

BS ISO 17987-6:2016



BSI Standards Publication

# Road vehicles — Local Interconnect Network (LIN)

Part 6: Protocol conformance test  
specification

**National foreword**

This British Standard is the UK implementation of ISO 17987-6:2016.

The UK participation in its preparation was entrusted to Technical Committee AUE/16, Data Communication (Road Vehicles).

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2016.  
Published by BSI Standards Limited 2016

ISBN 978 0 580 81733 5

ICS 43.040.15

**Compliance with a British Standard cannot confer immunity from legal obligations.**

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 October 2016.

**Amendments/corrigenda issued since publication**

Date	Text affected
------	---------------

---

INTERNATIONAL  
STANDARD

BS ISO 17987-6:2016

**ISO**  
**17987-6**

First edition  
2016-10-01

---

---

**Road vehicles — Local Interconnect  
Network (LIN) —**

Part 6:  
**Protocol conformance test  
specification**

*Véhicules routiers — Réseau Internet local (LIN) —*

*Partie 6: Spécification du protocole d'essai de conformité*



Reference number  
ISO 17987-6:2016(E)

© ISO 2016



**COPYRIGHT PROTECTED DOCUMENT**

© ISO 2016, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office  
Ch. de Blandonnet 8 • CP 401  
CH-1214 Vernier, Geneva, Switzerland  
Tel. +41 22 749 01 11  
Fax +41 22 749 09 47  
copyright@iso.org  
www.iso.org

# Contents

Page

<b>Foreword</b> .....	<b>vii</b>
<b>Introduction</b> .....	<b>viii</b>
<b>1 Scope</b> .....	<b>1</b>
<b>2 Normative references</b> .....	<b>1</b>
<b>3 Terms, definitions, symbols and abbreviated terms</b> .....	<b>1</b>
3.1 Terms and definitions.....	1
3.2 Symbols.....	2
3.3 Abbreviated terms.....	3
<b>4 Conventions</b> .....	<b>4</b>
<b>5 General test specification considerations</b> .....	<b>4</b>
5.1 General.....	4
5.2 Test conditions.....	4
5.3 Mandatory requirements for IUT as master.....	4
5.4 Mandatory requirements for IUT as slave.....	5
5.5 Test case architecture.....	5
5.6 Classification.....	6
5.7 Test system requirements.....	6
5.7.1 Generation of LIN frames.....	6
5.7.2 Standard requirements for the test cases.....	6
5.7.3 Special requirements for bit timing testing.....	6
5.7.4 Test system for IUT as slave node.....	6
5.7.5 Sleep state verification for IUT as slave node.....	6
5.8 Test system definition.....	7
5.9 Global predefinitions for the test setup.....	7
5.9.1 Configuration of IUT and test system.....	7
5.9.2 Default delays for frame headers.....	9
5.9.3 Default bit rate.....	9
5.9.4 Time measurement.....	9
5.9.5 Default spaces between the different frame parts of a LIN message.....	9
<b>6 Essential test cases before test start</b> .....	<b>10</b>
6.1 General.....	10
6.2 [PT-CT 1] Diagnostic frame “master request”, IUT as slave.....	10
6.3 [PT-CT 2] Diagnostic frame “slave response”, IUT as slave.....	10
6.4 [PT-CT 3] Error in received frame, IUT as slave.....	10
<b>7 Timing parameters</b> .....	<b>11</b>
7.1 General.....	11
7.2 [PT-CT 4] Length of break field low phase, IUT as master.....	11
7.3 [PT-CT 5] Variation of length of break field low phase, IUT as slave.....	11
7.4 [PT-CT 6] Length of break delimiter, IUT as master.....	12
7.5 [PT-CT 7] Variation of length of break delimiter, IUT as slave.....	12
7.6 [PT-CT 8] Inconsistent break field error, IUT as slave.....	13
7.7 [PT-CT 9] Inconsistent sync byte field error, IUT as slave.....	13
7.8 [PT-CT 10] Verification of the sync byte field, IUT as master.....	14
7.9 [PT-CT 11] Incomplete frame reception, IUT as slave.....	14
7.10 [PT-CT 12] Unknown frame reception, IUT as slave.....	15
7.11 [PT-CT 13] Length of header, IUT as master.....	16
7.12 [PT-CT 14] Variation of length of header, IUT as slave.....	16
7.13 [PT-CT 15] Bit rate tolerance, IUT as master.....	17
7.14 [PT-CT 16] Bit rate tolerance, IUT as slave without making use of synchronization.....	17
7.15 [PT-CT 17] Bit rate tolerance, IUT as slave with making use of synchronization.....	18
7.16 Length of response.....	18
7.16.1 [PT-CT 18] Length of response, IUT as slave.....	18

7.16.2	[PT-CT 19] Length of response, IUT as master.....	19
7.16.3	[PT-CT 20] Acceptance of response field, IUT as slave.....	20
7.17	Verification of schedule table timing.....	21
7.17.1	[PT-CT 21] Verification of jitter, IUT as master.....	21
7.17.2	[PT-CT 22] Schedule table management, IUT as master.....	22
7.18	[PT-CT 23] Sample point test, IUT as slave.....	22
7.19	[PT-CT 24] Initialization time, IUT as slave.....	23
<b>8</b>	<b>Communication without failure.....</b>	<b>24</b>
8.1	Variation of LIN identifier.....	24
8.1.1	[PT-CT 25] Variation of LIN PID, IUT as master.....	24
8.1.2	[PT-CT 26] Variation of LIN PIDs of subscribed frames, IUT as slave.....	24
8.1.3	[PT-CT 27] Variation of LIN identifier of published frames, IUT as slave.....	25
8.2	Transmission of the checksum byte.....	25
8.2.1	[PT-CT 28] Transmission of the checksum byte “classic checksum”, IUT as slave.....	25
8.2.2	[PT-CT 29] Transmission of the checksum byte “enhanced checksum”, IUT as slave.....	26
8.2.3	[PT-CT 30] Transmission of the checksum byte “classic checksum”, IUT as master.....	26
8.2.4	[PT-CT 31] Transmission of the checksum byte of unconditional frames, IUT as master.....	26
8.3	Unused bits.....	27
8.3.1	[PT-CT 32] Unused bits, IUT as master.....	27
8.3.2	[PT-CT 33] Unused bits, IUT as slave.....	27
8.4	Reserved frame.....	28
8.4.1	[PT-CT 34] Reserved frame, IUT as slave.....	28
8.5	[PT-CT 35] Diagnostic frame master request, IUT as master.....	28
8.6	Supported frames according to the IUT specification.....	29
8.6.1	[PT-CT 36] Supported Tx frames according to the IUT specification, IUT as slave.....	29
8.6.2	[PT-CT 37] Supported Rx frames according to the IUT specification, IUT as slave.....	29
<b>9</b>	<b>Communication with failure.....</b>	<b>30</b>
9.1	General.....	30
9.2	[PT-CT 38] Bit error, IUT as slave.....	30
9.3	[PT-CT 39] Framing error in header of published frame, IUT as slave.....	31
9.4	[PT-CT 40] Framing error in response field of subscribed frame, IUT as slave.....	32
9.5	[PT-CT 41] Checksum error by inversion, IUT as slave.....	32
9.6	[PT-CT 42] Checksum error by carry, IUT as slave.....	32
9.7	[PT-CT 43] Communication robustness, IUT as slave.....	33
<b>10</b>	<b>Event triggered frames.....</b>	<b>33</b>
10.1	General.....	33
10.2	[PT-CT 44] Event triggered frame, IUT as slave.....	33
10.3	Event triggered frame with collision.....	34
10.3.1	[PT-CT 45] Event triggered frame with collision resolving, IUT as slave.....	34
10.3.2	[PT-CT 46] Event triggered frame with errors in collision resolving, IUT as slave.....	34
10.3.3	[PT-CT 47] Event triggered frame with collision resolving, IUT as master.....	35
10.3.4	[PT-CT 48] Error in transmitted frame with collision, IUT as slave.....	35
<b>11</b>	<b>Status management.....</b>	<b>36</b>
11.1	[PT-CT 49] Error in received frame, IUT as slave.....	36
11.2	[PT-CT 50] Error in transmitted frame, IUT as slave.....	36
11.3	[PT-CT 51] response_error signal handling, IUT as slave.....	37
<b>12</b>	<b>Sleep/wake up/power mode tests.....</b>	<b>37</b>
12.1	[PT-CT 52] Send “go-to-sleep command”, IUT as master.....	37
12.2	[PT-CT 53] Receive “go-to-sleep command”, IUT as slave.....	38
12.3	[PT-CT 54] Receive a wake up signal, IUT as master.....	39

12.4	[PT-CT 55] Receive a wake up signal, IUT as slave.....	39
12.5	Send a wake up signal.....	40
12.5.1	[PT-CT 56] Send a wake up signal, IUT as master and IUT as slave.....	40
12.5.2	[PT-CT 57] Send a block of wake up signals, IUT as slave.....	40
12.5.3	[PT-CT 58] Wait after one block of wake up signals, IUT as slave.....	41
12.5.4	[PT-CT 59] Send a wake up signal, frame header from a master following, IUT as slave.....	41
12.6	[PT-CT 60] ECU power loss, IUT as master.....	42
12.7	[PT-CT 61] Powered up with LIN shorted, IUT as master.....	43
12.8	[PT-CT 62] LIN shorted before scheduling, IUT as master.....	43
12.9	[PT-CT 63] LIN shorted after start of scheduling, IUT as master.....	44
<b>13</b>	<b>Sleep state after bus idle.....</b>	<b>44</b>
13.1	[PT-CT 64] Sleep state after event and bus idle, IUT as slave.....	44
13.2	[PT-CT 65] Sleep state after bus idle with power up and wake up signal, IUT as slave.....	45
13.3	[PT-CT 66] Timeout after bus idle, IUT as slave.....	46
<b>14</b>	<b>Frame ID range assignment.....</b>	<b>46</b>
14.1	[PT-CT 67] Frame ID range assignment with indirect response, IUT as slave.....	46
14.2	[PT-CT 68] Frame ID range unassignment with indirect response, IUT as slave.....	47
<b>15</b>	<b>Wildcards.....</b>	<b>48</b>
15.1	[PT-CT 69] Request with direct response, IUT as slave.....	48
<b>16</b>	<b>ReadByIdentifier command.....</b>	<b>48</b>
16.1	LIN product identification.....	48
16.1.1	[PT-CT 70] LIN product identification request with direct response, IUT as slave.....	48
16.1.2	[PT-CT 71] LIN product identification — With interleaved unconditional frame, IUT as slave.....	49
16.2	[PT-CT 72] ReadByIdentifier command with correct NAD, IUT as slave.....	49
16.3	[PT-CT 73] ReadByIdentifier command with incorrect addressing, IUT as slave.....	50
<b>17</b>	<b>NAD assignment.....</b>	<b>51</b>
17.1	General.....	51
17.2	[PT-CT 74] NAD assignment — Followed by ReadByIdentifier service, IUT as slave.....	51
17.3	[PT-CT 75] NAD assignment — With positive response, IUT as slave.....	51
17.4	[PT-CT 76] NAD assignment — Initial NAD, IUT as slave.....	51
<b>18</b>	<b>Save Configuration.....</b>	<b>52</b>
18.1	General.....	52
18.2	[PT-CT 77] Save Configuration — With positive response, IUT as slave.....	52
18.3	[PT-CT 78] Save Configuration — Save a new NAD, IUT as slave.....	52
18.4	[PT-CT 79] Save Configuration — Save new frame identifiers, IUT as slave.....	53
<b>19</b>	<b>Transport protocol.....</b>	<b>54</b>
19.1	[PT-CT 80] Transport layer functional request, IUT as slave.....	54
19.2	[PT-CT 81] Abort diagnostic communication with new diagnostic request, IUT as slave.....	54
19.3	[PT-CT 82] IUT receives a segmented request as specified, IUT as slave.....	54
19.4	[PT-CT 83] IUT receives a segmented request interleaved with unconditional frame, IUT as slave.....	55
19.5	[PT-CT 84] IUT receives a segmented request with interleaved functional request, IUT as slave.....	56
19.6	IUT shall ignore request after timeout.....	57
19.6.1	[PT-CT 85] IUT shall ignore segmented requests on N_Cr <sub>Max</sub> timeout, IUT as slave.....	57
19.6.2	[PT-CT 86] IUT shall observe transport layer N_As <sub>Max</sub> timeout, IUT as slave.....	58
19.7	[PT-CT 87] IUT shall ignore segmented requests with wrong sequence numbering, IUT as slave.....	59
19.8	[PT-CT 88] IUT shall respond with correct segmented response, IUT as slave.....	60
19.9	IUT sends a segmented response with interleaved unconditional frames.....	61

19.9.1	[PT-CT 89] IUT sends a segmented response with interleaved unconditional frame, IUT as slave.....	61
19.9.2	[PT-CT 90] IUT sends a segmented response with interleaved functional request, IUT as slave.....	62
19.10	[PT-CT 91] IUT shall not respond to slave response header if there is no request before, IUT as slave.....	63
19.11	[PT-CT 92] IUT shall not respond to slave response header if the response is already sent, IUT as slave.....	63
19.12	[PT-CT 93] IUT shall abort segmented response on N_CsMax timeout, IUT as slave.....	64
<b>Bibliography</b> .....		<b>66</b>



## Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see [www.iso.org/directives](http://www.iso.org/directives)).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see [www.iso.org/patents](http://www.iso.org/patents)).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: [www.iso.org/iso/foreword.html](http://www.iso.org/iso/foreword.html).

The committee responsible for this document is ISO/TC 22, *Road vehicles*, Subcommittee SC 31, *Data communication*.

A list of all parts in the ISO 17987 series can be found on the ISO website.

## Introduction

ISO 17987 (all parts) specifies the use cases, the communication protocol and physical layer requirements of an in-vehicle communication network called Local Interconnect Network (LIN).

The LIN protocol as proposed is an automotive focused low speed universal asynchronous receiver transmitter (UART) based network. Some of the key characteristics of the LIN protocol are signal based communication, schedule table based frame transfer, master/slave communication with error detection, node configuration and diagnostic service transportation.

The LIN protocol is for low cost automotive control applications, for example, door module and air condition systems. It serves as a communication infrastructure for low-speed control applications in vehicles by providing:

- signal based communication to exchange information between applications in different nodes;
- bit rate support from 1 kbit/s to 20 kbit/s;
- deterministic schedule table based frame communication;
- network management that wakes up and puts the LIN cluster into sleep state in a controlled manner;
- status management that provides error handling and error signalling;
- transport layer that allows large amount of data to be transported (such as diagnostic services);
- specification of how to handle diagnostic services;
- electrical physical layer specifications;
- node description language describing properties of slave nodes;
- network description file describing behaviour of communication;
- application programmer's interface.

ISO 17987 (all parts) is based on the open systems interconnection (OSI) basic reference model as specified in ISO/IEC 7498-1 which structures communication systems into seven layers.

The OSI model structures data communication into seven layers called (top down) *application layer* (layer 7), *presentation layer*, *session layer*, *transport layer*, *network layer*, *data link layer* and *physical layer* (layer 1). A subset of these layers is used in ISO 17987 (all parts).

ISO 17987 (all parts) distinguishes between the services provided by a layer to the layer above it and the protocol used by the layer to send a message between the peer entities of that layer. The reason for this distinction is to make the services, especially the application layer services and the transport layer services, reusable also for other types of networks than LIN. In this way, the protocol is hidden from the service user and it is possible to change the protocol if special system requirements demand it.

ISO 17987 (all parts) provides all documents and references required to support the implementation of the requirements related to the following.

- ISO 17987-1: This part provides an overview of ISO 17987 (all parts) and structure along with the use case definitions and a common set of resources (definitions, references) for use by all subsequent parts.
- ISO 17987-2: This part specifies the requirements related to the transport protocol and the network layer requirements to transport the PDU of a message between LIN nodes.
- ISO 17987-3: This part specifies the requirements for implementations of the LIN protocol on the logical level of abstraction. Hardware related properties are hidden in the defined constraints.

- ISO 17987-4: This part specifies the requirements for implementations of active hardware components which are necessary to interconnect the protocol implementation.
- ISO 17987-5: This part specifies the LIN application programmers interface (API) and the node configuration and identification services. The node configuration and identification services are specified in the API and define how a slave node is configured and how a slave node uses the identification service.
- ISO 17987-6: This part specifies tests to check the conformance of the LIN protocol implementation according to ISO 17987-2 and ISO 17987-3. This comprises tests for the data link layer, the network layer and the transport layer.
- ISO 17987-7: This part specifies tests to check the conformance of the LIN electrical physical layer implementation (logical level of abstraction) according to ISO 17987-4.



# Road vehicles — Local Interconnect Network (LIN) —

## Part 6: Protocol conformance test specification

### 1 Scope

This document specifies the LIN protocol conformance test. This test verifies the conformance of LIN communication controllers with respect to ISO 17987-2 and ISO 17987-3.

This document provides all necessary technical information to ensure that test results are identical even on different test systems, provided that the particular test suite and the test system are compliant to the content of this document.

### 2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 17987-2:2016, *Road vehicles — Local Interconnect Network (LIN) — Part 2: Transport protocol and network layer services*

ISO 17987-3:2016, *Road vehicles — Local Interconnect Network (LIN) — Part 3: Protocol specification*

ISO 17987-4:2016, *Road vehicles — Local Interconnect Network (LIN) — Part 4: Electrical Physical Layer (EPL) specification 12V/24V*

### 3 Terms, definitions, symbols and abbreviated terms

#### 3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

##### 3.1.1

##### **class B device**

µC-based LIN device

Note 1 to entry: These are devices where it is possible to take measurements on the Rx and Tx interface circuits between the µC and the transceiver.

##### 3.1.2

##### **class C device**

integrated LIN devices (ECU) with µC and transceiver

Note 1 to entry: These are devices where it is not possible to take measurements on the Rx and Tx interface circuits between the µC and the transceiver.

### 3.2 Symbols

$F_{TOL\_RES\_MASTER}$	bit rate tolerance of the master node (absolute value), according to ISO 17987-3	%
$F_{TOL\_RES\_SLAVE}$	bit rate tolerance of a slave node without making use of synchronization (absolute value), according to ISO 17987-3	%
$F_{TOL\_SYNC}$	bit rate tolerance of a slave node making use of synchronization (relative value to master node after synchronization, valid for the complete message), according to ISO 17987-3	%
$F_{TOL\_UNSYNC}$	bit rate tolerance of a slave node making use of synchronization, according to ISO 17987-3	%
$T_{AWAKE}$	measured time between end of wake up signal and start of break of a header	s
$T_{BIT}$	Length of a bit (time), depending on the bit rate	s
$T_{BIT\_MAX\_MASTER}$	$T_{BIT\_MAX\_MASTER} = T_{BIT} (1 - F_{TOL\_RES\_MASTER})$	s
$T_{BIT\_MIN\_MASTER}$	$T_{BIT\_MIN\_MASTER} = T_{BIT} (1 + F_{TOL\_RES\_MASTER})$	s
$T_{BIT\_NOM\_MASTER}$	$T_{BIT\_NOM\_MASTER} = T_{BIT}$	s
$T_{BRKDEL}$	break delimiter, according to ISO 17987-3	1 – 14,6 $T_{BIT}$
$T_{BRKDEL\_MAX}$	calculated maximum of break delimiter: $T_{HEADER\_MAX} - (T_{BRKFLD\_MIN} + 20 T_{BIT})$	14,6 $T_{BIT}$
$T_{BRKDEL\_MIN}$	minimum of break delimiter, according to ISO 17987-3	1 $T_{BIT}$
$T_{BRKFLD}$	break field low phase, according to ISO 17987-3	13 – 26,6 $T_{BIT}$
$T_{BRKFLD\_MAX}$	calculated maximum of break field low phase: $T_{HEADER\_MAX} - (T_{BRKDEL\_MIN} + 20 T_{BIT})$	26,6 $T_{BIT}$
$T_{BRKFLD\_MIN}$	minimum of break field low phase, according to ISO 17987-3	13 $T_{BIT}$
$T_{FRAME}$	length of a 8 byte frame, according to ISO 17987-3 (see frame length) $T_{FRAME} = T_{HEADER} + T_{RESPONSE}$	124 – 173,6 $T_{BIT}$
$T_{FRAME\_MAX}$	maximum length of a 8 byte frame, according to ISO 17987-3	173,6 $T_{BIT}$
$T_{FRAME\_MIN}$	minimum length of a 8 byte frame, according to ISO 17987-3	124 $T_{BIT}$
$T_{FRAME\_SLOT\_MEASURE}$	shall be measured between falling edges of the break field	s
$T_{FRAME\_SLOT\_SPECIFIED}$	the length is specified in the LDF	s

T <sub>H_INTERBYTE</sub>	inter-byte space between sync byte field and protected identifier	0 – 13,6 T <sub>BIT</sub>
T <sub>HEADER</sub>	length of the header of a message frame based on T <sub>BIT</sub> nominal	34 – 47,6 T <sub>BIT</sub>
T <sub>HEADER_MAX</sub>	maximum length of the header of a message frame, according to ISO 17987-3	47,6 T <sub>BIT</sub>
T <sub>HEADER_MIN</sub>	minimum length of the header of a message frame, according to ISO 17987-3	34 T <sub>BIT</sub>
T <sub>JITTER_DEFINED</sub>	jitter according LDF or NCF of the IUT	s
T <sub>JITTER_MEASURE</sub>	measured jitter as described in ISO 17987-3 (see frame slot)	s
T <sub>RESPONSE_MAX</sub>	maximum response length	126 T <sub>BIT</sub>
T <sub>RESPONSE_MIN</sub>	nominal response length	90 T <sub>BIT</sub>
T <sub>SLEEP</sub>	measured time after that a slave node enters automatically a sleep state ISO 17987-2:2016, 5.1.4	s

### 3.3 Abbreviated terms

AC	alternate current
API	application programming interface
BFS	byte field synchronization
CF	transport layer consecutive frame
DC	direct current
EBS	earliest bit sample
EMC	electromagnetic compatibility
EMI	electromagnetic interference
EPL	electrical physical layer
ESD	electrostatic discharge
FF	transport layer first frame
GND	ground
IUT	implementation under test
LBS	latest bit sample
Max	maximum
Min	minimum
NVM	non-volatile memory

no.	number
OSI	open systems interconnection
PID	protected identifier
PDU	protocol data unit
PT-CT	LIN data link layer, network layer and transport layer protocol conformance test
RSID	response service identifier
Rx	Rx pin of the transceiver
RXD	receive data
SF	transport layer single frame
SID	service identifier
SR	sample window repetition
TC	test case
TRX	transceiver
Tx	Tx pin of the transceiver
TXD	transmit data
Typ	typical
UART	universal asynchronous receiver transmitter

## 4 Conventions

ISO 17987 and ISO 14229-7 are based on the conventions specified in the OSI service conventions (ISO/IEC 10731) as they apply for physical layer, protocol, network and transport protocol and diagnostic services.

