case:</b>	--
<b>Dependencies:</b>	ECU state manager
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

## 5.1.6.5 Fault Operation

### 5.1.6.5.1 [BSW05151] FlexRay Transceiver Driver Must Check Transceiver Control

<b>ID:</b>	BSW05151
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver shall check the control communication to the transceiver and the reaction of the transceiver for correctness.
<b>Type:</b>	New
<b>Importance:</b>	Medium
<b>Description:</b>	Depending on the supported transceiver device, the driver shall check the correctness of the executed control communication and the operation mode a transceiver is in.
<b>Rationale:</b>	Diagnostics and trouble shooting
<b>Use Case:</b>	<ol style="list-style-type: none"> <li>1. Detection of defect or misbehaving transceiver hardware</li> <li>2. Detection of corrupted SPI communication</li> </ol> <p>The check shall only be applied to errors within the transceiver or the transceiver control communication (ports or SPI), i.e. errors caused by malfunction of the <math>\mu</math>C, SW or a defect transceiver device. "Errors" caused by the "outer world" (e.g. disturbed communication</p>

	Channels or ground offsets) are not in the scope of this API.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.1.6.5.2 [BSW05168] Indicate FlexRay Transceiver Error State (modify according to Monitoring Concept and Concept Reliability)

<b>ID:</b>	BSW05168
<b>initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	Indicate the error state of a FlexRay Transceiver device to the DEM.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transceiver Driver shall indicate error states of a FlexRay Transceiver device to the DEM.
<b>Rationale:</b>	The Transceiver error state is necessary for error handling.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	<a href="#">[FR_EPL_SPEC]</a>
<b>Contributes to:</b>	--

#### 5.1.6.5.3 [BSW005212] Detect Error Information in Bus Driver

<b>ID:</b>	BSW005212
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Detect Error Information in Bus Driver
<b>Importance:</b>	<b>High</b>   medium   low
<b>Description:</b>	The FlexRay Transceiver Driver shall provide an API that detects errors in bus driver and notify them to application level.
<b>Rationale:</b>	The FlexRay modules should provide information that only the modules can detect.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	<a href="#">[BSW05175]</a> Provide Error Information
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00264 BRF00304

#### 5.1.6.5.4 [BSW005213] Hardware Checking Function on Bus Driver

<b>ID:</b>	BSW005213
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Hardware Checking Function on Bus Driver
<b>Importance:</b>	<b>high</b>   medium   low
<b>Description:</b>	The FlexRay Transceiver Driver's initialization function shall check error status in BD to ensure the hardware is working properly.
<b>Rationale:</b>	This functionality ensures that the hardware is working as expected.
<b>Use Case:</b>	Improvement of hardware reliability.
<b>Dependencies:</b>	<a href="#">[BSW05212]</a> Detect Error Information in Bus Driver
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00306

### 5.1.6.5.5 [BSW005214] Hardware Resetting Function on Bus Driver

<b>ID:</b>	BSW005214
<b>Initiator:</b>	Toyota/Bosch (WP FlexRay)
<b>Date:</b>	03.10.2008
<b>Short Description:</b>	Hardware Resetting Function on Bus Driver
<b>Importance:</b>	high   medium   low
<b>Description:</b>	FlexRay Transceiver Driver shall provide a method that reinitializes BD's functionality
<b>Rationale:</b>	When trouble occurs in the hardware level, it's likely to fix the cause by resetting the hardware.
<b>Use Case:</b>	This function shall be executed when so many errors are detected in FlexRay modules.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00307

## 5.2 Non-Functional Requirements

### 5.2.1 FlexRay Interface

#### 5.2.1.1 Abstraction Requirements

##### 5.2.1.1.1 [BSW05006] Abstraction of FlexRay-Specific Features

<b>ID:</b>	BSW05006
<b>Initiator:</b>	BMW
<b>Date:</b>	23.07.2004
<b>Short Description:</b>	Abstraction from the specific features of the FlexRay communication system.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	For data reception and sending, the FlexRay Interface shall provide to the upper software layers an abstraction from the specific features of the FlexRay communication system as well as a PDU-based API.
<b>Rationale:</b>	All bus interfaces supported by the AUTOSAR BSW (e.g. CAN, FlexRay, LIN) should provide the same API semantics/signature.
<b>Use Case:</b>	Several bus interfaces (e.g. CAN, FlexRay, and LIN) on one ECU.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

#### 5.2.1.2 Resource Usage

##### 5.2.1.2.1 [BSW05009] Local Memory Space Usage

<b>ID:</b>	BSW05009
<b>Initiator:</b>	BMW
<b>Date:</b>	23.07.2004
<b>Short Description:</b>	Local Memory Space Usage
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface shall allocate the needed memory space only once for a PDU sent multiple times in the FlexRay matrix.