## 5 General test specification considerations

### 5.1 General

This test specification is not able to cover all contingencies. Due to the fact of the missing vehicle environment, it is possible that the IUT's behaviour differs.

### 5.2 Test conditions

The tests shall be done at temperature in the range of 15 °C to 35 °C.

### 5.3 Mandatory requirements for IUT as master

The LDF is mandatory to perform the LIN conformance tests for IUT as master.

If the LDF is not able to describe all features of the IUT, an additional device specific datasheet is necessary (for example, used diagnostic services).



Depending on the implementation of the IUT as master, it is allowed to use all possible master request frames (e.g. instead of TST\_FRM\_ASSIGNIDRANGE) for testing, except mandatory supported frames.

IUT initialization is required before each test case. Deviations are marked in the test case respectively.

#### 5.4 Mandatory requirements for IUT as slave

The NCF or alternatively the LDF is mandatory to perform the LIN conformance tests for IUT as slave.

The used test tool shall verify the syntax of the NCF/LDF for plausibility (not for the content).

The NCF/LDF shall match with the implementation of the device.

If the NCF/LDF is not able to describe all features of the IUT, an additional device specific datasheet is necessary (for example, used diagnostic services).

If an IUT is not fully configured after reset, an IUT initialization is required before each test case, except if the AssignFrameIdentifierRange command is part of the test. Preconfigured slaves are fully configured after reset. Deviations are marked in the test case respectively.

#### 5.5 Test case architecture

In the description of each test case, it is specified for which device type the test case is applicable, for master or slave.

Each specification of a test case consists of five parts:

- Set Up
  - defines the IUT as master or slave;
  - defines settings for the implementation under test (IUT) and the test system (for details, see [5.9.1](#));
  - defines the bit rate for the respective test case;
- System Init
  - defines to what state the IUT shall have been set before starting the execution of the test. If not otherwise defined, the IUT as master sends requests respective the IUT as slave waits for requests;
  - an initialization of the IUT shall be performed before each test case. To initialize the IUT, a reset is carried out and thereafter, the IUT shall be reconfigured, e.g. by a Frame ID configuration process;
- Test
  - defines the way of stimulating the IUT;
  - if more than one step is defined in this field, the steps shall be executed in the order as they are stated in the document;
- Verification
  - defines the expected behaviour of the IUT when executing the test steps;
- Reference
  - defines the reference to this document or other parts of the ISO 17987 series.

## 5.6 Classification

The classification describes the LIN integration level.

[Table 1](#) defines the classification.

**Table 1 — Classification**

No.	Classification	Description	Comment
1	Class A device	Transceiver devices	Data Link Layer and Node Configuration/ Network Management Tests not applicable
2	Class B device	µC based devices	IUT as master or slave with Rx and Tx pin connectors
3	Class C device	integrated devices (ECU) with µC and transceiver	IUT as master or slave with analog LIN bus connector available

## 5.7 Test system requirements

### 5.7.1 Generation of LIN frames

The test system shall ensure the precision of the bit time of a master node defined in ISO 17987-3.

### 5.7.2 Standard requirements for the test cases

For proper measurement and verification of LIN frames, the test system shall use a minimum over sampling factor of 16, see [Formula \(1\)](#):

$$\text{sample resolution} \leq \frac{T_{\text{BIT}}}{16} \quad (1)$$

### 5.7.3 Special requirements for bit timing testing

For proper measurement and verification of the bit timing tests, the test system shall measure with a minimum over sampling of 10 to the precision of the master tolerance given by the ISO 17987-4 [see [Formula \(2\)](#)]. See [7.13](#), [7.14](#), [7.15](#).

$$\text{bit time sample resolution} \leq \frac{T_{\text{BIT}} \frac{0,5}{100}}{10} \quad (2)$$

### 5.7.4 Test system for IUT as slave node

If the IUT is a slave node the test system acts as a master node. The transmission of a frame by the test system is a synonym for the transmission of the frame header if the specific frame is published by the IUT according to the frame definition.

EXAMPLE “The test system sends TST\_FRM\_IUT\_TX” is corresponding to “The test system sends the header of TST\_FRM\_IUT\_TX”.

### 5.7.5 Sleep state verification for IUT as slave node

Some test cases require verification of the slave node bus sleep state after transmission of a go-to-sleep command or bus idle timeout. As a direct verification is not possible due to the design of the test system, an indirect method is needed. Depending on the IUT specification, one of the following verification alternatives shall be selected for the sleep state verification procedure.

- Observing of the current consumption of the IUT. It shall be verified that the current consumption is reduced after sleep state criteria is fulfilled and the ECU specific delay is passed. The ECU specific

delay between entering sleep state and reduction of ECU current shall be provided by IUT vendor and considered in the test verification.

Prerequisite: ECU enters low power mode after a specified delay when the LIN sleep state is entered.

- The test system forces the sleep state transition. After that, the IUT is stimulated to send a wakeup signal in sleep state. If possible, it should also be verified that a wake up frame can't be triggered during normal communication (a slave node shall not be able to send a wakeup if it is in operational state).

Prerequisite: ECU wake up can be triggered by an external stimulus.

- The test system forces the sleep state transition. In the next step, the test system sends the header of TST\_FRM\_STATUS\_SIG multiple times without preceding wake up one or more headers are not answered by the slave node because the first SynchBreak is taken as wake up event and the slave node performs its initialization sequence. Latest after 100 ms, the slave shall answer to the headers.

Prerequisite: ECU enters a low power mode when sleep state is entered and requires more than one header to recover from sleep state (otherwise, it could not be distinguished from operational state).

- A monitor or debugger is used to verify the network management state of the IUT.

Prerequisite: Accessible port for monitor or debugger.

- An ECU test pin is used to indicate the current network management state.

Prerequisite: Free and accessible test pin.

- In Class B devices, the LIN transceiver is set into sleep state – verify LIN transceiver state by oscilloscope/logger/...

Prerequisite: LIN transceiver is set to sleep state (with a specified delay) when the slave node enters sleep state.

- A sleep counter is implemented in the slave node which is incremented anytime the sleep state is entered. The current sleep counter state can be read out by a ReadByIdentifier request. Depending on the ECU power mode, the counter is located in RAM or non-volatile memory. The test system verifies the sleep counter before and after performing the test and derives the verdict from the counter difference.

Prerequisite: The counter is implemented in the ECU and accessible via ReadByIdentifier request.

Hint: LIN sleep state and ECU power mode are not identical. ECU power mode transitions depend on ECU requirements that can be triggered by LIN sleep state transitions. This test only verifies the valid reception of a sleep state frame and entering LIN sleep state. The ECU power mode may only be used as indicator for the LIN sleep state.

## 5.8 Test system definition

A detailed test system definition is not part of this document. The test system shall be implemented in a way that the test cases are performed as specified and independent from test persons and from other test cases and of the reproducibility.

## 5.9 Global predefinitions for the test setup

### 5.9.1 Configuration of IUT and test system

Before executing the tests, some settings shall be done concerning the following:

- the protected identifier (PID) which shall be sent by a master;

- the contents of the data of a slave answer;
- the delay between master requests;
- the bit rate which shall be used for the test.

In [Table 2](#), several frames for the IUT and the test system are defined. In each test case specification, there is a row called “configuration” mostly referring to a row in the table below. The configuration described in that row shall be applied to the respective test case.

If there is no reference to [Table 2](#), the settings are described in the test case specification itself.

- If the IUT is not able to support the given PID, the table shall be adapted to the possibilities of the IUT.
- For the delay between the master requests, see [5.9.2](#).

All used frames shall be valid and according to the device specification. [Table 2](#) defines the configuration of IUT and test system.

**Table 2 — Configuration of IUT and test system**

TST_Frame	Requirements for the test frame
TST_FRM_ASSIGNIDRANGE	AssignFrameIdentifierRange service
TST_FRM_RDBI_0	ReadByIdentifier (Identifier = 0) all other parameters shall be filled with default values according to the IUT specification and according to the test case specification.
TST_FRM_RDBI_X	ReadByIdentifier (Identifier ≠ 0) shall be used with the correct NAD, supplier ID and function ID according to the IUT specification. The ID is variable and >0
TST_FRM_RDBI_0_NAD	ReadByIdentifier (Identifier = 0) with different NAD
TST_FRM_IUT_TX	device specific transmit frame (IUT is publisher)
TST_FRM_IUT_RX	device specific receive frame (IUT is subscriber)
TST_FRM_REQ_EVT	request for event triggered frame
TST_HDR_SR_3D	slave response header, Identifier = 3D <sub>16</sub>
TST_FRM_STATUS_SIG	read response_error signal
TST_HDR_RESERVE_3E	reserved frame header: Identifier = 3E <sub>16</sub>
TST_HDR_RESERVE_3F	reserved frame header: Identifier = 3F <sub>16</sub>
TST_FRM_SLEEP_CMD	go-to-sleep command
TST_FRM_REQ_FUNC_7E	diagnostic request with functional NAD = 7E <sub>16</sub>
TST_FRM_ASSIGN_NAD	assign NAD
TST_FRM_SAVE_CONFIG	save configuraton request
TST_FRM_MR_3C_SF, TST_FRM_MR_3C_FF, TST_FRM_MR_3C_CF	this is a (SF, FF, CF) master request message (3C <sub>16</sub> ) which can carry any kind of diagnostic message, except messages of configuration and identification.
TST_FRM_SR_3D_SF, TST_FRM_SR_3D_FF, TST_FRM_SR_3D_CF	this is a (SF, FF, CF) slave response message (3D <sub>16</sub> ) which can carry any kind of diagnostic message, except messages of configuration and identification.
TST_FRM_UNKNOWN	all undefined (not configured in the IUTs LDF, NCF) frame IDs from 0 <sub>10</sub> – 59 <sub>10</sub> with all combination of parity bits and all known frame IDs with possible incorrect parity bits Number of data bytes 8, data bytes 1 to 8 shall be filled with 00 <sub>16</sub> , 01 <sub>16</sub> , 02 <sub>16</sub> , 03 <sub>16</sub> , 04 <sub>16</sub> , 05 <sub>16</sub> , 06 <sub>16</sub> , 07 <sub>16</sub>

If it is specified in the “Test” part of the test case description that a TST\_FRM\_XXXXX is sent by the test system, this can mean that only the header of the frame is sent, but not the entire frame. The response transmission is expected to be sent by the IUT.

### 5.9.2 Default delays for frame headers

If not otherwise specified, the delay between the master headers is defined as shown in [Table 3](#).

**Table 3 — Default delays for frame headers**

Bit rate	Master delay
2 400 bit/s	80 ms
9 600 bit/s	20 ms
10 417 bit/s	20 ms
19 200 bit/s	10 ms

Remark:

Default delay shall be used, unless  $P2_{\min}$  or  $\text{min\_period}$  for a frame is defined in the NCF. If  $P2_{\min}$  is longer than the default value,  $P2_{\min}$  is to be used.

### 5.9.3 Default bit rate

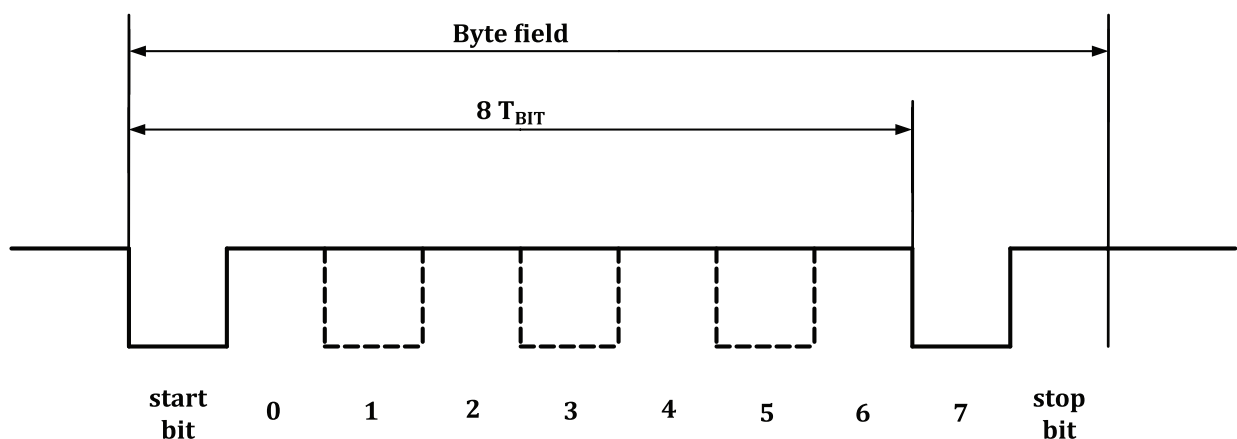
The default bit rate is set to the highest bit rate that is specified by the specification of the IUT. Test cases with different bit rate settings can only be performed when the required bit rate is in accordance with the specification of the IUT.

### 5.9.4 Time measurement

The time measurement for master nodes is done at the sync byte field as shown below. For slave nodes, the answer to ReadByIdentifier ( $ID = 2_{10}$ ) and evaluation of RSID ( $7F_{16}$ ) in the mandatory negative response is used. In this case, the distance between falling edges of “start bit” and bit 7 is measured.

The time measurement shall be done between falling edges of the pattern. The falling edge is available in a distance of 8 bit times, which allow a simple calculation of the basic bit time  $T_{\text{BIT}}$ .

[Figure 1](#) shows the byte field for time measurement.



**Figure 1 — Byte field for time measurement**

### 5.9.5 Default spaces between the different frame parts of a LIN message

If not otherwise specified in the test cases, inter byte spaces between the header bytes and data fields are set to zero.

## 6 Essential test cases before test start

### 6.1 General

These test cases shall have the result “pass”; otherwise, a further execution of the remaining test cases make no sense.

### 6.2 [PT-CT 1] Diagnostic frame “master request”, IUT as slave

This test verifies if the IUT as slave is able to receive master request frames. In this test case, the diagnostic frame type “master request” is checked.

[Table 4](#) defines the test system for diagnostic frame “master request”.

**Table 4 — Test system: Diagnostic frame “master request”**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_ASSIGNIDRANGE assigning a valid PID to TST_FRM_STATUS_SIG. The test system sends TST_FRM_STATUS_SIG.	
<b>Verification</b>	IUT shall answer	
<b>Reference</b>	ISO 17987-3:2016, 5.2.4.5 and 6.3.6.5	

### 6.3 [PT-CT 2] Diagnostic frame “slave response”, IUT as slave

This test verifies if the IUT as slave is able to answer to a slave response header. In this test case, the “slave response frame” is checked.

[Table 5](#) defines the test system for diagnostic frame “slave response”.

**Table 5 — Test system: Diagnostic frame “slave response”**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_RDBI_0. The test system sends TST_HDR_SR_3D.	
<b>Verification</b>	The IUT shall answer. The IUT shall receive the request from the test system and the response shall have 8 data bytes and use the classic checksum.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.5	

### 6.4 [PT-CT 3] Error in received frame, IUT as slave

This test verifies if the IUT as slave is able to report its status to the network.

[Table 6](#) defines the test system for “error in received frame”.

**Table 6 — Test system: Error in received frame**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_RDBI_0 with inverted checksum. After that, TST_FRM_STATUS_SIG is sent from the test system.	
<b>Verification</b>	The IUT shall answer to TST_FRM_STATUS_SIG and sends the mandatory response_error signal = TRUE.	
<b>Reference</b>	ISO 17987-3:2016, 5.5.4	

## 7 Timing parameters

### 7.1 General

This clause verifies if the timing parameters of the IUT are conforming to the timing parameters that are defined in ISO 17987-3. The IUT is stimulated to transmit or receive defined frames and the resulting behaviour is observed and compared with the expected behaviour.

### 7.2 [PT-CT 4] Length of break field low phase, IUT as master

This test verifies the correct length of the break field low phase  $T_{BRKFLD}$  as defined in ISO 17987-3.

[Table 7](#) defines the test system for “length of break field low phase”.

**Table 7 — Test system: Length of break field low phase**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT sends TST_FRM_ASSIGNIDRANGE.	
<b>Verification</b>	$T_{BRKFLD\_MIN} \leq T_{BRKFLD} \leq T_{BRKFLD\_MAX}$ . The measurement shall be performed at $0,4 V_{sup}$ of the slopes. Reference: ISO 17987-4:2016, Param 17.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.3	

### 7.3 [PT-CT 5] Variation of length of break field low phase, IUT as slave

This test verifies if the IUT as slave recognizes a header with different length of break field low phase  $T_{BRKFLD}$ .

The test is to be repeated with different lengths of break field low phase  $T_{BRKFLD}$  according to the test definition.

[Table 8](#) defines the test system for “variation of length of break field low phase”.

**Table 8 — Test system: Variation of length of break field low phase**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	a) The test system sends TST_FRM_IUT_TX with the length of break field low phase $T_{BRKFLD}$ from $0,5 T_{BIT}$ to $9,3 T_{BIT}$ , with $0,1 T_{BIT}$ steps. Every step is repeated five times. b) The test system sends TST_FRM_IUT_TX with the length of break field low phase $T_{BRKFLD}$ from 11 to $26,6 T_{BIT}$ , with $0,1 T_{BIT}$ steps. Every step is repeated 5 times.	
<b>Verification</b>	In test part a), the IUT shall not answer any of the headers. In test case b), the IUT shall answer to all of the headers in the range of break field low phase $T_{BRKFLD}$ of $13 T_{BIT}$ to $26,6 T_{BIT}$ . <sup>a</sup>	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.3	
<sup>a</sup> The answer of the IUT to the header with the length of break field of $11 T_{BIT}$ to $12,9 T_{BIT}$ shall be documented.		

#### 7.4 [PT-CT 6] Length of break delimiter, IUT as master

This test verifies the correct length of the break delimiter  $T_{BRKDEL}$ .

[Table 9](#) defines the test system for “length of break delimiter”.

**Table 9 — Test system: Length of break delimiter**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C Devices
	Configuration	TST_FRM_IUT_TX, TST_FRM_IUT_RX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT sends TST_FRM_IUT_TX or TST_FRM_IUT_RX.	
<b>Verification</b>	$T_{BRKDEL\_MIN} \leq T_{BRKDEL} \leq T_{BRKDEL\_MAX}$ . The measurement should be performed at $0,6 V_{sup}$ of the slopes. Reference: ISO 17987-4, Param 18.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.3	

#### 7.5 [PT-CT 7] Variation of length of break delimiter, IUT as slave

This test verifies if the IUT as slave recognizes a header with different length of break delimiter  $T_{BRKDEL}$ .

The test consists of three separate test cases. The length of break delimiter  $T_{BRKDEL}$  is set according to [Table 11](#).

The IUT as slave shall accept a break delimiter of 12th/16th bit length. The shortened delimiter on the LIN bus results from capacity loads on LIN.

[Table 10](#) defines the test system for “variation of length of break delimiter”.



**Table 10 — Test system: Variation of length of break delimiter**

State	Description	
Set Up	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX
	bit rate	Default
System Init	Default	
Test	The test system sends TST_FRM_IUT_TX with the length of break delimiter $T_{BRKDEL}$ as defined in <a href="#">Table 11</a> .	
Verification	The IUT shall respond to the header. NOTE 10th/16th would reflect the maximum bit sample point. Considering slave nodes which are not synchronized to the master bit rate, an additional margin of 14 % shall be added resulting in 12th/16th $T_{BRKDEL}$ .	
Reference	ISO 17987-3:2016, 5.2.2.3	

[Table 11](#) defines the test cases for “variation of length of break delimiter”.

**Table 11 — Test cases: Variation of length of break delimiter**

PT-CT-TC	$T_{BRKFLD}$	$T_{BRKDEL}$	$T_{H\_Interbyte}$	Comment
[PT-CT 7].1	$T_{BRKFLD\_MIN}$	$T_{BRKDEL\_MIN}$ class B device: $1 T_{BIT}$ class C device: $0,75 T_{BIT}$	0	minimum header length
[PT-CT 7].2	$T_{BRKFLD\_MIN}$	$T_{BRKDEL\_MAX}$	0	—
[PT-CT 7].3	$T_{BRKFLD\_MIN}$	10 Bit	0	—

## 7.6 [PT-CT 8] Inconsistent break field error, IUT as slave

This test verifies if the IUT as slave detects an inconsistent break field error. The IUT as slave shall not accept a zero byte ( $00_{16}$ ) as a break field.

[Table 12](#) defines the test system for “inconsistent break field error”.

**Table 12 — Test system: Inconsistent break field error**

State	Description	
Set Up	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
System Init	Default	
Test	The test system sends TST_FRM_RDBI_0 with 13 bit break field and TST_HDR_SR_3D. After that, the test system sends TST_FRM_RDBI_0 with 9 bit break field and TST_HDR_SR_3D.	
Verification	The IUT shall respond to the first master request. The IUT shall detect the Inconsistent break field error and shall not respond to the second master request.	
Reference	ISO 17987-3:2016, 5.2.2.3	

## 7.7 [PT-CT 9] Inconsistent sync byte field error, IUT as slave

This test verifies if the IUT as slave detects a sync byte field error, i.e. the edges of the sync byte field are not detected within the given tolerance.

Table 13 defines the test system for “inconsistent sync byte field error”.

**Table 13 — Test system: Inconsistent sync byte field error**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_STATUS_SIG with sync byte field as defined in Table 14.	
<b>Verification</b>	The IUT shall not respond to TST_FRM_STATUS_SIG.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.4	

Table 14 defines the test cases for “inconsistent sync byte field error”.

**Table 14 — Test cases: Inconsistent sync byte field error**

PT-CT-TC	IUT bit rate	Sync Byte Field
[PT-CT 9].1	Default	54 <sub>16</sub>
[PT-CT 9].2	Default	5D <sub>16</sub>

## 7.8 [PT-CT 10] Verification of the sync byte field, IUT as master

This test verifies if the IUT as master sends a sync byte field with the data “55<sub>16</sub>”.

Table 15 defines the test system for “verification of the sync byte field”.

**Table 15 — Test system: Verification of the sync byte field**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT as master sends TST_FRM_RDBI_0.	
<b>Verification</b>	The IUT shall send the sync byte field without failure.	
<b>Reference</b>	ISO 17987-4:2016, 5.2.2	

## 7.9 [PT-CT 11] Incomplete frame reception, IUT as slave

This test verifies if the IUT as slave responds to a frame correctly after a incomplete frame and if the response\_error signal is set as expected.

Table 16 defines the test system for “incomplete frame reception”.

**Table 16 — Test system: Incomplete frame reception**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_STATUS_SIG, TST_FRM_IUT_RX
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends the profile described in <a href="#">Table 17</a> followed by TST_FRM_STATUS_SIG. The minimum distance between the test profile and TST_FRM_STATUS_SIG should be the frame max. time.	
<b>Verification</b>	The IUT shall respond to TST_FRM_STATUS_SIG by and set the response_error signal as described in <a href="#">Table 17</a> .	
<b>Reference</b>	ISO 17987-3:2016, 5.5.4	

[Table 17](#) defines the test cases for “incomplete frame reception”.

**Table 17 — Test cases: Incomplete frame reception**

PT-CT-TC	Profile test	response_error signal during verification step
[PT-CT 11].1	The test system sends only the break field	FALSE
[PT-CT 11].2	The test system sends only the break field and the sync byte field	FALSE
[PT-CT 11].3	The test system sends just the header of TST_FRM_IUT_RX	FALSE
[PT-CT 11].4	The test system sends just the header of TST_FRM_IUT_RX and the first data byte	TRUE

### 7.10 [PT-CT 12] Unknown frame reception, IUT as slave

This test verifies if the IUT as slave ignores an unknown complete or unknown incomplete frame and does not set the response\_error signal to TRUE.

[Table 18](#) defines the test system for “unknown frame reception”.

**Table 18 — Test system: Unknown frame reception**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_UNKNOWN, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_UNKNOWN with the profile described in <a href="#">Table 19</a> followed by TST_FRM_STATUS_SIG.	
<b>Verification</b>	The IUT shall respond to TST_FRM_STATUS_SIG by sending the mandatory response_error signal = FALSE.	
<b>Reference</b>	ISO 17987-3:2016, 5.5.4	

[Table 19](#) defines the Test cases for “unknown frame reception”.

**Table 19 — Test cases: Unknown frame reception**

PT-CT-TC	Response field
[PT-CT 12].1	No response field
[PT-CT 12].2	Just the first data byte
[PT-CT 12].3	All data bytes (according to TST_FRAME specification or frame length according to LDF/NCF), but wrong checksum

### 7.11 [PT-CT 13] Length of header, IUT as master

This test verifies the correct length of the header  $T_{\text{HEADER}}$ .

[Table 20](#) defines the test system for “length of header”.

**Table 20 — Test system: Length of header**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_RX, TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT sends all defined TST_FRM_IUT_RX and all defined TST_FRM_IUT_TX.	
<b>Verification</b>	$T_{\text{HEADER\_MIN}} \leq T_{\text{HEADER}} \leq T_{\text{HEADER\_MAX}}$	
<b>Reference</b>	ISO 17987-3:2016, 5.2.3	

### 7.12 [PT-CT 14] Variation of length of header, IUT as slave

This test verifies if the IUT as slave recognizes headers with different length  $T_{\text{HEADER}}$ .

The test consists of four separate test cases. The length of break field low phase  $T_{\text{BRKFLD}}$ , break delimiter  $T_{\text{BRKDEL}}$  and the inter byte space between sync byte field and Identifier  $T_{\text{H\_INTERBYTE}}$  are set according to [Table 21](#).

**Table 21 — Test system: Variation of length of header**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_RDBI_0 followed by TST_HDR_SR_3D. Both frames use the length of the parts of header as defined in <a href="#">Table 22</a> .	
<b>Verification</b>	The IUT shall answer to the request.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.3	

[Table 22](#) defines the test cases for “variation of length of header”.

**Table 22 — Test cases: Variation of length of header**

PT-CT-TC	T <sub>BRKFLD</sub>	T <sub>BRKDEL</sub>	T <sub>H_INTERBYTE</sub>	≥T <sub>HEADER</sub>	T <sub>BIT</sub>
[PT-CT 14].1	T <sub>BRKFLD_MIN</sub>	T <sub>BRKDEL_MIN</sub>	0 T <sub>BIT</sub>	T <sub>HEADER_MIN</sub>	T <sub>BIT_MIN_MASTER</sub>
[PT-CT 14].2	18 T <sub>BIT</sub>	2 T <sub>BIT</sub>	6,6 T <sub>BIT</sub>	T <sub>HEADER_MAX</sub> - 1 T <sub>BIT</sub>	T <sub>BIT_MAX_MASTER</sub>
[PT-CT 14].3	15 T <sub>BIT</sub>	3 T <sub>BIT</sub>	2 T <sub>BIT</sub>	40 T <sub>BIT</sub>	T <sub>BIT_MAX_MASTER</sub>
[PT-CT 14].4	T <sub>BRKFLD_MIN</sub>	T <sub>BRKDEL_MIN</sub>	12,6 T <sub>BIT</sub>	T <sub>HEADER_MAX</sub> - 1 T <sub>BIT</sub>	T <sub>BIT_MAX_MASTER</sub>

### 7.13 [PT-CT 15] Bit rate tolerance, IUT as master

This test verifies if the master bit rate tolerance  $F_{TOL\_RES\_MASTER} < \pm 0,5 \%$ . For this test, a bit rate measurement is done in the sync byte field.

[Table 23](#) defines the test system for “bit rate tolerance”.

**Table 23 — Test system: Bit rate tolerance**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE
	bit rate	According to the IUT specification
<b>System Init</b>	Default	
<b>Test</b>	The IUT sends TST_FRM_ASSIGNIDRANGE with all bit rates defined in the IUT specification or LDF.	
<b>Verification</b>	$(\text{test case bit rate} - F_{TOL\_RES\_MASTER}) \leq \text{measured bit rate} \leq (\text{test case bit rate} + F_{TOL\_RES\_MASTER})$	
<b>Reference</b>	ISO 17987-4 (see Bit rate tolerance 12 V/24 V)	

### 7.14 [PT-CT 16] Bit rate tolerance, IUT as slave without making use of synchronization

This test verifies if the IUTs bit rate tolerance  $F_{TOL\_RES\_SLAVE} < \pm 1,5 \%$ .

For this test, a bit rate measurement is done in the slave answer. The measurement is done according to [5.9.4](#).

[Table 24](#) defines the test system for “bit rate tolerance”.

**Table 24 — Test system: Bit rate tolerance**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_X (Identifier = 2), TST_HDR_SR_3D
	bit rate	According to the IUT specification
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_RDBI_X (Identifier = 2) and TST_HDR_SR_3D according to <a href="#">Table 25</a> with all bit rate specified in the IUT specification or NCF $\pm 0,5 \%$ tolerance.	
<b>Verification</b>	The IUT shall answer to the request with negative response (RSID = 7F <sub>16</sub> ). $(\text{test system bit rate} - F_{TOL\_RES\_SLAVE}) \leq \text{measured slave bit rate of the slave answer} \leq (\text{test system bit rate} + F_{TOL\_RES\_SLAVE})$	
<b>Reference</b>	ISO 17987-4 (see Bit rate tolerance 12 V/24 V)	

[Table 25](#) defines the test cases for “bit rate tolerance”.

**Table 25 — Test cases: Bit rate tolerance**

PT-CT-TC	TST_FRAME	Bit rate tolerance
[PT-CT 16].1	TST_FRM_RDBI_X	+0,5 %
[PT-CT 16].2	TST_FRM_RDBI_X	-0,5 %

## 7.15 [PT-CT 17] Bit rate tolerance, IUT as slave with making use of synchronization

This test verifies if the IUT as slave is able to synchronize itself with a bit rate tolerance.

$F_{TOL\_SYNC} < \pm 2\%$  (relative value to the master node). For this test, a bit rate measurement is done in the slave answer.

NOTE This test case verifies the tolerance of a synchronized node. A slave node making use of synchronization and direct break detection may have a bit rate tolerance  $F_{TOL\_UNSYNC}$  of  $< \pm 14\%$  in case of lost synchronization.

The measurement is done according to [5.9.4](#).

[Table 26](#) defines the test system for “bit rate tolerance”.

**Table 26 — Test system: Bit rate tolerance**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_X (Identifier = 2), TST_HDR_SR_3D
	bit rate	Min/max bit rates according to the IUT specification
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_RDBI_X (Identifier = 2) followed by TST_HDR_SR_3D according to <a href="#">Table 27</a> with all bit rate specified in the IUT specification or $NCF \pm 0,5\%$ tolerance.	
<b>Verification</b>	The IUT shall answer to the request with negative response (RSID = 7F <sub>16</sub> ). (test system bit rate - $F_{TOL\_SYNC}$ ) $\leq$ measured slave bit rate of the slave answer $\leq$ (test system bit rate + $F_{TOL\_SYNC}$ ).	
<b>Reference</b>	ISO 17987-4 (see Bit rate tolerance 12 V/24 V)	

[Table 27](#) defines the test cases: bit rate tolerance.

**Table 27 — Test cases: Bit rate tolerance**

PT-CT-TC	TST_FRAME	Bit rate tolerance
[PT-CT 17].1	TST_HDR_SR_3D	+0,5 %
[PT-CT 17].2	TST_HDR_SR_3D	-0,5 %

## 7.16 Length of response

### 7.16.1 [PT-CT 18] Length of response, IUT as slave

This test verifies if the length of  $T_{FRAME\_MAX}$  is not exceeded by the IUTs in-frame-response space and inter-byte spaces.

[Table 28](#) defines the test system for “length of response (slave)”.

**Table 28 — Test system Length of response (slave)**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX according <a href="#">Table 29</a> <sup>a</sup>
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The header length is set to the maximum $T_{\text{HEADER\_MAX}}$ (worst case). The test system sends the headers of all TST_FRM_IUT_TX with different available data length.	
<b>Verification</b>	The IUT shall respond to the header with $T_{\text{Response}} \leq T_{\text{RESPONSE\_MAX}}$ The verification shall be done as shown in <a href="#">Figure 2</a> : Frame structure.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.3	

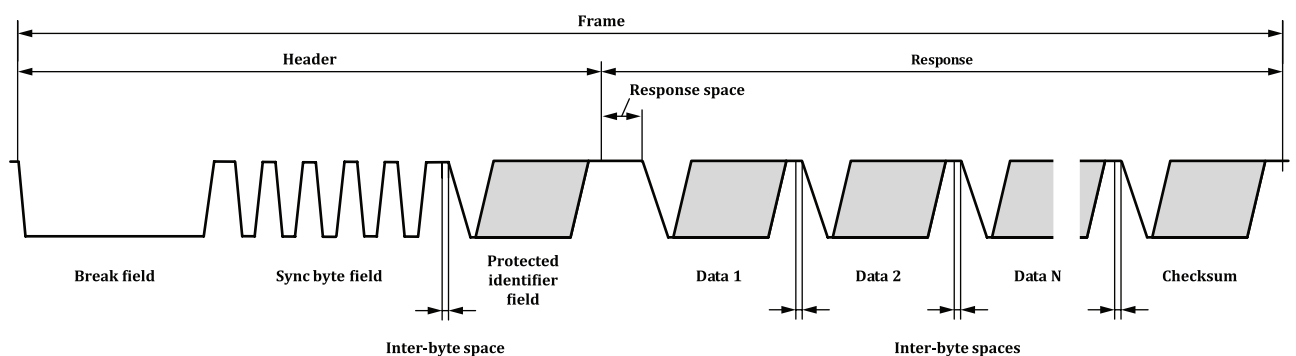
<sup>a</sup> If there is no TST\_FRM\_IUT\_TX available with the data length 8 byte, then use TST\_FRM\_RDBI\_0 according [Table 29](#) and TST\_HDR\_SR\_3D according [Table 29](#) or TST\_FRM\_RDBI\_X according [Table 29](#) and TST\_HDR\_SR\_3D according [Table 29](#).

[Table 29](#) defines the test cases for “length of response (slave)”.

**Table 29 — Test cases Length of response (slave)**

Requirements	Value
header size	maximum
break field low phase length	19 bit
break delimiter length	8 bit
$T_{\text{H\_Interbyte}}$	0

[Figure 2](#) shows the frame structure.



**Figure 2 — Frame structure**

### 7.16.2 [PT-CT 19] Length of response, IUT as master

This test verifies if the length of  $T_{\text{FRAME\_MAX}}$  is not exceeded by the IUT in-frame-response space and inter byte spaces.

[Table 30](#) defines the test system for “length of response (master)”.

**Table 30 — Test system: Length of response (master)**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT as master sends all TST_FRM_IUT_TX with the maximum length of header.	
<b>Verification</b>	The IUT shall send the response with $T_{\text{Response}} \leq T_{\text{RESPONSE\_MAX}}$ . The verification shall be done as shown in <a href="#">Figure 2</a> .	
<b>Reference</b>	ISO 17987-3:2016, 5.2.3	

### 7.16.3 [PT-CT 20] Acceptance of response field, IUT as slave

This test verifies if the IUT accepts frames with different lengths of the response field (variation of length of response space and inter byte space).

[Table 31](#) defines the test system for “acceptance of response field”.

**Table 31 — Test system: Acceptance of response field**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_RDBI_0 with the length of the response field as defined in <a href="#">Table 32</a> . The test system sends TST_HDR_SR_3D.	
<b>Verification</b>	The IUT shall answer to TST_HDR_SR_3D.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.3	

[Table 32](#) defines the test cases for “acceptance of response field”.



**Table 32 — Test cases: Acceptance of response field**

PT-CT-TC	Response space	Inter byte space	Comments
[PT-CT 20].1	0 T <sub>BIT</sub>	each space = 0 T <sub>BIT</sub>	T <sub>RESPONSE_MIN</sub>
[PT-CT 20].2	2 T <sub>BIT</sub>	each space = 4 T <sub>BIT</sub>	T <sub>RESPONSE_MAX</sub> - 2 T <sub>BIT</sub> (tolerance required due to response verification in LIN controller hardware)
[PT-CT 20].3	0 T <sub>BIT</sub>	space between data byte data 7 and data 8 = (T <sub>RESPONSE_MAX</sub> - T <sub>RESPONSE_MIN</sub> ) - 2 T <sub>BIT</sub> = 34 T <sub>BIT</sub> . other spaces = 0 T <sub>BIT</sub>	T <sub>RESPONSE_MAX</sub> - 2 T <sub>BIT</sub> Maximum inter-byte space if no response space exist (tolerance required due to response verification in LIN controller hardware)
[PT-CT 20].4	(T <sub>RESPONSE_MAX</sub> - T <sub>RESPONSE_MIN</sub> ) - 2 T <sub>BIT</sub> = 34 T <sub>BIT</sub>	each space = 0 T <sub>BIT</sub>	T <sub>RESPONSE_MAX</sub> - 2 T <sub>BIT</sub> Maximum response space if no inter byte space exist (tolerance required due to response verification in LIN controller hardware)

## 7.17 Verification of schedule table timing

### 7.17.1 [PT-CT 21] Verification of jitter, IUT as master

This test verifies the jitter based on one application schedule table defined in the LDF file.

[Table 33](#) defines the test system for “verification of jitter”.

**Table 33 — Test system: Verification of jitter**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	Defined schedule tables
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT as master sends out at least 100 consecutive frames of the schedule table.	
<b>Verification</b>	The jitter (see <a href="#">Figure 3</a> ) shall be compared with the defined jitter in the LDF. T <sub>FRAME_SLOT_MEASURE</sub> shall be measured between falling edges of the break field. T <sub>JITTER_MEASURE</sub> ≤ T <sub>JITTER_DEFINED</sub>	
<b>Reference</b>	ISO 17987-3:2016, 5.2.3	

[Figure 3](#) shows the frame structure and jitter.

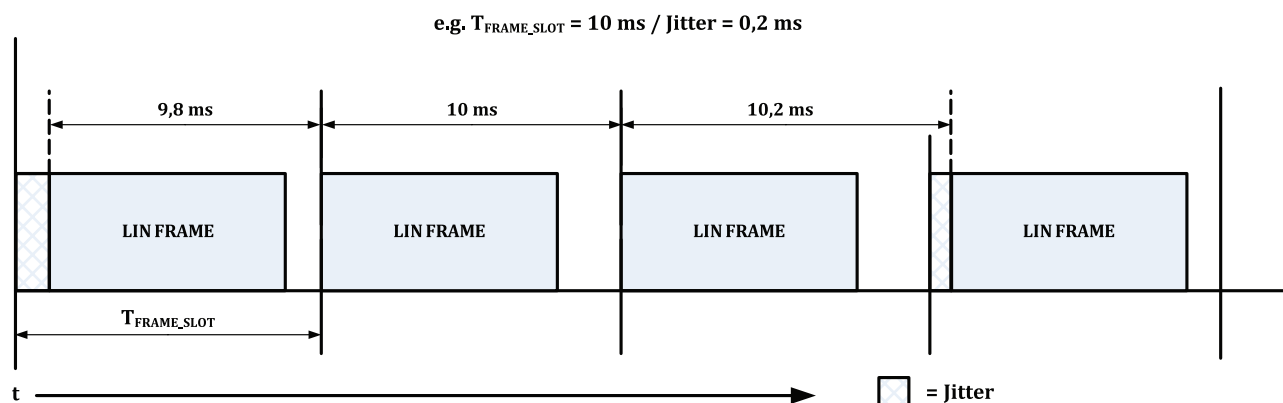


Figure 3 — Frame structure and jitter

### 7.17.2 [PT-CT 22] Schedule table management, IUT as master

This test verifies if the IUT as master respects  $T_{\text{FRAME\_SLOT}}$  space.

[Table 34](#) defines the test system for “schedule table management”.

Table 34 — Test system: Schedule table management

State	Description	
Set Up	IUT as	Master
	Classification	Class B, class C devices
	Configuration	Defined schedule tables, TST_FRM_IUT_RX
	bit rate	Default
System Init	Default	
Test	The IUT as master sends out at least 100 consecutive frames of the schedule table. The test system sends the response field of all the TST_FRM_IUT_RX frames as described in <a href="#">Table 35</a> .	
Verification	$T_{\text{FRAME\_SLOT\_SPECIFIED}} - \text{jitter\_ldf} < T_{\text{FRAME\_SLOT\_MEASURE}} < T_{\text{FRAME\_SLOT\_SPECIFIED}} + \text{jitter\_ldf}$ jitter_ldf is defined in the LDF file. $T_{\text{FRAME\_SLOT\_MEASURE}}$ shall be measured between falling edges of the break field. The IUT as master shall respect $T_{\text{FRAME\_SLOT}}$ spaces.	
Reference	ISO 17987-3:2016, 5.3.3	

[Table 35](#) defines the test cases for “schedule table management”.