<b>Rationale:</b>	The use of only one local memory space for each PDU, instead of one local memory space for each occurrence of the PDU in the FlexRay matrix, implicates a much better performance.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05004]</a> PDU-Based API
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.2.1.3 Configuration

#### 5.2.1.3.1 [BSW05056] Configuration of the FlexRay Interface at System Configuration Time

<b>ID:</b>	BSW05056
<b>Initiator:</b>	BMW
<b>Date:</b>	19.10.2004
<b>Short Description:</b>	Configuration <a href="#">at system configuration time</a> .
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Interface configuration shall be done <a href="#">at system configuration time</a> , i.e. the configuration parameters have to be defined before the execution of the FlexRay Interface code (even prior to executing the initialization code).
<b>Rationale:</b>	The purpose of the FlexRay Interface is to execute data processing that has been configured <a href="#">at system configuration time</a> , i.e. before the execution of the FlexRay Interface code. It shall not decide or calculate during runtime any communication parameters like the assignment of Slots to ECU or the packaging of PDUs into FlexRay Frames.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05042]</a> Switch Configuration during Runtime
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

## 5.2.2 Transport Layer FlexRay

### 5.2.2.1 [BSW05098] Usage of ISO 15765-2 and ISO 15765-4 Specifications

<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Compliance with ISO 15765-2 and with ISO 15765-4.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	It shall be possible to configure the AUTOSAR FlexRay Transport Layer <a href="#">at system configuration time</a> to be compliant with the ISO 15765-2 and the ISO 15765-4 specification per ECU
<b>Rationale:</b>	Reuse of <b>OLD/OUTDATED</b> standards for AUTOSAR BSW for backwards compatibility reasons.
<b>Use Case:</b>	--
<b>Dependencies:</b>	Either BSW05098 or BSW05073 shall be enabled per ECU at the same time.
<b>Conflicts:</b>	
<b>Supporting Material:</b>	<a href="#">[ISO 15765-2]</a> and <a href="#">[ISO 15765-4]</a> specification.

### 5.2.2.2 [BSW05073] Usage of ISO 10681-2 Specifications

<b>ID:</b>	BSW05073
<b>Initiator:</b>	DAI
<b>Date:</b>	20. 06. 2008
<b>Short Description:</b>	Support ISO 10681-2
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	It shall be possible to configure the AUTOSAR FlexRay Transport Layer <a href="#">at system configuration time</a> to be compliant with the ISO 10681-2 specification per ECU
<b>Rationale:</b>	Usage of an ISO standard for Flexray
<b>Use Case:</b>	--
<b>Dependencies:</b>	Either BSW05073 or BSW05098 shall be enabled at the same time.
<b>Conflicts:</b>	
<b>Supporting Material:</b>	ISO 10681-2.
<b>Contributes to:</b>	BRF00252

### 5.2.2.3 [BSW05074] FlexRay Transport Layer Interfaces

<b>ID:</b>	BSW05074
<b>Initiator:</b>	BMW
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Location of the FlexRay Transport Layer software module.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transport Layer software module shall be located between the PDU Router and the FlexRay Interface for FlexRay PDUs requiring Transport Protocol functionalities. The FlexRay Transport Layer is used by the PDU Router to transmit and receive FlexRay PDUs coming from the Diagnostic Communication Manager (or from a SW-Component).
<b>Rationale:</b>	Design of AUTOSAR Software Architecture.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	AUTOSAR Layered Software Architecture <a href="#">[LSA]</a>
<b>Contributes to:</b>	--

### 5.2.2.4 [BSW05075] Independence of the Network Configuration

<b>ID:</b>	BSW05075
<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Independence of the Network Configuration.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transport Layer implementation shall be independent of the network configuration represented by the FlexRay communication matrix. The API just deals with universal identifiers and data units (N-SDU) properties.
<b>Rationale:</b>	Design of AUTOSAR Software Architecture.
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--

<b>Supporting Material:</b>	AUTOSAR Layered Software Architecture <a href="#">[LSA]</a>
<b>Contributes to:</b>	--