Table 35 — Test cases: Schedule table management

PT-CT-TC	Response field of TST_FRM_IUT_RX frame
[PT-CT 22].1	The test system shall not send a response field.
[PT-CT 22].2	The test system sends only the first data byte of the response field.
[PT-CT 22].3	The test system sends the complete response field. The end of the response shall correspond with the start of the next slot.

### 7.18 [PT-CT 23] Sample point test, IUT as slave

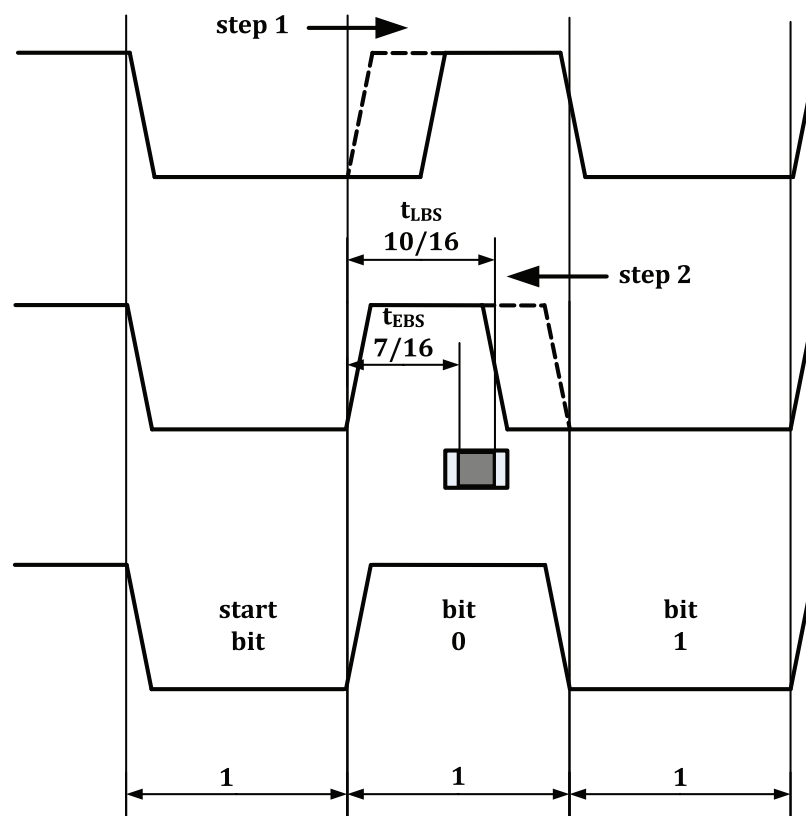
The bits of a byte field shall be sampled according to ISO 17987-4.

[Table 36](#) defines the test system for “sample point test”.

**Table 36 — Test system: Sample point test**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	<p>The test system sends TST_FRM_RDBI_0. After that, the test system sends TST_HDR_SR_3D as in the following:</p> <ul style="list-style-type: none"> <li>— The test system shall overwrite the recessive level of the PID bit 0 in 1 <math>\mu</math>s or alternatively 1/16 bit steps starting from the rising edge of the bit until the slave does not respond anymore.</li> <li>— In the second step, the test system shall overwrite the recessive level of the PID bit 0 in 1 <math>\mu</math>s or alternatively 1/16 bit steps starting from the falling edge of the bit towards the start bit until the slave does not respond any more.</li> </ul>	
<b>Verification</b>	<p>The device shall respond to the master node as long as the disturbance is outside the range <math>t_{EBS} - t_{LBS}</math> (see <a href="#">Figure 4</a>). For the execution of the test, <math>t_{LBS}</math> is taken with the fixed value of <math>10/16 t_{BIT}</math>.</p>	
<b>Reference</b>	ISO 17987-4:2016, 5.2.3	

[Figure 4](#) shows the sample point test.



**Figure 4 — Sample point test**

### 7.19 [PT-CT 24] Initialization time, IUT as slave

According to ISO 17987-2:2016, Figure 2 after power on the slave shall be initialized within 100 ms.

[Table 37](#) defines the test system for “Initialization time test”.

**Table 37 — Test system: Initialization time test**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system switches the power of the slave on and schedules TST_FRM_IUT_TX headers in each frame slot.	
<b>Verification</b>	The IUT shall reply to the first header latest after 100 ms. Verify the time when the slave provides the first response. The IUT shall furthermore respond on each frame header without missing responses after the first responded frame header.	
<b>Reference</b>	ISO 17987-2:2016, 11.2	

## 8 Communication without failure

### 8.1 Variation of LIN identifier

#### 8.1.1 [PT-CT 25] Variation of LIN PID, IUT as master

This test verifies if the IUT as master is able to send defined PIDs.

According to the device specification, all in the LDF defined identifiers shall be tested in one schedule table.

[Table 38](#) defines the test system for “variation of LIN PID”.

**Table 38 — Test system: Variation of LIN PID**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_RX, TST_FRM_IUT_TX, identifier as defined in ISO 17987-3:2016, 5.2.2.5.
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT sends TST_FRM_IUT_TX or TST_FRM_IUT_RX with all defined PID's.	
<b>Verification</b>	The IUT shall send the PID without failure. <sup>a</sup>	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.5	

<sup>a</sup> A complete ECU shall only use messages from the LDF. A software driver implementation shall use all PIDs.

#### 8.1.2 [PT-CT 26] Variation of LIN PIDs of subscribed frames, IUT as slave

This test verifies if the IUT as slave is able to recognize all PIDs (ID 0<sub>10</sub> to 59<sub>10</sub>) assigned to the first subscribed frame (according NCF, subclause frames).

[Table 39](#) defines the test system for “variation of LIN PIDs of subscribed frames”.

**Table 39 — Test system: Variation of LIN PIDs of subscribed frames**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE, TST_FRM_IUT_RX, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be assigned a PID and sent twice by the test system to ensure that the response_error signal is cleared. All assigned PIDs shall be unassigned before testing.	
<b>Test</b>	The test system sends TST_FRM_ASSIGNIDRANGE assigning an ID to TST_FRM_IUT_RX and TST_FRM_STATUS_SIG. The test system sends TST_FRM_IUT_RX with previously assigned ID and corrupted checksum (invert 4th bit). After that, TST_FRM_STATUS_SIG shall be sent by the test system. <sup>a</sup> In the next cycle, a different ID is used until all IDs are tested.	
<b>Verification</b>	The IUT shall respond to TST_FRM_STATUS_SIG by sending the mandatory response_error signal = TRUE.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.5	
<sup>a</sup> For the first test cycle, the TST_FRM_STATUS_SIG uses the ID 59 <sub>10</sub> , for the remaining test cycles TST_FRM_STATUS_SIG uses the ID 0 <sub>10</sub> .		

### 8.1.3 [PT-CT 27] Variation of LIN identifier of published frames, IUT as slave

This test verifies if the IUT as slave is able to recognize all PIDs (ID 0<sub>10</sub> to 59<sub>10</sub>) assigned to the first published frame (according NCF).

[Table 40](#) defines the test system for ‘variation of LIN identifier of published frames’.

**Table 40 — Test system: Variation of LIN identifier of published frames**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE, TST_FRM_IUT_TX
	bit rate	DEFAULT
<b>System Init</b>	All assigned PIDs shall be unassigned before testing.	
<b>Test</b>	The test system sends TST_FRM_ASSIGNIDRANGE with every ID. The test system sends the header of TST_FRM_IUT_TX.	
<b>Verification</b>	The IUT shall respond to the header of TST_FRM_IUT_TX according to the IUT specification.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.5	

## 8.2 Transmission of the checksum byte

### 8.2.1 [PT-CT 28] Transmission of the checksum byte “classic checksum”, IUT as slave

This test verifies if the IUT as slave sends a correct checksum byte.

[Table 41](#) defines the test system for transmission of the checksum byte “classic checksum”.

**Table 41 — Test system: Transmission of the checksum byte “classic checksum”**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_RDBI_0 followed by TST_HDR_SR_3D.	
<b>Verification</b>	The IUT shall answer to the request and the checksum byte shall be correct.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.7	

**8.2.2 [PT-CT 29] Transmission of the checksum byte “enhanced checksum”, IUT as slave**

This test verifies if the IUT as slave sends a correct checksum byte.

[Table 42](#) defines the test system for transmission of the checksum byte “enhanced checksum”.

**Table 42 — Test system: Transmission of the checksum byte “enhanced checksum”**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	All supported TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends the header of all supported TST_FRM_IUT_TX.	
<b>Verification</b>	The IUT shall respond to the header and the checksum byte shall be correct.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.7	

**8.2.3 [PT-CT 30] Transmission of the checksum byte “classic checksum”, IUT as master**

This test verifies if the IUT as master sends a correct checksum byte.

[Table 43](#) defines the test system for transmission of the checksum byte “classic checksum”.

**Table 43 — Test system: Transmission of the checksum byte “classic checksum”**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT as master sends TST_FRM_RDBI_0.	
<b>Verification</b>	The IUT shall send the correct checksum.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.7	

**8.2.4 [PT-CT 31] Transmission of the checksum byte of unconditional frames, IUT as master**

This test verifies that the IUT as master sends a correct checksum byte according to the checksum model of the subscribing slave.

[Table 44](#) defines the test system for “transmission of the checksum byte of unconditional frames”.

**Table 44 — Test system: Transmission of the checksum byte of unconditional frames**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT as master sends all supported TST_FRM_IUT_TX.	
<b>Verification</b>	The IUT shall send the correct checksum.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.7	

### 8.3 Unused bits

#### 8.3.1 [PT-CT 32] Unused bits, IUT as master

This test verifies if the IUT as master sends recessive bits if the TST\_FRM\_IUT\_TX contains unused data bits (bits which are not defined in the LDF).

[Table 45](#) defines the test system for “unused bits (master)”.

**Table 45 — Test system: Unused bits (master)**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT as master sends all TST_FRM_IUT_TX and unused data bits if applicable.	
<b>Verification</b>	The unused bits of TST_FRM_IUT_TX shall be sent as recessive bits.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.4.1	

#### 8.3.2 [PT-CT 33] Unused bits, IUT as slave

This test verifies if the IUT as slave sends recessive bits if the TST\_FRM\_IUT\_TX contains unused data bits (bits which are not defined in the NCF).

[Table 46](#) defines the test system for “unused bits (slave)”.

**Table 46 — Test system: Unused bits (slave)**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX
	bit rate	Default

**Table 46** (continued)

State	Description
<b>System Init</b>	Default
<b>Test</b>	The test system sends all TST_FRM_IUT_TX with unused data bits if applicable.
<b>Verification</b>	The unused bits of TST_FRM_IUT_TX shall be sent as recessive bits.
<b>Reference</b>	ISO 17987-3:2016, 5.2.4.1

## 8.4 Reserved frame

### 8.4.1 [PT-CT 34] Reserved frame, IUT as slave

This test verifies if the IUT as slave ignores a reserved frame header. The test system sends TST\_HDR\_RESERVE\_3E and TST\_HDR\_RESERVE\_3F.

[Table 47](#) defines the test system for “reserved frame (slave)”.

**Table 47 — Test system: Reserved frame (slave)**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_HDR_RESERVE_3E and TST_HDR_RESERVE_3F with default delay between the two headers.
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_HDR_RESERVE_3E and TST_HDR_RESERVE_3F.	
<b>Verification</b>	The IUT shall not answer to TST_HDR_RESERVE_3E and TST_HDR_RESERVE_3F.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.5	

### 8.5 [PT-CT 35] Diagnostic frame master request, IUT as master

This test verifies if the IUT as master is able to send master request frames. In this test case, the master request frame is checked.

[Table 48](#) defines the test system for “diagnostic frame master request”.

**Table 48 — Test system: Diagnostic frame master request**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE or, if not available TST_FRM_SLEEP_CMD
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT sends TST_FRM_ASSIGNIDRANGE.	
<b>Verification</b>	The IUT shall send the frame without failure. The data field length shall be 8 bytes. The checksum field shall be correct.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.4.5 and 6.3.6.5	



## 8.6 Supported frames according to the IUT specification

### 8.6.1 [PT-CT 36] Supported Tx frames according to the IUT specification, IUT as slave

This test verifies if the IUT as slave is able to send all supported frames according to the specification of the IUT.

For preconfigured slaves, this test shall be repeated without TST\_FRM\_ASSIGNIDRANGE. After power on reset, these slaves shall be operational and use frame IDs according to the LDF.

[Table 49](#) defines the test system for “supported Tx frames according to the IUT specification”.

**Table 49 — Test system: Supported Tx frames according to the IUT specification**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE, TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_ASSIGNIDRANGE. The test system sends all supported frames (TST_FRM_IUT_TX), according to IUT specification.	
<b>Verification</b>	The IUT responds to the requests as specified. The test system verifies by monitoring data length and checksum for received frames.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.5	

### 8.6.2 [PT-CT 37] Supported Rx frames according to the IUT specification, IUT as slave

This test verifies if the IUT as slave is able to receive all supported frames according to the specification of the IUT.

[Table 50](#) defines the test system for “Supported Rx frames according to the IUT specification”.

**Table 50 — Test system: Supported Rx frames according to the IUT specification**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE, TST_FRM_IUT_RX, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_ASSIGNIDRANGE (if more than four RX frames are supported by the IUT the TST_FRM_ASSIGNIDRANGE is repeated with appropriate configuration data). The test system transmits all supported frames TST_FRM_IUT_RX according to IUT specification. Every TST_FRM_IUT_RX is sent twice: first with failure (inverted checksum) then again without failure. TST_FRM_STATUS_SIG shall be sent in between.	
<b>Verification</b>	Recognition of received frames shall be verified by monitoring the response_error signal (in case with failure it shall be TRUE, in case without failure it shall be FALSE).	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.5	

## 9 Communication with failure

### 9.1 General

This clause verifies if the IUT does error and exception handling as specified in ISO 17987-3.

NOTE The fault confinement is not part of the LIN protocol, thus the way of verification is implementation specific.

### 9.2 [PT-CT 38] Bit error, IUT as slave

This test verifies if the IUT as slave detects a “bit error”, i.e. there is a difference between the bit value sent by the IUT and the corresponding bit value monitored by the IUT.

The test consists of separate test cases in order to test the “bit error” detection of the answer sent by the IUT as slave.

[Table 51](#) defines the test system for “bit error”.

**Table 51 — Test system: Bit error**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_X (Identifier = 2), TST_HDR_SR_3D, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_RDBI_X (Identifier = 2) followed by TST_HDR_SR_3D. The test system inverts bit in the slave answer sent by the IUT at the position as defined in <a href="#">Table 52</a> . Then TST_FRM_STATUS_SIG shall be sent by the test system to verify the response_error signal.	
<b>Verification</b>	The IUT shall detect the “bit error”, after detection of the “bit error” the transmission shall be aborted latest at the next byte boundary. The response_error signal shall be set.	
<b>Reference</b>	ISO 17987-3:2016, 5.4.3.3	

[Table 52](#) defines the test cases for “bit error”.

**Table 52 — Test cases: Bit error**

PT-CT-TC	Slave answer component/comments	Bit to be inverted <sup>a</sup>	Inter byte space
[PT-CT 38].1	Data byte 1	Stop bit	0
[PT-CT 38].2	Data byte 1	Bit 1	0
[PT-CT 38].3	Inter-byte data 1-2	Bit 1	≥1 <sup>b</sup>
[PT-CT 38].4	Data byte 2	Stop bit	0
[PT-CT 38].5	Data byte 2	Bit 2	0
[PT-CT 38].6	Inter byte data 2-3	Bit 1	≥1 <sup>b</sup>
[PT-CT 38].7	Data byte 3	Stop bit	0

<sup>a</sup> Counting starts with 0 at start bit of the respective slave answer component.

<sup>b</sup> Testable if the inter byte space is larger or equal than 1 bit. If the inter byte space is larger than 1, bit the inverted bit time shall be at the beginning of the space.

<sup>c</sup> The part of the verification to abort the transmission cannot be done.

Table 52 (continued)

PT-CT-TC	Slave answer component/comments	Bit to be inverted <sup>a</sup>	Inter byte space
[PT-CT 38].8	Data byte 3	Bit 3	0
[PT-CT 38].9	Inter byte data 3–4	Bit 1	≥1 <sup>b</sup>
[PT-CT 38].10	Data byte 4	Stop bit	0
[PT-CT 38].11	Data byte 4	Bit 4	0
[PT-CT 38].12	Inter byte data 4–5	Bit 1	≥1 <sup>b</sup>
[PT-CT 38].13	Data byte 5	Stop bit	0
[PT-CT 38].14	Data byte 5	Bit 5	0
[PT-CT 38].15	Inter byte data 5–6	Bit 1	≥1 <sup>b</sup>
[PT-CT 38].16	Data byte 6	Stop bit	0
[PT-CT 38].17	Data byte 6	Bit 6	0
[PT-CT 38].18	Inter byte data 6–7	Bit 1	≥1 <sup>b</sup>
[PT-CT 38].19	Data byte 7	Stop bit	0
[PT-CT 38].20	Data byte 7	Bit 7	0
[PT-CT 38].21	Inter byte data 7–8	Bit 1	≥1 <sup>b</sup>
[PT-CT 38].22	Data byte 8	Stop bit	0
[PT-CT 38].23	Data byte 8	Bit 8	0
[PT-CT 38].24 <sup>c</sup>	Inter byte data 8-checksum	Bit 1	≥1 <sup>b</sup>
[PT-CT 38].25 <sup>c</sup>	checksum field	Stop bit	0

<sup>a</sup> Counting starts with 0 at start bit of the respective slave answer component.

<sup>b</sup> Testable if the inter byte space is larger or equal than 1 bit. If the inter byte space is larger than 1, bit the inverted bit time shall be at the beginning of the space.

<sup>c</sup> The part of the verification to abort the transmission cannot be done.

### 9.3 [PT-CT 39] Framing error in header of published frame, IUT as slave

This test case verifies if a slave does not respond to a header with inverted stop bit of the PID.

Table 53 defines the test system for “framing error in header of published frame”.

Table 53 — Test system: Framing error in header of published frame

State	Description	
Set Up	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX, TST_FRM_STATUS_SIG
	bit rate	Default
System Init	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
Test	The test system sends the header of TST_FRM_IUT_TX and inverts the stop bit of the PID. Then the test system sends TST_FRM_STATUS_SIG.	
Verification	TST_FRM_IUT_TX shall not be answered. The response_error signal shall be FALSE.	
Reference	ISO 17987-3:2016, 5.4.3.3	

#### 9.4 [PT-CT 40] Framing error in response field of subscribed frame, IUT as slave

This test case verifies if a slave detects a framing error (inverted stop bit of the first data byte) in a subscribed frame and sets the response\_error signal.

[Table 54](#) defines the test system for “framing error in response field of subscribed frame”.

**Table 54 — Test system: Framing error in response field of subscribed frame**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_RX, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_IUT_RX with inverted stop bit of the first data byte. Then the test system sends TST_FRM_STATUS_SIG.	
<b>Verification</b>	The response_error signal shall be set. For UART based implementations, the test case shall be executed on the second data byte (applicable if TST_FRM_IUT_RX has a minimum length of two byte).	
<b>Reference</b>	ISO 17987-3:2016, 5.4.3.3	

#### 9.5 [PT-CT 41] Checksum error by inversion, IUT as slave

This test verifies if the IUT as slave detects a wrong classic checksum.

[Table 55](#) defines the test system for “checksum error by inversion”.

**Table 55 — Test system: Checksum error by inversion**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_RDBI_0. The test system inverts the first bit of the checksum byte of the command of TST_FRM_RDBI_0. TST_FRM_STATUS_SIG follows to request for response_error signal.	
<b>Verification</b>	The verification is done by monitoring the response_error signal = TRUE.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.7	

#### 9.6 [PT-CT 42] Checksum error by carry, IUT as slave

This test verifies if the IUT as slave does not add the carry on the sum of all data bytes and checksum byte during the verification of checksum. The slave shall detect a wrong classic checksum where the checksum calculation over all response bytes and the checksum byte results in 1FE<sub>16</sub>.

[Table 56](#) defines the test system for “checksum error by carry”.

**Table 56 — Test system: Checksum error by carry**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_STATUS_SIG, TST_FRM_MR_3C_SF
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_MR_3C_SF. The data content of TST_FRM_MR_3C_SF is defined as follows. Data byte 2 to 8 is 00 <sub>16</sub> . Data byte 1 and checksum is FF <sub>16</sub> . <sup>a</sup> TST_FRM_STATUS_SIG follows to request the response_error signal.	
<b>Verification</b>	The Verification is done by monitoring the response_error signal = TRUE.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.7	
<sup>a</sup> Checksum FF <sub>16</sub> is the inverted value, the correct checksum is 00 <sub>16</sub> .		

## 9.7 [PT-CT 43] Communication robustness, IUT as slave

This test verifies if the IUT as slave is able to ignore incomplete headers with different (invalid) values for the sync byte fields and continue with communication in the next slot.

[Table 57](#) defines the test system for “Communication robustness”.

**Table 57 — Test system: Communication robustness**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends an incomplete frame header without PID in a loop with values for the sync byte field in the range 00 <sub>16</sub> to FF <sub>16</sub> . Next, the test system sends TST_FRM_IUT_TX.	
<b>Verification</b>	The slave shall be able to respond to TST_FRM_IUT_TX after the previous incorrect header.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.2.7	

## 10 Event triggered frames

### 10.1 General

These test cases are only applicable if event triggered frames are supported by the IUT.

### 10.2 [PT-CT 44] Event triggered frame, IUT as slave

This test verifies if the IUT as slave is able to answer to event triggered frames, mandatory if the IUT supports event triggered frames.

[Table 58](#) defines the test system for “event triggered frame”.

**Table 58 — Test system: Event triggered frame**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_REQ_EVT
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	<p>The test system or a manually stimulation is triggering an update of a signal carried by TST_FRM_REQ_EVT (e.g. by pulling a switch).</p> <p>The test system sends the header of TST_FRM_REQ_EVT.</p> <p>The test system sends again the header of TST_FRM_REQ_EVT.</p>	
<b>Verification</b>	<p>The slave shall respond to the first header of TST_FRM_REQ_EVT with the first data byte equal to the PID of the corresponding unconditional frame.</p> <p>Then, the slave shall not answer to the second header.</p>	
<b>Reference</b>	ISO 17987-3:2016, 5.2.4.3	

### 10.3 Event triggered frame with collision

#### 10.3.1 [PT-CT 45] Event triggered frame with collision resolving, IUT as slave

This test verifies if the IUT as slave is able to answer to event triggered frames, mandatory if the IUT supports event triggered frames.

[Table 59](#) defines the test system for “event triggered frame with collision resolving”.

**Table 59 — Test system: Event triggered frame with collision resolving**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_REQ_EVT, TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	<p>The test system or a manually stimulation is triggering an update of a signal carried by TST_FRM_REQ_EVT (e.g. by pulling a switch).</p> <p>The test system sends the header of TST_FRM_REQ_EVT and produces a bit error in the response of the slave.</p> <p>The test system shall resolve the collision by sending the corresponding TST_FRM_IUT_TX according to the NCF.</p> <p>The test system sends the header of TST_FRM_REQ_EVT without the stimulation of the input triggering.</p>	
<b>Verification</b>	The IUT as slave shall not respond to the second header of the event triggered frame.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.4.3	

#### 10.3.2 [PT-CT 46] Event triggered frame with errors in collision resolving, IUT as slave

This test verifies if the IUT as slave is able to answer to event triggered frames, mandatory if the IUT supports event triggered frames.

The focus of this test case is verifying whether the slave node is able to store the event until it is sent successfully.

Table 60 defines the test system for “event triggered frame with errors in collision resolving”.

**Table 60 — Test system: Event triggered frame with errors in collision resolving**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_REQ_EVT, TST_FRM_IUT_TX
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	<p>The test system or a manually stimulation is triggering an update of a signal carried by TST_FRM_REQ_EVT (e.g. by pulling a switch).</p> <p>The test system sends the header of TST_FRM_REQ_EVT and produces a bit error in the response of the slave by overwriting the first data byte of the response with 00<sub>16</sub>.</p> <p>The test system resolves the collision by sending the corresponding TST_FRM_IUT_TX according to the NCF, but it overwrites the first data byte of the slave response with 00<sub>16</sub>.</p> <p>The test system sends the header of TST_FRM_REQ_EVT again.</p>	
<b>Verification</b>	The IUT as slave shall respond to the second header of the event triggered frame TST_FRM_REQ_EVT with the first data byte equal to the PID of the corresponding unconditional frame, because the event triggered frame could not be sent successfully.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.4.3	

### 10.3.3 [PT-CT 47] Event triggered frame with collision resolving, IUT as master

This test verifies if the IUT as master is able to resolve collisions in event triggered frames, mandatory if the IUT supports event triggered frames.

Table 61 defines the test system for “event triggered frame with collision resolving (master)”.

**Table 61 — Test system: Event triggered frame with collision resolving (master)**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_REQ_EVT
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT as master sends the header of TST_FRM_REQ_EVT. The test system responds with only one data byte with 00 <sub>16</sub> .	
<b>Verification</b>	The IUT as master shall switch to the event triggered resolving schedule table. The collision resolving schedule table is sent once. After the collision resolving schedule table, the master shall switch back to the previous schedule table. The master shall continue with the schedule entry subsequent to the schedule entry where the collision occur.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.4.3	

### 10.3.4 [PT-CT 48] Error in transmitted frame with collision, IUT as slave

This test verifies if the IUT as slave is able to report its status to the network.

The collision shall not effect status reporting.

Table 62 defines the test system for “error in transmitted frame with collision”.

**Table 62 — Test system: Error in transmitted frame with collision**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_REQ_EVT, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	IUT shall be setup to answer to TST_FRM_REQ_EVT (IUT specific event occurred). TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_REQ_EVT and the response to cause a collision. After that, TST_FRM_STATUS_SIG is sent from the test system.	
<b>Verification</b>	The IUT shall answer to TST_FRM_STATUS_SIG and sends the mandatory response_error signal = FALSE.	
<b>Reference</b>	ISO 17987-3:2016, 5.2.4.3	

## 11 Status management

### 11.1 [PT-CT 49] Error in received frame, IUT as slave

This test verifies if the IUT as slave is able to report its status to the network.

[Table 63](#) defines the test system for “error in received frame (slave)”.

**Table 63 — Test system: Error in received frame (slave)**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_ASSIGNIDRANGE with inverted stop bit of the first data byte. Then, the test system sends TST_FRM_STATUS_SIG.	
<b>Verification</b>	The IUT shall answer to TST_FRM_STATUS_SIG and sends the mandatory response_error signal = TRUE.  For UART based implementations, the test case shall be executed on the second data byte.	
<b>Reference</b>	ISO 17987-3:2016, 5.5.4	

### 11.2 [PT-CT 50] Error in transmitted frame, IUT as slave

This test verifies if the IUT as slave is able to report its status to the network.

[Table 64](#) defines the test system for “error in transmitted frame”.



**Table 64 — Test system: Error in transmitted frame**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_RDBI_0, followed by TST_HDR_SR_3D. The test system disturbs the frame response from the slave as defined in <a href="#">Table 65</a> . After that, TST_FRM_STATUS_SIG is sent from the test system.	
<b>Verification</b>	The IUT shall answer to TST_FRM_STATUS_SIG and sends the mandatory response_error signal = TRUE.	
<b>Reference</b>	ISO 17987-3:2016, 5.5.4	

[Table 65](#) defines the test cases for “error in transmitted frame”.

**Table 65 — Test cases: Error in transmitted frame**

PT-CT-TC	TST_HDR_SR_3D
[PT-CT 50].1	One inverted checksum data bit.
[PT-CT 50].2	Inverted stop bit of data byte 1.

### 11.3 [PT-CT 51] response\_error signal handling, IUT as slave

This test verifies if the IUT as slave is able to handle the response\_error signal correctly.

[Table 66](#) defines the test system for “response\_error signal handling (slave)”.

**Table 66 — Test system: response\_error signal handling (slave)**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_RX, TST_FRM_STATUS_SIG
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	The test system sends TST_FRM_IUT_RX and inverts one checksum data bit. Then, the test system sends TST_FRM_IUT_RX correctly. After that, TST_FRM_STATUS_SIG is sent twice from the test system.	
<b>Verification</b>	The IUT shall answer to the first TST_FRM_STATUS_SIG and sends the mandatory response_error signal = TRUE. Then, the IUT shall answer to the second TST_FRM_STATUS_SIG and sends the mandatory response_error signal = FALSE.	
<b>Reference</b>	ISO 17987-3:2016, 5.5.4	

## 12 Sleep/wake up/power mode tests

### 12.1 [PT-CT 52] Send “go-to-sleep command”, IUT as master

This test verifies if the IUT as master is able to send a go-to-sleep command.

In this test case, the “go-to-sleep command” is checked.

[Table 67](#) defines the test system for send “go-to-sleep command”.

**Table 67 — Test system: Send “go-to-sleep command”**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	TST_FRM_SLEEP_CMD
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT is caused to send go-to-sleep command.	
<b>Verification</b>	<p>The IUT shall send the frame without failure and the frame content shall be as defined in ISO 17987-2:2016, 5.1.4.</p> <p>The IUT shall stop sending after completion of this message and the LIN bus shall be in recessive state.</p>	
<b>Reference</b>	ISO 17987-2:2016, 5.1.4	

## 12.2 [PT-CT 53] Receive “go-to-sleep command”, IUT as slave

This test verifies if the IUT as slave is able to receive a go-to-sleep command.

In this test case, the “go-to-sleep command” is checked.

[Table 68](#) defines the test system for receive “go-to-sleep command”.

**Table 68 — Test system: Receive “go-to-sleep command”**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_SLEEP_CMD
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	See <a href="#">Table 69</a>	
<b>Verification</b>	See <a href="#">Table 69</a>	
<b>Reference</b>	ISO 17987-2, 5.1.4	

[Table 69](#) defines the test cases for receive “go-to-sleep command”.

**Table 69 — Test cases: Receive “go-to-sleep command”**

PT-CT-TC	Test	Verification
[PT-CT 53].1	The test system sends TST_FRM_SLEEP_CMD with data bytes 2 to 8 are filled with FF <sub>16</sub> .	The IUT shall receive the frame without failure. The IUT shall enter sleep state <sup>a</sup> instantly.
[PT-CT 53].2	The test system sends TST_FRM_SLEEP_CMD with data bytes 2 to 8 are filled with 01 <sub>16</sub> , 02 <sub>16</sub> , 03 <sub>16</sub> , 04 <sub>16</sub> , 05 <sub>16</sub> , 06 <sub>16</sub> , 07 <sub>16</sub> .	The IUT shall receive the frame without failure. The IUT shall enter sleep state <sup>a</sup> instantly. The IUT shall ignore the data content of the data byte 2 to 8.

<sup>a</sup> Verification method should be selected according to [5.7.5](#).

### 12.3 [PT-CT 54] Receive a wake up signal, IUT as master

This test verifies if the IUT as master is able to receive a wake up signal and to wake up after having received the request if applicable.

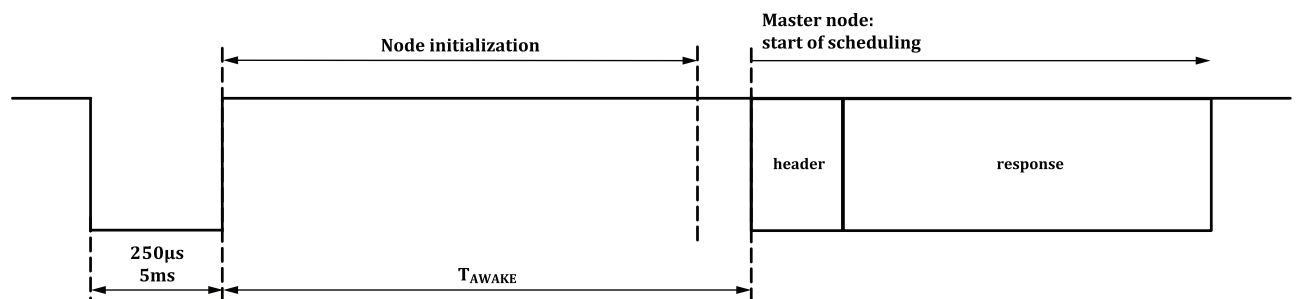
If there is a slave node in the network which is able to send a wake up signal, the master node shall also wake up and, when the slave nodes are ready (according to ISO 17987-2:2016, Clause 5), start transmitting headers to find out the cause (using signals) of the wake up.

[Table 70](#) defines the test system for “receive a wake up signal (master)”.

**Table 70 — Test system: Receive a wake up signal, IUT as master**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	Wake up signal
	bit rate	Default
<b>System Init</b>	The IUT is in sleep state.	
<b>Test</b>	The test system sends a wake up signal with a dominant state of 250 $\mu$ s.	
<b>Verification</b>	<p>The IUT shall receive the wake up signal without failure and shall run through the start-up procedure.</p> <p>The IUT shall start to send frame headers according to ISO 17987-2:2016, Clause 5 and as depicted in <a href="#">Figure 5</a>.</p> <p>The time <math>T_{AWAKE}</math> shall be measured.</p>	
<b>Reference</b>	ISO 17987-2:2016, 5.1.3	

[Figure 5](#) shows the wake up signal and start of scheduling, IUT as master.



**Figure 5 — Receive a wake up signal, IUT as master**

### 12.4 [PT-CT 55] Receive a wake up signal, IUT as slave

This test verifies if the IUT as slave is able to receive a wake up signal and to wake up after having received the request.

[Table 71](#) defines the test system for “receive a wake up signal (slave)”.

**Table 71 — Test system: Receive a wake up signal, IUT as slave**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	wake up signal, TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	The IUT is in sleep state.	
<b>Test</b>	The test system sends a wake up signal as defined in <a href="#">Table 72</a> . After 100 ms, TST_FRM_RDBI_0 is sent from the test system, followed by TST_HDR_SR_3D.	
<b>Verification</b>	The IUT shall receive the wake up signal without failure. The IUT shall run through the start-up procedure and answer to master request.	
<b>Reference</b>	ISO 17987-2:2016, 5.1.3	

[Table 72](#) defines the test cases for “receive a wake up signal (slave)”.

**Table 72 — Test cases: Receive a wake up signal (slave)**

PT-CT-TC	Wake up signal
[PT-CT 55].1	250 $\mu$ s
[PT-CT 55].2	5 ms
[PT-CT 55].3	5 nominal bit times

## 12.5 Send a wake up signal

### 12.5.1 [PT-CT 56] Send a wake up signal, IUT as master and IUT as slave

This test only applies to slaves which are allowed to wake up the bus (see NCF or datasheet) and masters which support the wake up signal.

[Table 73](#) defines the test system for “send a wake up signal (master and slave)”.

**Table 73 — Test system: Send a wake up signal (master and slave)**

State	Description	
<b>Set Up</b>	IUT as	Slave, master
	Classification	Class B, class C devices
	Configuration	no definitions required
	bit rate	Default
<b>System Init</b>	The IUT is in sleep state.	
<b>Test</b>	The IUT is caused to send a wake up signal.	
<b>Verification</b>	The IUT shall send the wake up signal according to ISO 17987-2:2016, 5.1.3 (250 $\mu$ s < t < 5 ms)	
<b>Reference</b>	ISO 17987-2:2016, 5.1.3	

### 12.5.2 [PT-CT 57] Send a block of wake up signals, IUT as slave

This test only applies to slaves which are allowed to wake up the bus (see NCF or datasheet).

This test verifies if the IUT as slave repeats a wake up signal if there is no scheduling started by the master after having sent the wake up signal.

[Table 74](#) defines the test system for “send a block of wake up signals (slave)”.

**Table 74 — Test system: Send a block of wake up signals (slave)**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	no definitions required
	bit rate	Default
<b>System Init</b>	The IUT is in sleep state.	
<b>Test</b>	The IUT is caused to send a wake up signal. There is no answer from the test system for the duration of this test.	
<b>Verification</b>	The IUT shall send the wake up signal according to ISO 17987-2:2016, 5.1.3 (250 $\mu$ s < t < 5 ms) The IUT as slave shall transmit a second and third wake up signal each within a distance of 150 ms to 250 ms as defined in ISO 17987-2:2016, 5.1.3. The verification shall be done for the first 3 wake up pulses only.	
<b>Reference</b>	ISO 17987-2:2016, 5.1.3	

### 12.5.3 [PT-CT 58] Wait after one block of wake up signals, IUT as slave

This test only applies to slaves which are allowed to wake up the bus (see NCF or datasheet).

This test verifies if the IUT as slave repeats a wake up signal if there is no scheduling started by the master after having sent the wake up signal. The slave node shall not send another wake up signal before 1,5 s after the last block of wake up signals.

[Table 75](#) defines the test system for “wait after one block of wakeup signals”.

**Table 75 — Test system: Wait after one block of wakeup signals**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	no definitions required
	bit rate	Default
<b>System Init</b>	The IUT is in sleep state.	
<b>Test</b>	The IUT is caused to send a wake up signal. There is no answer from a master for the duration of this test.	
<b>Verification</b>	The IUT shall send the wake up signal according to ISO 17987-2:2016, 5.1.3 (250 $\mu$ s < t < 5 ms). The IUT as slave shall transmit a second and third wake up signal each within 150 ms to 250 ms distance as defined in ISO 17987-2:2016, 5.1.3. The slave node shall not send another wake up signal before 1,5 s after the last wake up signal.	
<b>Reference</b>	ISO 17987-2:2016, 5.1.3	

### 12.5.4 [PT-CT 59] Send a wake up signal, frame header from a master following, IUT as slave

This test verifies if the IUT as slave stops repeating wake up signals if there is a frame header from a master.

[Table 76](#) defines the test system for “send a wake up signal, frame header from a master following”.

**Table 76 — Test system: Send a wake up signal, frame header from a master following**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	The IUT is in sleep state.	
<b>Test</b>	The IUT is caused to send wake up signals. The test system sends TST_FRM_RDBI_0 100 ms after second wake up signal. After that, TST_HDR_SR_3D is sent by the test system.	
<b>Verification</b>	The IUT shall send the wake up signal according to ISO 17987-2:2016, 5.1.3 (250 μs < t < 5 ms). The IUT retransmits a wake up signal after 150 ms – 250 ms. The IUT shall stop retransmission of wake up signals and answer to TST_HDR_SR_3D.	
<b>Reference</b>	ISO 17987-2:2016, 5.1.3	

## 12.6 [PT-CT 60] ECU power loss, IUT as master

This test only applies to slaves which are allowed to wake up the bus (see NCF or datasheet) and masters which support the wake up signal.

This test verifies if the IUT as master is able to recover communication after a power cycle.

[Table 77](#) defines the test system for “communication recovery after a power cycle”.

**Table 77 — Test system: Communication recovery after a power cycle**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	—
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	Apply power to the master device. The test system triggers a LIN wake up signal if the network is not woken up by the master node already. After 10 s, remove power from the master long enough to power down and force a reset of the master. Re-apply power to the master. The test system triggers a LIN wake up signal if the network is not woken up by the master node already.	
<b>Verification</b>	Monitor the LIN communication. If a wake up is triggered by the test system, verify that the master does not transmit a second wake up signal. Verify that scheduling is started within the time specified in ISO 17987-2. After a master reset, communication should recover again.	
<b>Reference</b>	ISO 17987-2:2016, Clause 5 SAE J2602-1:2012, 7.10.1	

## 12.7 [PT-CT 61] Powered up with LIN shorted, IUT as master

This test verifies if the IUT as master is able to reinit LIN communication after a short.

[Table 78](#) defines the test system for “powered up with LIN shorted, IUT as master”.

**Table 78 — Test system: Powered up with LIN shorted, IUT as master**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	—
	bit rate	Default
<b>System Init</b>	ECU in power down state.	
<b>Test</b>	<p>Short the LIN bus to ground/<math>V_{Bat}</math>.</p> <p>Apply power to the master and remove the LIN short after <math>T_{BUS\_SHORT}</math>.</p> <p>After removing the short wake, the master if not already woken up by the master node.</p> <p>The test is repeated for the following <math>T_{BUS\_SHORT}</math> timings:</p> <p>— 1 s, 2 s, 4 s, 10 s, 20 s, 40 s and 1 min.</p> <p>This test cycle is performed first for LIN bus short to ground and then with short to <math>V_{Bat}</math>.</p>	
<b>Verification</b>	Verify LIN Master reinitiates communications according to ISO 17987-2.	
<b>Reference</b>	ISO 17987-2:2016, Clause 5 SAE J2602-1:2012, 7.10.2	

## 12.8 [PT-CT 62] LIN shorted before scheduling, IUT as master

This test verifies if the IUT as master is able to recover communication when short to ground before start of scheduling.

[Table 77](#) defines the test system for “LIN shorted to ground before scheduling, IUT as master”.

**Table 79 — Test system: LIN shorted to ground before scheduling, IUT as master**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	—
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	<p>Apply power to the master and force the master to start communication.</p> <p>Short the LIN bus to ground/<math>V_{Bat}</math> 10 ms before the master node would transmit the first header/frame. The bus short shall be maintained for <math>T_{Short}</math> (the IUT timing for start of scheduling shall be known or measured before).</p> <p>The test shall be executed for the following <math>T_{Short}</math> values: 100 ms, 1 s, 4 s, 10 s, 1 min.</p> <p>It is allowed to wake the master if it has gone to sleep only for the 1 min short duration.</p> <p>This test cycle is performed first for LIN bus short to ground and then with short to <math>V_{Bat}</math>.</p>	
<b>Verification</b>	Verify LIN master reinitiates communications within the application specified time after $T_{Short}$ .	
<b>Reference</b>	ISO 17987-2:2016, Clause 5 SAE J2602-1:2012, 7.10.2	

## 12.9 [PT-CT 63] LIN shorted after start of scheduling, IUT as master

This test verifies if the IUT as master is able to recover communication when shorted after scheduling has started.

[Table 77](#) defines the test system for “LIN shorted after start of scheduling, IUT as master”.