### 5.2.2.5 [BSW05123] Configuration Modifiable by a Flashing Process

<b>ID:</b>	BSW05123
<b>Initiator:</b>	BMW
<b>Date:</b>	15.02.2005
<b>Short Description:</b>	Modification of configuration data <a href="#">by a flashing process</a> .
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	All configuration data of the FlexRay Transport Layer defined <a href="#">at system configuration time</a> shall be modifiable <a href="#">by a flashing process</a> .  This encompasses (amongst other configuration items) all connection-specific parameters and also all global parameters required for the configuration of the FlexRay Transport Layer.
<b>Rationale:</b>	Change of FlexRay Transport Layer parameters without recompiling of the FlexRay Transport Layer software.
<b>Use Case:</b>	--
<b>Dependencies:</b>	<a href="#">[BSW05034]</a> Configuration Modifiable by a Flashing Process
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.2.2.6 [BSW05076] Multiple Logical FlexRay Transport Layer Channels

<b>ID:</b>	BSW05076
<b>Initiator:</b>	VW / AUDI
<b>Date:</b>	20.01.2005
<b>Short Description:</b>	Support of at least 32 logical FlexRay Transport Layer Channels being used concurrently.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	The FlexRay Transport Layer shall support at least 32 logical FlexRay Transport Layer Channels being used concurrently. The number of Channels shall be configurable <a href="#">at system configuration time</a> .
<b>Rationale:</b>	Enable multiple concurrent FlexRay Transport Layer connections.
<b>Use Case:</b>	The FlexRay Transport Layer configuration may be specific for each Transport Layer connection.
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

## 5.2.3 FlexRay Transceiver Driver

The FlexRay Transceiver Driver has strong relations to the AUTOSAR NM and to the AUTOSAR ECU state manager since they control and use this BSW component. The requirements and needs of these two BSW components may have strong influence on this specification and maybe vice versa.

The FlexRay Transceiver Driver itself will use the drivers for ports (DIO) or SPI. Therefore, the FlexRay Transceiver Driver will be a user for their services as it is. If

the FlexRay Transceiver Driver uses other drivers for e.g. initialization access, they must be available before the FlexRay Transceiver initialization is executed!

### 5.2.3.1.1 Timing Requirements

#### 5.2.3.1.1.1 [BSW05152] Transceiver-specific Timing Requirements

<b>ID:</b>	BSW05152
<b>Initiator:</b>	Bosch
<b>Date:</b>	30.08.2005
<b>Short Description:</b>	The FlexRay Transceiver Driver shall handle the transceiver-specific timing requirements internally.
<b>Type:</b>	New
<b>Importance:</b>	High
<b>Description:</b>	<p>The communication between the <math>\mu</math>C and the transceiver is performed via ports or SPI or both. If ports are used, applying values in a predefined sequence and with a given timing to the ports are used to communicate and change the hardware operation modes. These sequences and timings must be handled within the FlexRay Transceiver Driver.</p> <p>Small times may be implemented as a wait loop inside the driver. Disadvantages are that this time is lost for the other software and the wait time depends on the used <math>\mu</math>C and e.g. system clock. Large wait times (e.g. <math>&gt;200\mu</math>s) may require an asynchronous API of the FlexRay Transceiver Driver. Disadvantage is then that the complete API and usage will be different for such a hardware device.</p>
<b>Rationale:</b>	Correct handling of used transceiver
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	--

### 5.3 [BSW999999999]Not Applicable Requirements

<b>ID:</b>	BSW999999999
<b>Initiator:</b>	--
<b>Date:</b>	--
<b>Short Description:</b>	This Requirement contributes to all requirements which are not applicable for this SWS.
<b>Type:</b>	--
<b>Importance:</b>	--
<b>Description:</b>	--
<b>Rationale:</b>	--
<b>Use Case:</b>	--
<b>Dependencies:</b>	--
<b>Conflicts:</b>	--
<b>Supporting Material:</b>	--
<b>Contributes to:</b>	BRF00270

## 6 References

### 6.1 Deliverables of AUTOSAR

**[GLOSSARY]** Glossary,  
AUTOSAR\_TR\_Glossary.pdf

**[LSA]** Layered Software Architecture,  
AUTOSAR\_EXP\_LayeredSoftwareArchitecture.pdf

**[VFB]** Virtual Function Bus,  
AUTOSAR\_EXP\_VFB.pdf

### 6.2 Related Standards and Norms

#### 6.2.1 FlexRay

**[FR\_PROT\_SPEC]** FlexRay Communications System Protocol Specification Version 2.1 Revision A,  
<http://www.FlexRay.com>

**[FR\_EPL\_SPEC]** FlexRay Communications System Electrical Physical Layer Specification Version 2.1 Revision A,  
<http://www.FlexRay.com>

#### 6.2.2 ISO

**[ISO\_10681-2]** ISO/DIS 10681-2 , Road vehicles – Communication on FlexRay – Part 2: Communication Layer services