**Table 80 — Test system: LIN shorted after start of scheduling, IUT as master**

State	Description	
<b>Set Up</b>	IUT as	Master
	Classification	Class B, class C devices
	Configuration	—
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	<p>Apply power to the master.</p> <p>10 ms to 20 ms after start of scheduling the LIN bus is shot to ground/<math>V_{Bus}</math>. The bus short shall be maintained for <math>T_{Short}</math>.</p> <p>The test shall be executed for the following <math>T_{Short}</math> values: 100 ms, 1 s, 1 min.</p> <p>It is allowed to wake the master if it has gone to sleep only for the 1 min short duration.</p> <p>This test cycle is performed first for LIN bus short to ground and then with short to <math>V_{Bat}</math>.</p>	
<b>Verification</b>	Verify LIN master reinitiates communications within the application specified time after $T_{Short}$ .	
<b>Reference</b>	ISO 17987-5:2016, Clause 5 SAE J2602-1:2012, 7.10.2	

## 13 Sleep state after bus idle

### 13.1 [PT-CT 64] Sleep state after event and bus idle, IUT as slave

This test verifies if the IUT as slave automatically enters sleep state if the bus is inactive for more than 4 s.

[Table 81](#) defines the test system for “sleep state after bus idle”.

**Table 81 — Test system: Sleep state after bus idle**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	The IUT is in operational state.	
<b>Test</b>	See <a href="#">Table 82</a> .	
<b>Verification</b>	See <a href="#">Table 82</a> .	
<b>Reference</b>	ISO 17987-2:2016, 5.1.4	

[Table 82](#) defines the test cases: sleep state after bus idle.



**Table 82 — Test cases: Sleep state after bus idle**

PT-CT-TC	Test	Verification
[PT-CT 64].1	After sending TST_FRM_RDBI_0 and TST_HDR_SR_3D, the test system receives the response from the IUT. The bus is idle after that.	The IUT shall enter the sleep state after $t_{\text{SLEEP}}$ , where $4 \text{ s} < t_{\text{SLEEP}} < 10 \text{ s}^{\text{a}}$
[PT-CT 64].2	The IUT receives a wake up signal ( $5 T_{\text{BIT}}$ ), after that, the IUT is connected to recessive level	After receiving the first wake up signal, the IUT shall enter operational state.  The IUT shall enter the sleep state (from the start of the connection to recessive level) after $t_{\text{SLEEP}}$ , where $4 \text{ s} < t_{\text{SLEEP}} < 10 \text{ s}^{\text{a}}$
[PT-CT 64].3	The IUT receives a wake up signal ( $5 T_{\text{BIT}}$ ), after that, the IUT is connected to dominant level for 10 s. Between the wake up signal and the IUT connection to dominant level is a one $T_{\text{BIT}}$ recessive level. After that, there is a one ms recessive level. Then, a wakeup signal is sent from the test system.	After receiving the first wake up signal, the IUT shall enter operational state. The IUT shall enter the sleep state (from the start of the connection to dominant level) after $t_{\text{SLEEP}}$ , where $4 \text{ s} < t_{\text{SLEEP}} < 10 \text{ s}^{\text{a}}$  After receiving the second wake up signal, the IUT shall enter operational state.
[PT-CT 64].4	The test system sends a break field and a sync byte field. The bus is recessive after that.	The IUT shall enter the sleep state after $t_{\text{SLEEP}}$ , where $4 \text{ s} < t_{\text{SLEEP}} < 10 \text{ s}^{\text{a}}$
[PT-CT 64].5	The test system sends a break field and a sync byte field. After that, the IUT is connected to dominant level.	The IUT shall enter the sleep state after $t_{\text{SLEEP}}$ , where $4 \text{ s} < t_{\text{SLEEP}} < 10 \text{ s}^{\text{a}}$
[PT-CT 64].6	The test system sends TST_FRM_RDBI_0 up to the 3rd data byte and stops sending. The bus is recessive after that.	The IUT shall enter the sleep state after $t_{\text{SLEEP}}$ , where $4 \text{ s} < t_{\text{SLEEP}} < 10 \text{ s}^{\text{a}}$
[PT-CT 64].7	The test system sends TST_FRM_RDBI_0 up to the 3rd data byte and stops sending. After that, the IUT is connected to dominant level.	The IUT shall enter the sleep state after $t_{\text{SLEEP}}$ , where $4 \text{ s} < t_{\text{SLEEP}} < 10 \text{ s}^{\text{a}}$

<sup>a</sup> Verification method should be selected according to 5.7.5.

### 13.2 [PT-CT 65] Sleep state after bus idle with power up and wake up signal, IUT as slave

This test verifies if the IUT as slave automatically enters a sleep state if the bus is inactive for more than 4 s after a wake up signal.

Table 83 defines the test system for “sleep state after bus idle with power up and wake up signal”.

**Table 83 — Test system: Sleep state after bus idle with power up and wake up signal**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	—
	bit rate	Default
<b>System Init</b>	The IUT gets a power up.	
<b>Test</b>	The test system sends a wake up signal after 100 ms.	
<b>Verification</b>	The IUT shall enter the sleep state after $t_{\text{SLEEP}}$ , where $4 \text{ s} < t_{\text{SLEEP}} < 10 \text{ s}$	
<b>Reference</b>	ISO 17987-2:2016, 5.1.4	

### 13.3 [PT-CT 66] Timeout after bus idle, IUT as slave

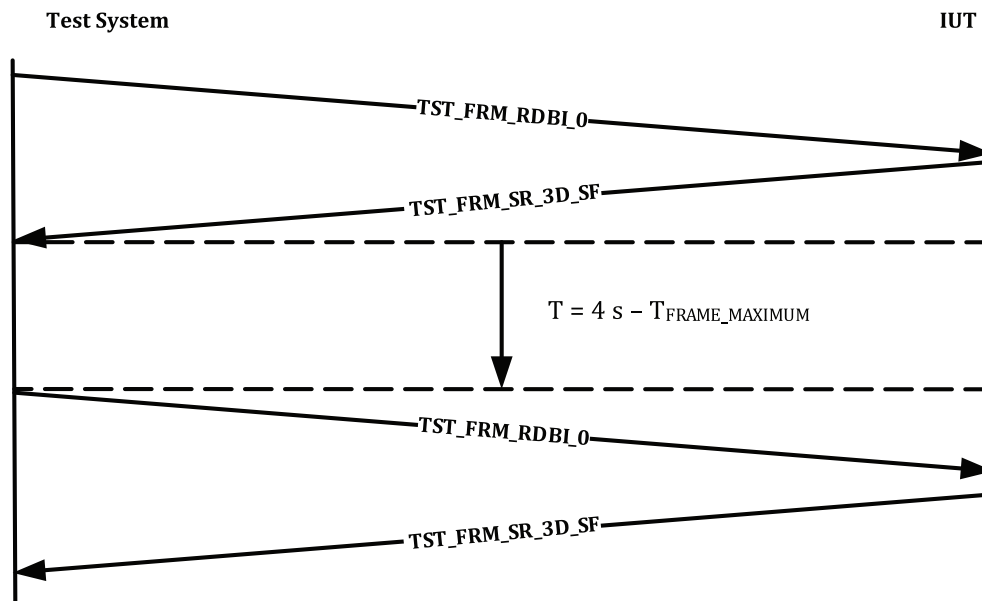
This test verifies that the IUT is able to process messages for the time of at least 4 s bus inactivity.

[Table 84](#) defines the test system for “timeout after bus idle”.

**Table 84 — Test system: Timeout after bus idle**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	The IUT is in operational state.	
<b>Test</b>	The test system sends TST_FRM_RDBI_0 and TST_HDR_SR_3D. After that, there shall be a pause of 4 s - T <sub>FRAME_MAXIMUM</sub> . Then, the test system sends TST_FRM_RDBI_0 and TST_HDR_SR_3D again (see <a href="#">Figure 6</a> ).	
<b>Verification</b>	The IUT shall answer to the request of the test system without entering sleep state.	
<b>Reference</b>	ISO 17987-2:2016, 5.1.4	

[Figure 6](#) shows the timeout after bus idle, IUT as slave.



**Figure 6 — Timeout after bus idle, IUT as slave**

## 14 Frame ID range assignment

### 14.1 [PT-CT 67] Frame ID range assignment with indirect response, IUT as slave

This test verifies if the IUT as slave is able to recognize the start indexes. The test is executed indirectly as no mandatory response is defined. No preconfigured PIDs shall be used.

[Table 85](#) defines the test system for “frame ID range assignment with indirect response”.

**Table 85 — Test system: Frame ID range assignment with indirect response**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE, TST_FRM_IUT_RX, TST_FRM_IUT_TX, TST_FRM_STATUS_SIG <sup>a</sup>
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	<p>The test system sends TST_FRM_ASSIGNIDRANGE where TST_FRM_IUT_TX shall be assigned, the other PIDs shall be filled with don't care values.</p> <p>The test system sends TST_FRM_ASSIGNIDRANGE where TST_FRM_IUT_RX shall be assigned, the other PIDs shall be filled with don't care value.</p> <p>The test system sends the assigned TST_FRM_IUT_TX.</p> <p>The test system sends the assigned TST_FRM_IUT_RX with invalid checksum.</p> <p>After that, TST_FRM_STATUS_SIG shall be sent by the test system.</p>	
<b>Verification</b>	<p>The IUT shall respond to TST_FRM_IUT_TX.</p> <p>The IUT shall respond to TST_FRM_STATUS_SIG with response_error signal = TRUE.</p>	
<b>Reference</b>	ISO 17987-3:2016, 6.3.6.5	
<sup>a</sup> The TST_FRM_STATUS_SIG could be identical to TST_FRM_IUT_TX especially if the IUT has only one unconditional transmit frame.		

## 14.2 [PT-CT 68] Frame ID range unassignment with indirect response, IUT as slave

This test verifies if the IUT as slave is able to unassign PIDs.

[Table 86](#) defines the test system for “frame ID range unassignment with indirect response”.

**Table 86 — Test system: Frame ID range unassignment with indirect response**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGNIDRANGE, TST_FRM_IUT_RX, TST_FRM_IUT_TX, TST_FRM_STATUS_SIG <sup>a</sup>
	bit rate	Default
<b>System Init</b>	TST_FRM_STATUS_SIG shall be sent twice by the test system to ensure that the response_error signal is cleared.	
<b>Test</b>	<p>The test system sends TST_FRM_ASSIGNIDRANGE. One TST_FRM_IUT_TX and one TST_FRM_IUT_RX shall be unassigned.</p> <p>The test system sends the unassigned TST_FRM_IUT_TX.</p> <p>If TST_FRM_IUT_TX is identical to TST_FRM_STATUS_SIG, the test system shall send TST_FRM_ASSIGNIDRANGE where TST_FRM_STATUS_SIG shall be assigned again a PID, the other PIDs shall be filled with don't care values.</p> <p>The test system sends the unassigned TST_FRM_IUT_RX with invalid checksum.</p> <p>After that, TST_FRM_STATUS_SIG shall be sent by the test system.</p>	
<b>Verification</b>	<p>The IUT shall not respond to the first TST_FRM_IUT_TX.</p> <p>The IUT shall respond to TST_FRM_STATUS_SIG with response_error signal = FALSE.</p>	
<b>Reference</b>	ISO 17987-3:2016, 6.3.6.5	
<sup>a</sup> The TST_FRM_STATUS_SIG could be identical to TST_FRM_IUT_TX especially if the IUT has only one unconditional transmit frame.		

## 15 Wildcards

### 15.1 [PT-CT 69] Request with direct response, IUT as slave

This test verifies if the IUT as slave is able to answer to wildcards.

[Table 87](#) defines the test system for “request with direct response”.

**Table 87 — Test system: Request with direct response**

State	Description	
Set Up	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
System Init	Default	
Test	The test system sends TST_FRM_RDBI_0 as defined in <a href="#">Table 88</a> . After that, TST_HDR_SR_3D shall be sent.	
Verification	The IUT shall respond on TST_HDR_SR_3D with the LIN product identification	
Reference	ISO 17987-3:2016, 6.2.3	

[Table 88](#) defines the test cases for “request with direct response”.

**Table 88 — Test cases: Request with direct response**

PT-CT-TC	substitution
[PT-CT 69].1	NAD as wildcard
[PT-CT 69].2	Supplier ID as wildcard
[PT-CT 69].3	Function ID as wildcard
[PT-CT 69].4	Supplier ID and function ID as wildcard

## 16 ReadByIdentifier command

### 16.1 LIN product identification

#### 16.1.1 [PT-CT 70] LIN product identification request with direct response, IUT as slave

This test verifies if the IUT as slave is able to answer to the LIN product identification request.

[Table 89](#) defines the test system for “LIN product identification request with direct response”.

**Table 89 — Test system: LIN product identification request with direct response**

State	Description	
Set Up	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default

**Table 89 (continued)**

State	Description
<b>System Init</b>	Default
<b>Test</b>	The test system sends TST_FRM_RDBI_0 with correct NAD. After that, TST_HDR_SR_3D shall be sent.
<b>Verification</b>	The IUT shall respond on TST_HDR_SR_3D with the LIN product identification
<b>Reference</b>	ISO 17987-2:2016, 11.2.1.2 ISO 17987-3:2016, 6.2 and 6.3.6.6

### 16.1.2 [PT-CT 71] LIN product identification — With interleaved unconditional frame, IUT as slave

This test verifies if the IUT as slave is able to answer to the LIN product identification request with a non-diagnostic frame send between master request diagnostic frame and the slave response diagnostic frame.

[Table 90](#) defines the test system for “LIN product identification request — With interleaved unconditional frame”.

**Table 90 — Test system: LIN product identification request — With interleaved unconditional frame**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_FRM_STATUS_SIG, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_RDBI_0. After that, the non-diagnostic frame TST_FRM_STATUS_SIG is sent by the test system. TST_HDR_SR_3D is sent by the test system.	
<b>Verification</b>	The IUT shall respond on TST_HDR_SR_3D with the LIN product identification	
<b>Reference</b>	ISO 17987-2:2016, 11.2.1.2 ISO 17987-3:2016, 6.2 and 6.3.6.6	

### 16.2 [PT-CT 72] ReadByIdentifier command with correct NAD, IUT as slave

The IUT shall be able to respond to all identifiers.

[Table 91](#) defines the test system for “ReadByIdentifier command with correct NAD”.

**Table 91 — Test system: ReadByIdentifier command with correct NAD**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_X, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	

**Table 91** (continued)

State	Description
<b>Test</b>	The test system sends TST_FRM_RDBI_X followed by TST_HDR_SR_3D. This sequence is repeated with all RDBI service “identifiers”. The request of the test system shall be sent with correct supplier ID and correct function ID. wildcards shall not be used.
<b>Verification</b>	The IUT shall respond positively if the identifier is supported. For all reserved or unsupported ID’s, the IUT shall respond with the negative response. User defined area: If the ID is supported, the positive response shall be checked for correct NAD, valid PCI (range 02 <sub>16</sub> – 06 <sub>16</sub> ) and RSID
<b>Reference</b>	ISO 17987-3:2016, 6.3.6.6

### 16.3 [PT-CT 73] ReadByIdentifier command with incorrect addressing, IUT as slave

The ReadByIdentifier command specifies, that an answer shall only be sent if NAD, PCI.length, supplier ID and function ID match.

[Table 92](#) defines the test system for “ReadByIdentifier command with incorrect addressing”.

**Table 92 — Test system: ReadByIdentifier command with incorrect addressing**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	Wildcards shall not be used. The test system sends TST_FRM_RDBI_0 with parameters according to the IUT specification except the parameter specified in the test case of <a href="#">Table 93</a> . The test system sends TST_HDR_SR_3D.	
<b>Verification</b>	The IUT shall not respond to the request.	
<b>Reference</b>	ISO 17987-3:2016, 6.3.6.6	

[Table 93](#) defines the test cases for “ReadByIdentifier command with incorrect addressing”.

**Table 93 — Test cases: ReadByIdentifier command with incorrect addressing**

PT-CT-TC	ReadByIdentifier
[PT-CT 73].1	Incorrect IUT NAD (but, valid: 1 <sub>16</sub> – 7D <sub>16</sub> )
[PT-CT 73].2	Incorrect supplier ID MSB First bit of incorrect field shall be inverted.
[PT-CT 73].3	Incorrect supplier ID LSB First bit of incorrect field shall be inverted.
[PT-CT 73].4	Incorrect function ID MSB First bit of incorrect field shall be inverted.
[PT-CT 73].5	Incorrect function ID LSB First bit of incorrect field shall be inverted.
[PT-CT 73].6	PCI.length = 05 <sub>16</sub>

## 17 NAD assignment

### 17.1 General

Assign NAD is used to resolve conflicting node addresses. A response shall only be sent if the NAD, the supplier ID and the function ID match.

The test cases in this clause can only be performed if the IUT supports this optional service, which is listed in the NCF.

### 17.2 [PT-CT 74] NAD assignment — Followed by ReadByIdentifier service, IUT as slave

[Table 94](#) defines the test system for “NAD assignment — Followed by ReadByIdentifier service”.

**Table 94 — Test system: NAD assignment — Followed by ReadByIdentifier service**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGN_NAD, TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_ASSIGN_NAD with a valid new configured NAD which shall not be the same value as the initial NAD. The test system sends TST_FRM_RDBI_0 with the new configured NAD followed by TST_HDR_SR_3D. The test system sends TST_FRM_RDBI_0 with initial NAD followed by TST_HDR_SR_3D.	
<b>Verification</b>	The IUT shall answer to the first RDBI request with the newly assigned configured NAD. The IUT shall not answer to the second RDBI request with the initial NAD.	
<b>Reference</b>	ISO 17987-3:2016, 6.3.6 and 6.3.6.1	

### 17.3 [PT-CT 75] NAD assignment — With positive response, IUT as slave

[Table 95](#) defines the test system for “NAD assignment — With positive response”.

**Table 95 — Test system: NAD assignment — With positive response**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGN_NAD, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_ASSIGN_NAD. After that, TST_HDR_SR_3D is sent by the test system.	
<b>Verification</b>	The IUT shall respond with the positive assign NAD response and its initial NAD.	
<b>Reference</b>	ISO 17987-3:2016, 6.3.6 and 6.3.6.1	

### 17.4 [PT-CT 76] NAD assignment — Initial NAD, IUT as slave

This test case shall detect the error in an IUT as slave, when the IUT does not use the initial NAD (in the first data) for the Assign NAD service.

[Table 96](#) defines the test system for “NAD assignment — Initial NAD, IUT as slave”.

**Table 96 — Test system: NAD assignment — Initial NAD, IUT as slave**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	slave nodes supporting AssignNAD
	Configuration	TST_FRM_ASSIGN_NAD, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	<p>The test system sends TST_FRM_ASSIGN_NAD with new configured NAD = initial NAD for initialization. The test system sends TST_HDR_SR_3D.</p> <p>The test system sends TST_FRM_ASSIGN_NAD. The new assigned configured NAD shall not be the initial NAD. The test system sends TST_HDR_SR_3D.</p> <p>The test system sends TST_FRM_ASSIGN_NAD. The new assigned configured NAD shall not be the initial NAD and differ from the previous assignment. The test system sends TST_HDR_SR_3D.</p>	
<b>Verification</b>	The IUT shall respond with the positive assign NAD response in all three cases.	
<b>Reference</b>	ISO 17987-3:2016, 6.3.6 and 6.3.6.1	

## 18 Save Configuration

### 18.1 General

When supporting the Save Configuration command, a slave node shall store the current configured NAD and PIDs to NVM when the Save Configuration command is received. After a power on cycle, the NVM NAD and PID configuration shall be used by the slave node instantly.

### 18.2 [PT-CT 77] Save Configuration — With positive response, IUT as slave

[Table 97](#) defines the test system for “Save Configuration — With positive response, IUT as slave”.

**Table 97 — Test system: Save Configuration — With positive response, IUT as slave**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_SAVE_CONFIG, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_SAVE_CONFIG followed by TST_HDR_SR_3D.	
<b>Verification</b>	The IUT shall respond with the positive Save Configuration response.	
<b>Reference</b>	ISO 17987-3:2016, 6.3.6.4	

### 18.3 [PT-CT 78] Save Configuration — Save a new NAD, IUT as slave

[Table 98](#) defines the test system for “Save Configuration – save a new NAD, IUT as slave”.



**Table 98 — Test system: Save Configuration — Save a new NAD, IUT as slave**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_ASSIGN_NAD, TST_FRM_RDBI_0, TST_FRM_SAVE_CONFIG, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	<p>The test system sends TST_FRM_ASSIGN_NAD command with a NAD value different to the initially configured NAD. The test system sends TST_FRM_RDBI_0 command with new configured NAD followed by TST_HDR_SR_3D.</p> <p>The test system sends TST_FRM_SAVE_CONFIG followed by TST_HDR_SR_3D. After 10 s a power cycle is performed.</p> <p>After a wake up sequence, the test system sends TST_FRM_RDBI_0 command with new configured NAD followed by TST_HDR_SR_3D.</p>	
<b>Verification</b>	The IUT shall accept the new configured NAD and send a positive response to all three TST_HDR_SR_3D.	
<b>Reference</b>	ISO 17987-3:2016, 6.3.6.4	

#### 18.4 [PT-CT 79] Save Configuration — Save new frame identifiers, IUT as slave

[Table 99](#) defines the test system for “Save Configuration — Save new frame identifiers, IUT as slave”.

**Table 99 — Test system: Save Configuration — Save new frame identifiers, IUT as slave**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_IUT_TX, TST_FRM_ASSIGNIDRANGE, TST_FRM_SAVE_CONFIG, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	<p>The test system sends the header of TST_FRM_IUT_TX.</p> <p>The test system changes the PID of the frame sent before using the TST_FRM_ASSIGNIDRANGE followed by TST_HDR_SR_3D.</p> <p>The test system sends TST_FRM_SAVE_CONFIG followed by TST_HDR_SR_3D. After 10 s a power cycle is performed.</p> <p>After a wake up sequence, the test system sends a header of TST_FRM_IUT_TX with the initially used PID.</p> <p>The test system sends a header with the newly assigned PID.</p>	
<b>Verification</b>	<p>The IUT shall responde to TST_FRM_IUT_TX.</p> <p>The IUT shall accept the new PID and send a positive response to the TST_FRM_ASSIGNIDRANGE request.</p> <p>The IUT shall accept the TST_FRM_SAVE_CONFIG request and send a positive response.</p> <p>After the power-on reset, the IUT shall ignore the header with the initial PID and respond to the new assigned PID only.</p>	
<b>Reference</b>	ISO 17987-3:2016, 6.3.6.4	

## 19 Transport protocol

### 19.1 [PT-CT 80] Transport layer functional request, IUT as slave

This test shall ensure that a slave ignores a functional request when processing a physical request already.

[Table 100](#) defines the test system for “transport layer functional request”.

**Table 100 — Test system: Transport layer functional request**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_FRM_REQ_FUNC_7E, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_RDBI_0. Then the test system sends TST_FRM_REQ_FUNC_7E. After that, TST_HDR_SR_3D is sent by the test system.	
<b>Verification</b>	The IUT shall respond to TST_HDR_SR_3D with a positive response according to TST_FRM_RDBI_0 and shall ignore the functional addressed request.	
<b>Reference</b>	ISO 17987-2:2016, 9.7.3 and ISO 17987-3:2016, 6.3.4.2	

### 19.2 [PT-CT 81] Abort diagnostic communication with new diagnostic request, IUT as slave

This test shall ensure that a slave is able abort a request if another slave is addressed before the response is sent back.

[Table 101](#) defines the test system for “abort diagnostic communication with new diagnostic request”.

**Table 101 — Test system: Abort diagnostic communication with new diagnostic request**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D, TST_FRM_RDBI_0_NAD
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_RDBI_0. After that, TST_HDR_SR_3D is sent by the test system. The test system sends TST_FRM_RDBI_0, then TST_FRM_RDBI_0_NAD. After that, TST_HDR_SR_3D is sent by the test system.	
<b>Verification</b>	The IUT shall send a positive response to the first request. The IUT shall not respond to the second TST_HDR_SR_3D.	
<b>Reference</b>	ISO 17987-2:2016, 8.4.2 ISO 17987-3:2016, 6.3.5	

### 19.3 [PT-CT 82] IUT receives a segmented request as specified, IUT as slave

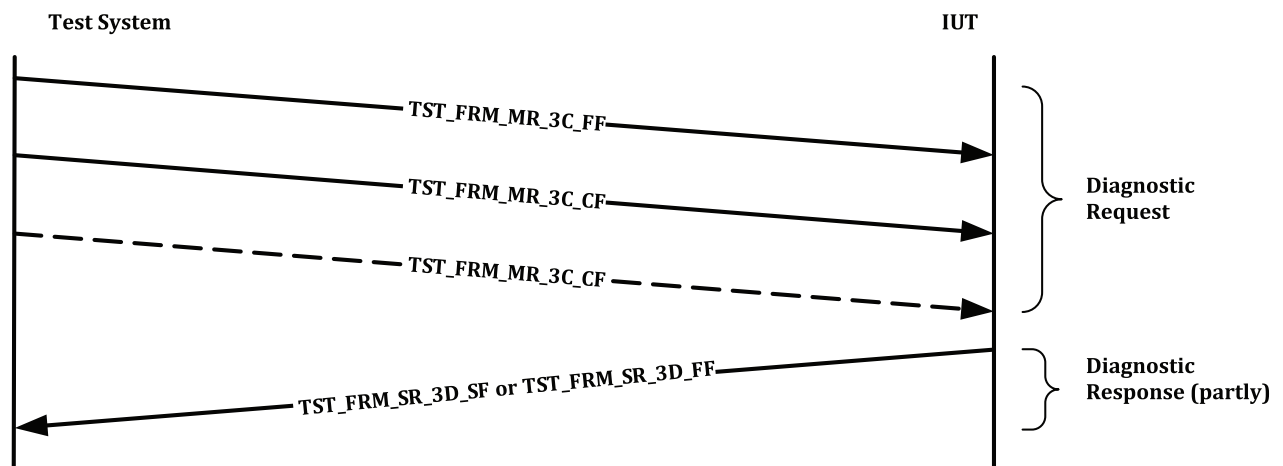
This test case is applicable for diagnostic classes II and III slave nodes when a segmented request is defined in the IUT specification. This test shall ensure that a slave is able to receive segmented requests.

Table 102 defines the test system for “IUT receives a segmented request as specified”.

**Table 102 — Test system: IUT receives a segmented request as specified**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_MR_3C_FF, TST_FRM_MR_3C_CF, TST_FRM_SR_3D_SF or TST_FRM_SR_3D_FF
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends a segmented request which shall be answered by a (segmented) response.	
<b>Verification</b>	Then the IUT sends the (first segment of the) response according to the received request before (see Figure 7). The IUT shall send a correct answer to the master request.	
<b>Reference</b>	ISO 17987-2:2016, 7.3	

Figure 7 shows the IUT receives a segmented request as specified, IUT as slave.



**Figure 7 — IUT receives a segmented request as specified, IUT as slave**

#### 19.4 [PT-CT 83] IUT receives a segmented request interleaved with unconditional frame, IUT as slave

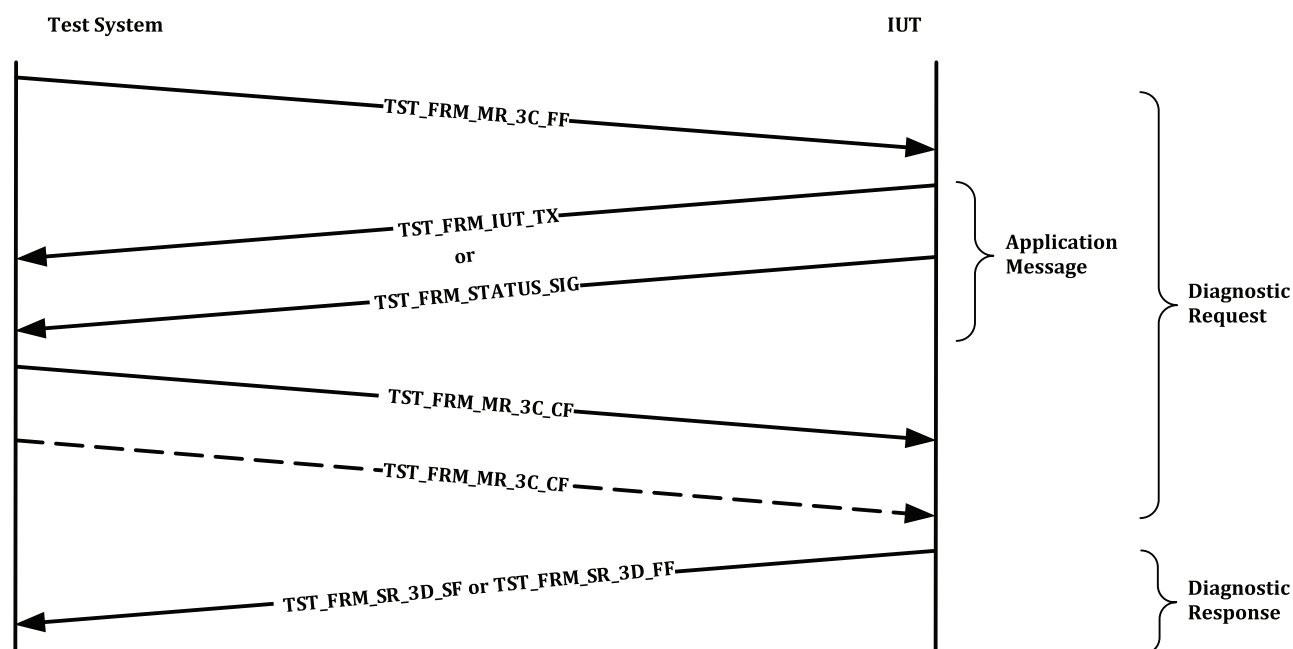
This test case is applicable for diagnostic classes II and III slave nodes when a segmented request is defined in the IUT specification. This test shall ensure that the slave is able to receive segmented requests with interleaved unconditional frame.

Table 103 defines the test system for “IUT receives a segmented request with interleaved unconditional frame”.

**Table 103 — Test system: IUT receives a segmented request with interleaved unconditional frames**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_MR_3C_FF, TST_FRM_MR_3C_CF, TST_FRM_IUT_TX, TST_FRM_STATUS_SIG, TST_FRM_SR_3D_SF or TST_FRM_SR_3D_FF
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT shall send the (first segment of the) response according to the preceding request (see <a href="#">Figure 8</a> ).	
<b>Verification</b>	The IUT shall send a correct answer to the master request.	
<b>Reference</b>	ISO 17987-2:2016, 9.6.2	

[Figure 8](#) shows the IUT receives a segmented request with interleaved unconditional frames, IUT as slave.



**Figure 8 — IUT receives a segmented request with interleaved unconditional frame, IUT as slave**

### 19.5 [PT-CT 84] IUT receives a segmented request with interleaved functional request, IUT as slave

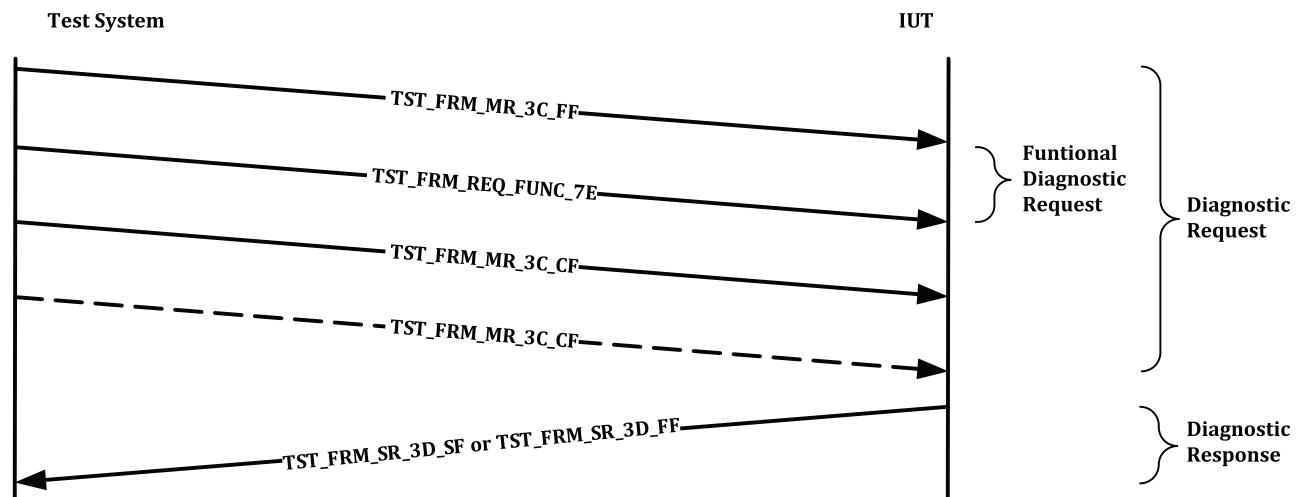
This test case is applicable for diagnostic classes II and III slave nodes when a segmented request is defined in the IUT specification. This test shall ensure that a slave is able to receive segmented request with interleaved functional request.

[Table 104](#) defines the test system for “IUT receives a segmented request request with interleaved functional request”.

**Table 104 — Test system: IUT receives a segmented request with interleaved functional request**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_MR_3C_FF, TST_FRM_MR_3C_CF, TST_FRM_REQ_FUNC_7E, TST_FRM_SR_3D_SF or TST_FRM_SR_3D_FF
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends a segmented request which is interleaved by TST_FRM_REQ_FUNC_7E.	
<b>Verification</b>	The IUT shall send the (first segment of the) response according to the request before (see <a href="#">Figure 9</a> ). The IUT shall send a correct answer to the master request and shall ignore the functional addressed request.	
<b>Reference</b>	ISO 17987-2:2016, 9.7.3	

[Figure 9](#) shows a slave receiving a segmented request with interleaved functional request.



**Figure 9 — Slave receiving a segmented request with interleaved functional request**

## 19.6 IUT shall ignore request after timeout

### 19.6.1 [PT-CT 85] IUT shall ignore segmented requests on $N_{Cr_{Max}}$ timeout, IUT as slave

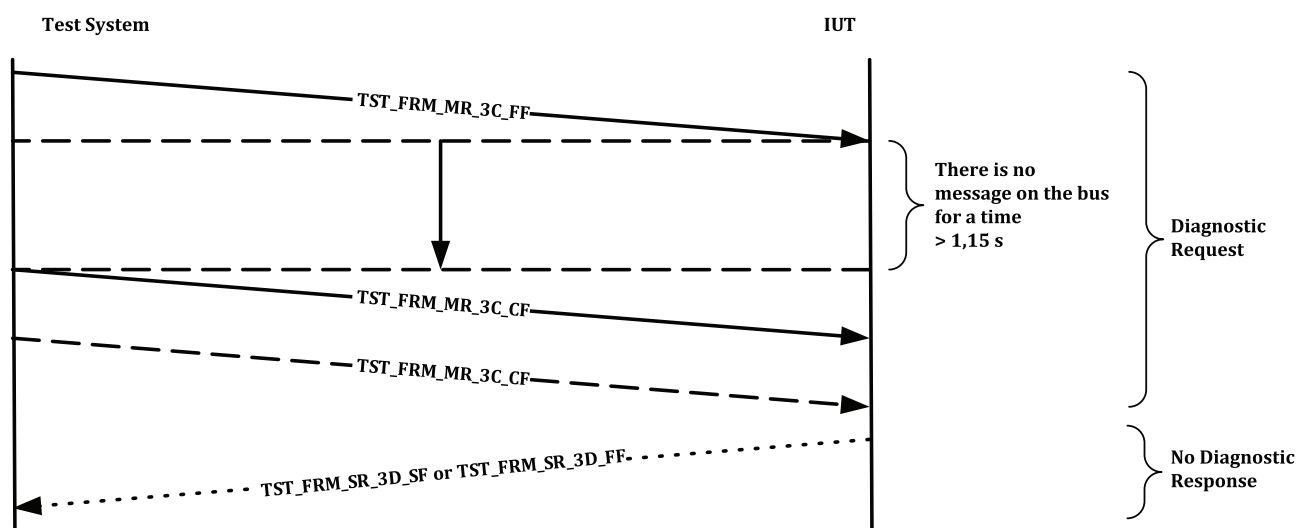
This test case is applicable for diagnostic classes II and III slave nodes when a segmented request is defined in the IUT specification. This test shall ensure that a slave ignores a master request if a  $N_{Cr_{Max}}$  timeout occurs. It shall be tested that if the time between two segmented messages of a master request is  $>1,15$  s, the IUT shall not answer.

[Table 105](#) defines the test system for “IUT shall ignore segmented requests on  $N_{Cr_{Max}}$  timeout”.

**Table 105 — Test system: IUT shall ignore segmented requests on N\_Cr<sub>Max</sub> timeout**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_MR_3C_FF, TST_FRM_MR_3C_CF, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends a segmented request which provokes a (segmented) response. After the TST_FRM_MR_3C_FF, there is no message on the bus for a time >1,15 s before the request is finished by the test system (see <a href="#">Figure 10</a> ). Then the test systems TST_HDR_SR_3D.	
<b>Verification</b>	The IUT shall not answer to the master request.	
<b>Reference</b>	ISO 17987-2:2016, 7.6.3	

[Figure 10](#) shows the ignoring of segmented requests on N\_Cr<sub>Max</sub> timeout, IUT as slave.



**Figure 10 — Ignore segmented requests on N\_Cr<sub>Max</sub> timeout, IUT as slave**

### 19.6.2 [PT-CT 86] IUT shall observe transport layer N\_As<sub>Max</sub> timeout, IUT as slave

This test shall ensure that a slave rejects a slave response if a N\_As<sub>Max</sub> timeout occurs. It shall be tested that if the time between two transport layer messages is >1,15 s<sup>1)</sup> and <4 s the IUT shall not answer. Only the 1,15 s shall be tested.

[Table 106](#) defines the test system for “IUT shall observe transport layer timeout”.

**Table 106 — Test system: IUT shall observe transport layer N\_As<sub>Max</sub> timeout**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default

1) The time measurement results of N\_As<sub>Max</sub> = max 1 s + slaves tolerance 14 % ≥ (approx. 1,14 s).

Table 106 (continued)

State	Description
System Init	Default
Test	The IUT receives a master request which the slave shall respond to. There shall be no communication on the bus for 1,15 s (see Figure 11). Then, the TST_HDR_SR_3D is sent.
Verification	The IUT shall not answer to the TST_FRM_RDBI_0 request.
Reference	ISO 17987-2:2016, 7.6.3

Figure 11 shows the IUT shall observe transport layer  $N_{AsMax}$  timeout, IUT as slave.

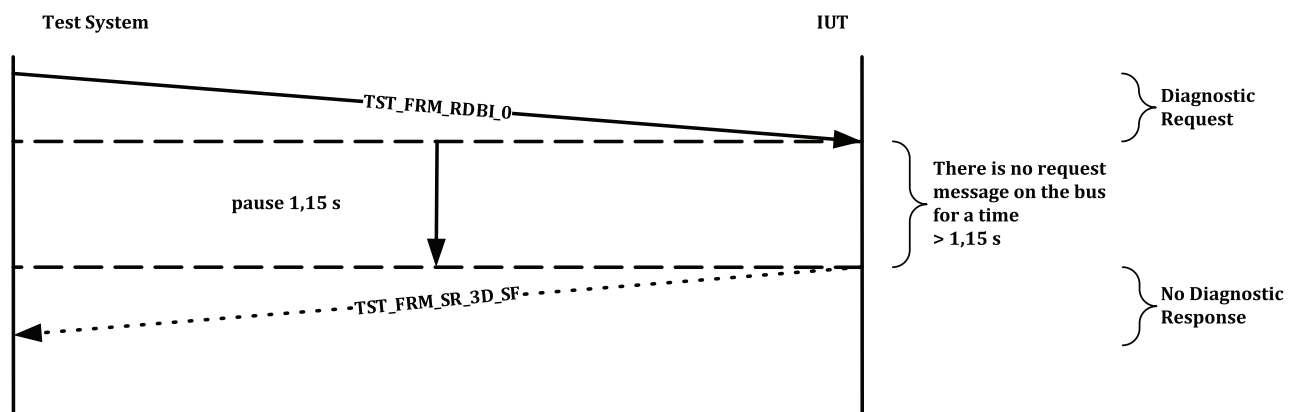


Figure 11 — IUT shall observe transport  $N_{AsMax}$  layer timeout, IUT as slave

### 19.7 [PT-CT 87] IUT shall ignore segmented requests with wrong sequence numbering, IUT as slave

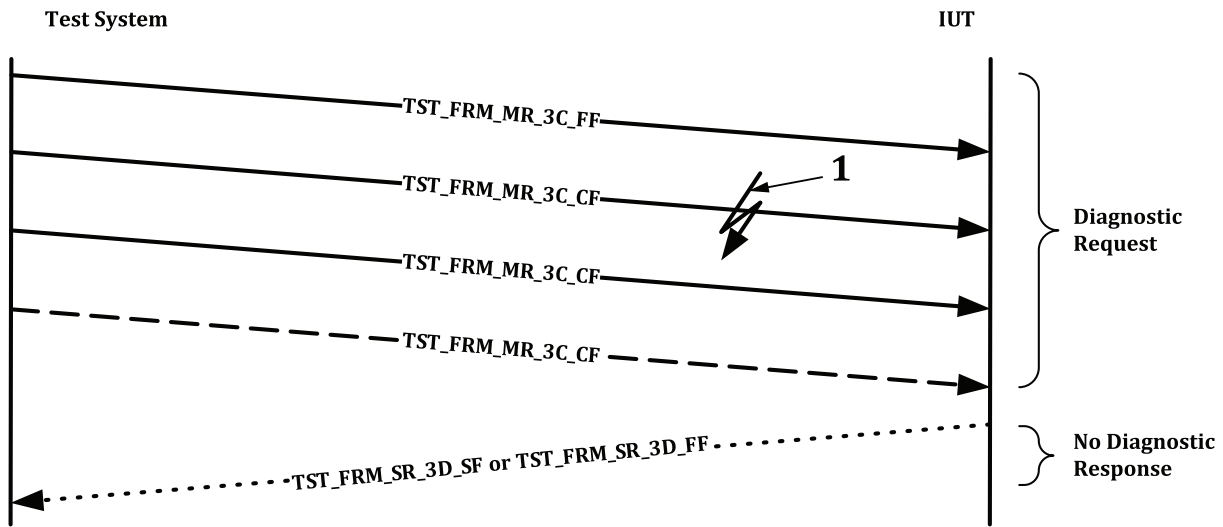
This test case is applicable for diagnostic classes II and III slave nodes when a segmented request is defined in the IUT specification. This test shall ensure that a slave ignores segmented requests with wrong sequence numbering.

Table 107 defines the test system for “IUT shall ignore segmented requests with wrong sequence numbering”.

Table 107 — Test system: IUT shall ignore segmented requests with wrong sequence numbering

State	Description	
Set Up	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_MR_3C_FF, TST_FRM_MR_3C_CF, TST_HDR_SR_3D
	bit rate	Default
System Init	Default	
Test	The IUT receives a segmented request which shall be answered by a (segmented) response. The second segment includes a wrong PCI, $22_{16}$ instead of $21_{16}$ . The other segments are sent with a correct PCI (see Figure 12).	
Verification	The IUT shall not answer to the master request.	
Reference	ISO 17987-2:2016, 7.6.3	

Figure 12 shows: IUT shall ignore segmented requests with wrong sequence numbering, IUT as slave.



**Key**

1 second segment includes a wrong PCI

**Figure 12 — Shows: IUT shall ignore segmented requests with wrong sequence numbering, IUT as slave**

**19.8 [PT-CT 88] IUT shall respond with correct segmented response, IUT as slave**

This test case is applicable for diagnostic classes II and III slave nodes when a segmented response is defined in the IUT specification. This test shall ensure that the IUT responds with correct length, PCI and sequence number. The data content is not in the focus of this test

Table 108 defines the test system for “IUT shall respond with correct segmented response”.

**Table 108 — Test system: IUT shall respond with correct segmented response**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_MR_3C_SF or TST_FRM_MR_3C_FF and TST_FRM_MR_3C_CF, TST_FRM_SR_3D_FF, TST_FRM_SR_3D_CF
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT receives a (segmented) request which shall be answered by a segmented response (see Figure 13).	
<b>Verification</b>	The IUT shall answer to the master request with a segmented response with correct length, PCI and sequence number. It is allowed that the IUT sends a response pending before.	
<b>Reference</b>	ISO 17987-2:2016, 7.3	

Figure 13 shows: IUT shall respond with correct segmented response, IUT as slave.



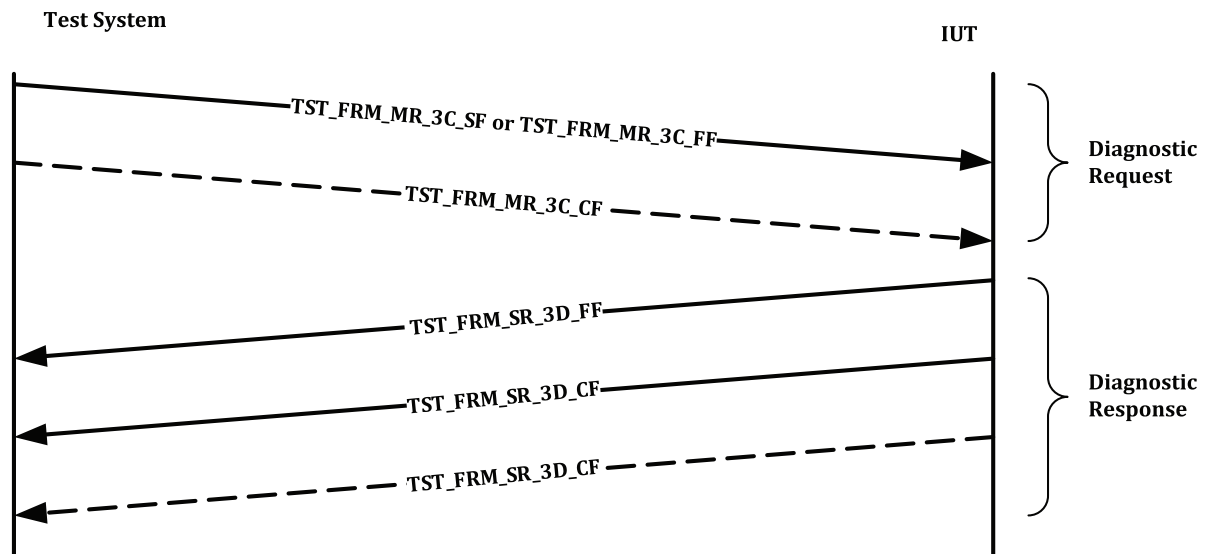


Figure 13 — IUT shall respond with correct segmented response, IUT as slave

## 19.9 IUT sends a segmented response with interleaved unconditional frames

### 19.9.1 [PT-CT 89] IUT sends a segmented response with interleaved unconditional frame, IUT as slave

This test case is applicable for diagnostic classes II and III slave nodes when a segmented request is defined in the IUT specification. This test shall ensure that a slave is able to send a segmented response with interleaved unconditional frame.

Table 109 defines the test system for “IUT sends a segmented response with interleaved unconditional frame”.

Table 109 — Test system: IUT sends a segmented response with interleaved unconditional frame

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_MR_3C_SF or TST_FRM_MR_3C_FF and TST_FRM_MR_3C_CF, TST_FRM_IUT_TX or TST_FRM_STATUS_SIG, TST_FRM_SR_3D_FF, TST_FRM_SR_3D_CF
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends a (segmented) request which shall be answered by a segmented response. The test system interleaves the IUT response with TST_FRM_IUT_TX (or TST_FRM_STATUS_SIG).	
<b>Verification</b>	The IUT shall send the response according to the request before (see Figure 14). The IUT shall send a correct answer to the request.	
<b>Reference</b>	ISO 17987-2:2016, 9.6.3	

Figure 14 shows IUT sends a segmented response with interleaved unconditional frame, IUT as slave.

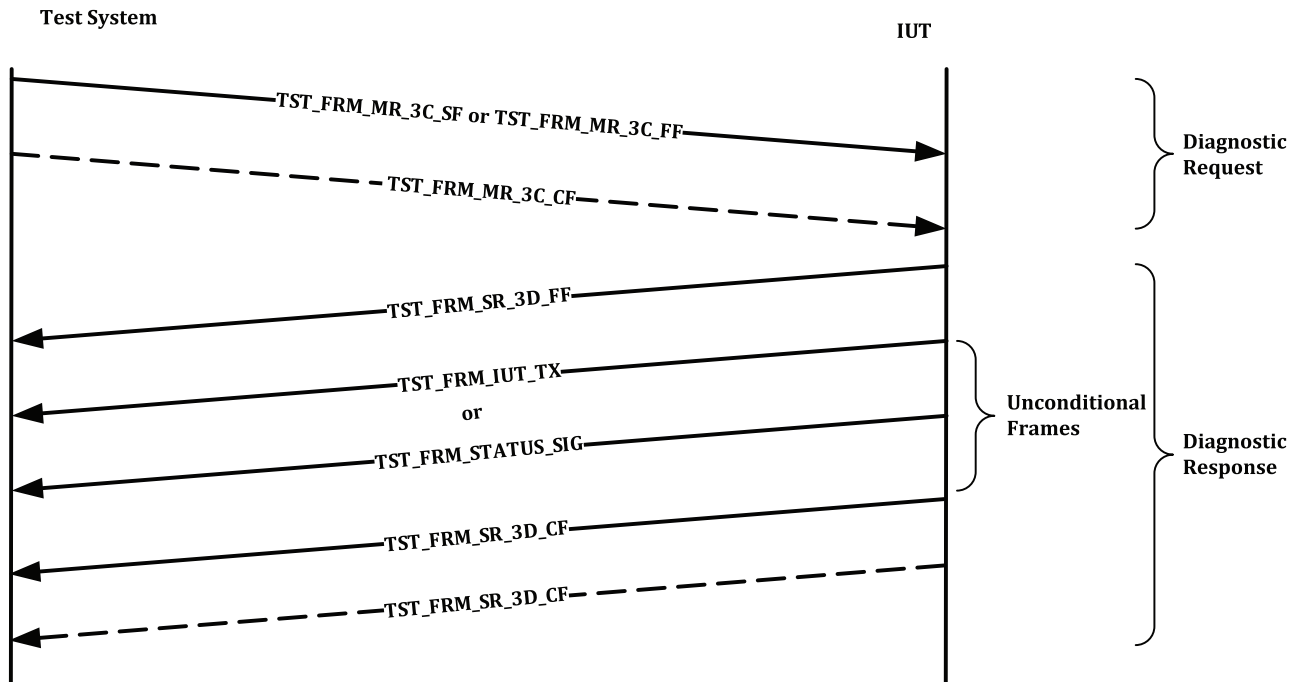


Figure 14 — IUT sends a segmented response with interleaved unconditional frame, IUT as slave

19.9.2 [PT-CT 90] IUT sends a segmented response with interleaved functional request, IUT as slave

This test case is applicable for diagnostic classes II and III slave nodes when a segmented response is defined in the IUT specification. This test shall ensure that the slave is able to send a segmented response with interleaved functional request.

Table 110 defines the test system for “IUT sends a segmented response with interleaved functional request”.

Table 110 — Test system: IUT sends a segmented response with interleaved functional request

State	Description	
Set Up	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_MR_3C_SF or TST_FRM_MR_3C_FF and TST_FRM_MR_3C_CF, TST_FRM_REQ_FUNC_7E TST_FRM_SR_3D_FF, TST_FRM_SR_3D_CF
	bit rate	Default
System Init	Default	
Test	The test system sends a request which shall be answered by a segmented response. The test system interleaves the response with a functional request.	
Verification	The IUT sends the response according to the request before (see Figure 15). The IUT shall send a correct answer to the request and shall ignore the functional addressed request.	
Reference	ISO 17987-2:2016, 9.7.3	

Figure 15 shows IUT sends a segmented response with interleaved functional request, IUT as slave.

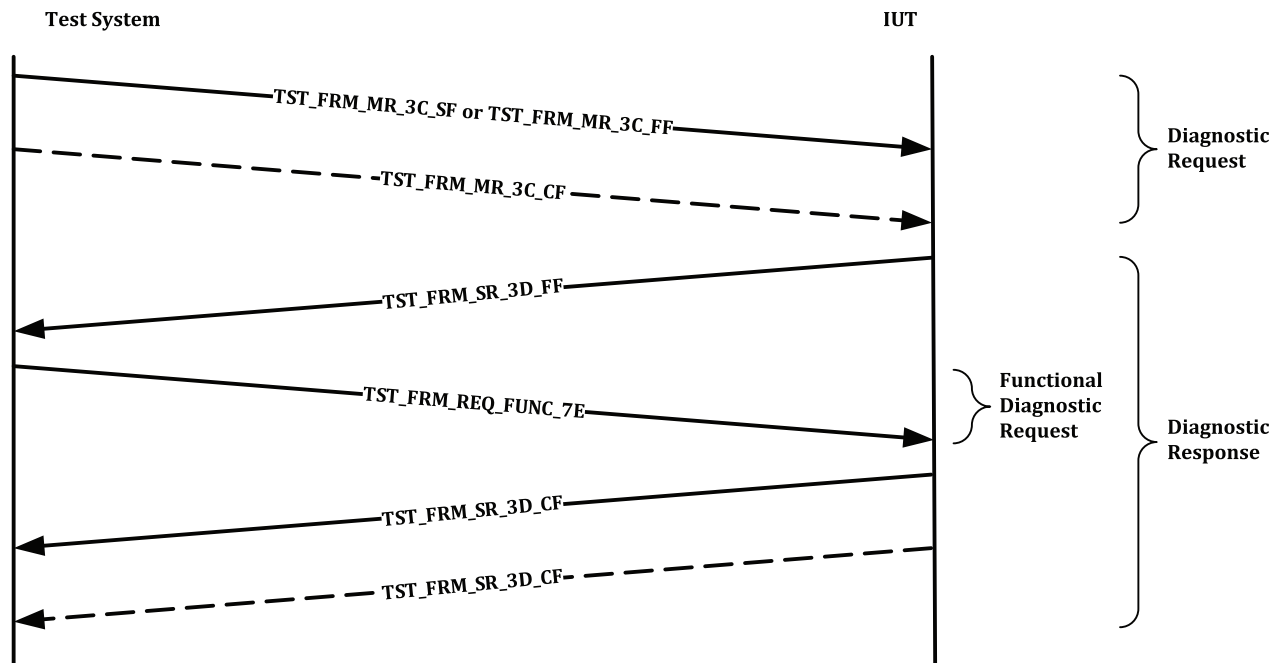


Figure 15 — IUT sends a segmented response with interleaved functional request, IUT as slave

**19.10 [PT-CT 91] IUT shall not respond to slave response header if there is no request before, IUT as slave**

This test shall ensure that the IUT responds only if a master request is sent before.

Table 111 defines the test system for “IUT shall not respond to slave response header if there is no request before”.

**Table 111 — Test system: IUT shall not respond to slave response header if there is no request before**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The IUT shall go awake. After a time >1,15 s (= transport layer timeout) and <4 s TST_HDR_SR_3D is sent by the test system.	
<b>Verification</b>	The IUT shall not answer.	
<b>Reference</b>	ISO 17987-2:2016, 9.7.3 (transition key 9)	

**19.11 [PT-CT 92] IUT shall not respond to slave response header if the response is already sent, IUT as slave**

This test shall ensure that the IUT responds only if a master request is sent before.

Table 112 defines the test system for “IUT shall not respond to slave response header if the response is already sent”.

**Table 112 — Test system: IUT shall not respond to slave response header if the response is already sent**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_RDBI_0, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	The test system sends TST_FRM_RDBI_0 and then two times TST_HDR_SR_3D.	
<b>Verification</b>	The IUT shall respond to the first TST_HDR_SR_3D. The second TST_HDR_SR_3D shall not be answered by the IUT.	
<b>Reference</b>	ISO 17987-2:2016, 9.7.3 (transition key 6)	

### 19.12 [PT-CT 93] IUT shall abort segmented response on N\_Cs<sub>Max</sub> timeout, IUT as slave

This test case is applicable for diagnostic classes II and III slave nodes when a segmented response is defined in the IUT specification. This test shall ensure that a slave aborts the segmented response if a N\_Cs<sub>Max</sub> timeout occurs.

[Table 113](#) defines the test system for “IUT shall abort a segmented response on timeout”.

**Table 113 — Test system: IUT shall abort a segmented response on N\_Cs<sub>Max</sub> timeout**

State	Description	
<b>Set Up</b>	IUT as	Slave
	Classification	Class B, class C devices
	Configuration	TST_FRM_MR_3C_SF or TST_FRM_MR_3C_FF and TST_FRM_MR_3C_CF, TST_FRM_SR_3D_FF, TST_HDR_SR_3D
	bit rate	Default
<b>System Init</b>	Default	
<b>Test</b>	<p>The test system sends a (segmented) request which shall be answered by a segmented response.</p> <p>After the request, the test system starts to repeat transmitting TST_HDR_SR_3Ds until the IUT answers the request with TST_FRM_SR_3D_FF.</p> <p>The test system stops sending TST_HDR_SR_3D on the bus for a time &gt;1,15s (see <a href="#">Figure 16</a>) before it continues again.</p>	
<b>Verification</b>	The IUT shall abort the response transmission after N_Cs <sub>Max</sub> timeout. TST_FRM_SR_3D_CF shall not be sent by the IUT anymore.	
<b>Reference</b>	ISO 17987-2:2016, 7.6.3	

[Figure 16](#) shows IUT shall abort a segmented response on N\_Cs<sub>Max</sub> timeout, IUT as slave.

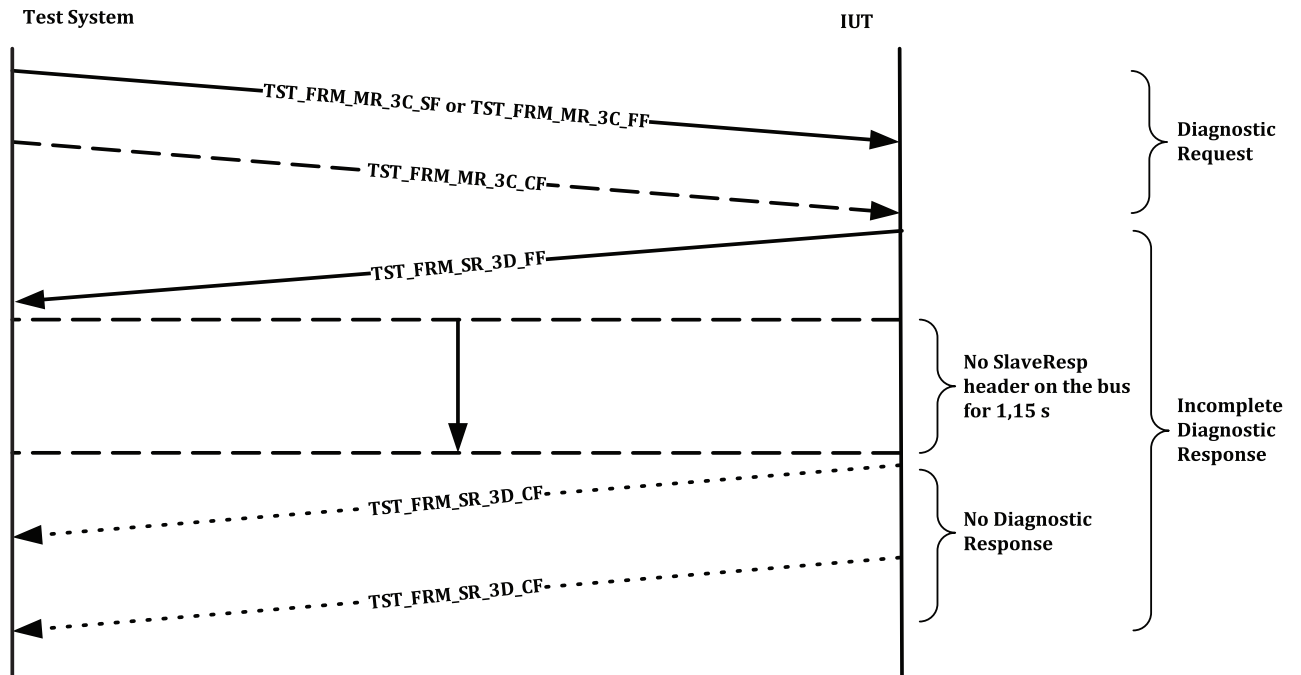


Figure 16 — IUT shall abort a segmented response on  $N_{CsMax}$  timeout, IUT as slave

## Bibliography

- [1] ISO/IEC 7498-1, *Information technology — Open Systems Interconnection — Basic Reference Model: The Basic Model*
- [2] ISO 14229-1, *Road vehicles — Unified diagnostic services (UDS) — Part 1: Specification and requirements*
- [3] ISO 14229-2, *Road vehicles — Unified diagnostic services (UDS) — Part 2: Session layer services*
- [4] ISO 14229-7, *Road vehicles — Unified diagnostic services (UDS) — Part 7: UDS on local interconnect network (UDSonLIN)*
- [5] ISO/IEC 10731, *Information technology — Open Systems Interconnection — Basic Reference Model — Conventions for the definition of OSI services*
- [6] ISO 17987-5:2016, *Road vehicles — Local Interconnect Network (LIN) — Part 5: Application programmers interface (API)*
- [7] SAE J2602-1:2012, *LIN Network for Vehicle Applications*









# British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

## About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards-based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

## Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at [bsigroup.com/standards](http://bsigroup.com/standards) or contacting our Customer Services team or Knowledge Centre.

## Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at [bsigroup.com/shop](http://bsigroup.com/shop), where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

## Copyright in BSI publications

All the content in BSI publications, including British Standards, is the property of and copyrighted by BSI or some person or entity that owns copyright in the information used (such as the international standardization bodies) and has formally licensed such information to BSI for commercial publication and use.

Save for the provisions below, you may not transfer, share or disseminate any portion of the standard to any other person. You may not adapt, distribute, commercially exploit, or publicly display the standard or any portion thereof in any manner whatsoever without BSI's prior written consent.

## Storing and using standards

Standards purchased in soft copy format:

- A British Standard purchased in soft copy format is licensed to a sole named user for personal or internal company use only.
- The standard may be stored on more than 1 device provided that it is accessible by the sole named user only and that only 1 copy is accessed at any one time.
- A single paper copy may be printed for personal or internal company use only.

Standards purchased in hard copy format:

- A British Standard purchased in hard copy format is for personal or internal company use only.
- It may not be further reproduced – in any format – to create an additional copy. This includes scanning of the document.

If you need more than 1 copy of the document, or if you wish to share the document on an internal network, you can save money by choosing a subscription product (see 'Subscriptions').

## Reproducing extracts

For permission to reproduce content from BSI publications contact the BSI Copyright & Licensing team.

## Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to [bsigroup.com/subscriptions](http://bsigroup.com/subscriptions).

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

**PLUS** is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit [bsigroup.com/shop](http://bsigroup.com/shop).

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email [subscriptions@bsigroup.com](mailto:subscriptions@bsigroup.com).

## Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

## Useful Contacts

### Customer Services

**Tel:** +44 345 086 9001

**Email (orders):** [orders@bsigroup.com](mailto:orders@bsigroup.com)

**Email (enquiries):** [cservices@bsigroup.com](mailto:cservices@bsigroup.com)

### Subscriptions

**Tel:** +44 345 086 9001

**Email:** [subscriptions@bsigroup.com](mailto:subscriptions@bsigroup.com)

### Knowledge Centre

**Tel:** +44 20 8996 7004

**Email:** [knowledgecentre@bsigroup.com](mailto:knowledgecentre@bsigroup.com)

### Copyright & Licensing

**Tel:** +44 20 8996 7070

**Email:** [copyright@bsigroup.com](mailto:copyright@bsigroup.com)

### BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK