

# C&S Conformance Test

## CAN – Data Link Layer

### Specification C&S enhancement tests

<b>Author(s) and Organization(s):</b>	Andreas Meitrodt C&S group GmbH
<b>Reviewed by:</b>	Christoph Wosnitza
<b>Revision:</b>	05
<b>Date:</b>	2019-02-19
<b>Status:</b>	Released
<b>Working document:</b>	-
<b>Document Scope:</b> (Short description only)	Description of C&S enhancement tests to ISO16845-1:2016

## Revision history

Date	Version	changes	Author
04.04.2017	03d00	Initial Version	LT
05.04.2017	03d01	Add "CC" Test BT Infos and C&S BT add-on tests	AM
07.04.2017	03d02	Add "FD" Test BT Infos	AM
02.05.2017	03d03	Add comments to TC 7.6.1 Mark all _CS add-on chapters	AM
12.05.2017	03d04	Reformat cases	LT
06.06.2017	03d05	Add C&S Test FD_FD 2.1.13	AM
12.06.2017	03d06	Reformat cases	LT
21.08.2017	03d07	Add info about number of C&S add-on tests in test groups	AM
23.08.2017	03	Reviewed	FF
29.09.2017	04d00	Add ISO11898-1 chapters to Test 7.6 To explain error frame position after CRC Error	AM
23.01.2018	04d01	Add comments about TC (8)3.5.11 Bus-off recovery	AM
03.09.2018	04d01	Extract of the C&S enhancement tests	CW
19.02.2019	05	Add comment about tested configuration range for bit timing testes in chapter 1.3.7	AM

# Table of Content

1	Organisation and methods .....	9
1.1	References and Conventions .....	9
1.1.1	References .....	9
1.1.2	Conventions .....	10
1.2	Global overview .....	14
1.2.1	Scope of test plan .....	14
1.2.2	Architecture of the Test Plan .....	14
1.3	Organisation .....	16
1.3.1	General organisation .....	16
1.3.2	Test Case organisation .....	16
1.3.3	Hierarchical structure of tests .....	17
1.3.4	Overview .....	17
1.3.5	Test Types .....	17
1.3.6	Test Classes .....	18
1.3.7	Test Cases .....	18
1.4	LT parameters .....	20
1.4.1	Overview .....	20
1.4.2	Description of parameters .....	20
2	Test type 1, received frame .....	23
2.1	Test class 1, valid frame format .....	23
2.1.1	Identifier and number of data test in base format .....	23
2.1.2	Identifier and number of data test in extended format .....	23
2.1.3	Reception after arbitration lost .....	23
2.1.4	Acceptance of non-nominal bit in base format frame .....	23
2.1.5	Acceptance of non-nominal bit in extended format frame .....	24
2.1.6	Protocol exception behaviour on non-nominal bit .....	24
2.1.7	Minimum time for bus idle after protocol exception handling .....	24
2.1.8	DLC greater than 8 .....	24
2.1.9	Absent bus idle — Valid frame reception .....	24
2.1.10	Stuff acceptance test in base format frame .....	24
2.1.11	Stuff acceptance test in extended format frame .....	25
2.1.12	Message validation .....	25
2.1.13	C&S, Form field tolerance test for FD frame format .....	25
2.2	Test class 2, error detection .....	27
2.2.1	Bit error in data frame .....	27
2.2.2	Stuff error for basic frame .....	27
2.2.3	Stuff error for extended frame .....	27
2.2.4	Stuff error for FD frame payload bytes .....	27

2.2.5	CRC error .....	27
2.2.6	Combination of CRC error and form error .....	28
2.2.7	Form error in data frame at “CRC delimiter” bit position .....	28
2.2.8	Form error at fixed stuff bit in FD frames .....	28
2.2.9	Form error in data frame at “ACK delimiter” bit position .....	28
2.2.10	Form error in data frame at “EOF” .....	28
2.2.11	Message non-validation .....	28
2.3	Test class 3, error frame management .....	29
2.3.1	Error flag longer than 6 bits .....	29
2.3.2	Data frame starting on the third bit of intermission field .....	29
2.3.3	Bit error in error flag .....	29
2.3.4	Form error in error delimiter .....	29
2.4	Test class 4, overload frame management .....	30
2.4.1	MAC overload generation during intermission field .....	30
2.4.2	Last bit of EOF .....	30
2.4.3	Eighth bit of an error and overload delimiter .....	30
2.4.4	Bit error in overload flag .....	30
2.4.5	Form error in overload delimiter .....	30
2.4.6	MAC overload generation during intermission field following an error frame .....	31
2.4.7	MAC overload generation during intermission field following an overload frame .....	31
2.5	Test class 5, passive error state class .....	32
2.5.1	Passive error flag completion test 1 .....	32
2.5.2	Data frame acceptance after passive error frame transmission .....	32
2.5.3	Acceptance of 7 consecutive dominant bits after passive error flag .....	32
2.5.4	Passive state unchanged on further errors .....	32
2.5.5	Passive error flag completion — Test case 2 .....	32
2.5.6	Form error in passive error delimiter .....	33
2.5.7	Transition from active to passive ERROR FLAG .....	33
2.6	Test class 6, error counter management .....	34
2.6.1	REC increment on bit error in active error flag .....	34
2.6.2	REC increment on bit error in overload flag .....	34
2.6.3	REC increment when active error flag is longer than 13 bits .....	34
2.6.4	REC increment when overload flag is longer than 13 bits .....	34
2.6.5	REC increment on bit error in the ACK field .....	34
2.6.6	REC increment on form error in CRC delimiter .....	35
2.6.7	REC increment on form error in ACK delimiter .....	35
2.6.8	REC increment on form error in EOF field .....	35
2.6.9	REC increment on stuff error .....	35
2.6.10	REC increment on CRC error .....	35
2.6.11	REC increment on dominant bit after end of error flag .....	35
2.6.12	REC increment on form error in error delimiter .....	36

2.6.13	REC increment on form error in overload delimiter .....	36
2.6.14	REC decrement on valid frame reception .....	36
2.6.15	REC decrement on valid frame reception during passive state.....	36
2.6.16	REC non-increment on last bit of EOF field .....	36
2.6.17	REC non-increment on 13-bit length overload flag .....	36
2.6.18	REC non-increment on 13-bit length error flag .....	37
2.6.19	REC non-increment on last bit of error delimiter.....	37
2.6.20	REC non-increment on last bit of overload delimiter .....	37
2.6.21	REC non-decrement on transmission.....	37
2.6.22	REC increment on form error at fixed stuff bit in FD frames .....	37
2.6.23	REC non-increment on protocol exception in FD frames .....	37
2.7	Test class 7, bit timing CAN classical.....	38
2.7.1	Sample point test.....	38
2.7.2	Hard synchronization on SOF reception .....	38
2.7.3	Synchronization when $e > 0$ and $e \leq SJW(N)$ .....	38
2.7.4	Synchronization when $e > 0$ and $e > SJW(N)$ .....	38
2.7.5	Synchronization when $e < 0$ and $ e  \leq SJW(N)$ .....	38
2.7.6	Synchronization when $e < 0$ and $ e  > SJW(N)$ .....	39
2.7.7	Glitch filtering test on positive phase error .....	39
2.7.8	Glitch filtering test on negative phase error.....	39
2.7.9	Glitch filtering test in idle state .....	40
2.7.10	Non-Synchronization after a dominant sampled bit .....	40
2.7.11	Synchronization when $e < 0$ and $ e  \leq SJW(N)$ at "ACK" bit position .....	40
2.7.12	C&S, Synchronisation for $e > 0$ and $e > SJW$ in first bit of intermission field after successful transmission (C&S).....	40
2.8	Test class 8, bit timing CAN FD .....	43
2.8.1	Sample point test.....	43
2.8.2	Hard synchronization on "res" bit .....	43
2.8.3	Synchronization when $e > 0$ and $e \leq SJW(D)$ .....	44
2.8.4	Synchronization when $e > 0$ and $e > SJW(D)$ .....	44
2.8.5	Synchronization when $e < 0$ and $ e  \leq SJW$ .....	45
2.8.6	Synchronization when $e < 0$ and $ e  > SJW$ .....	46
2.8.7	Glitch filtering test on positive phase error .....	46
2.8.8	Glitch filtering test on negative phase error.....	47
2.8.9	No synchronization after a dominant sampled bit .....	47
3	Test type 2, transmitted frame.....	49
3.1	Test class 1, valid frame format .....	49
3.1.1	Identifier and number of data bytes test in base format .....	49
3.1.2	Identifier and number of data bytes test in extended format .....	49
3.1.3	Arbitration in base format frame.....	49
3.1.4	Arbitration in extended format frame test.....	49

3.1.5	Message validation .....	50
3.1.6	Stuff bit generation capability in base format frame .....	50
3.1.7	Stuff bit generation capability in extended frame .....	50
3.1.8	Transmission on the third bit of intermission field after arbitration lost.....	50
3.2	Test class 2, error detection .....	51
3.2.1	Bit error test in base format frame .....	51
3.2.2	Bit error in extended format frame .....	51
3.2.3	Stuff error test in base format frame .....	51
3.2.4	Stuff error test in extended frame format .....	51
3.2.5	Form error test .....	51
3.2.6	Acknowledgement error .....	52
3.2.7	Form field tolerance test for FD frame format .....	52
3.2.8	Bit error at stuff bit position for FD frame payload Bytes .....	52
3.3	Test class 3, error frame management .....	53
3.3.1	Error flag longer than 6 bits .....	53
3.3.2	Transmission on the third bit of intermission field after error frame .....	53
3.3.3	Bit error in error flag .....	53
3.3.4	Form error in error delimiter .....	53
3.4	Test class 4, overload frame management .....	54
3.4.1	MAC overload generation in intermission field .....	54
3.4.2	Eighth bit of an error and overload delimiter .....	54
3.4.3	Transmission on the third bit of intermission after overload frame .....	54
3.4.4	Bit error in overload flag .....	54
3.4.5	Form error in overload delimiter .....	54
3.5	Test class 5, passive error state and bus-off.....	55
3.5.1	Acceptance of active error flag overwriting passive error flag .....	55
3.5.2	Frame acceptance after passive error frame transmission .....	55
3.5.3	Acceptance of 7 consecutive dominant bits after passive error flag .....	55
3.5.4	Reception of a frame during suspend transmission .....	55
3.5.5	Transmission of a frame after suspend transmission — Test case 1 .....	55
3.5.6	Transmission of a frame after suspend transmission — Test case 2 .....	56
3.5.7	Transmission of a frame after suspend transmission — Test case 3 .....	56
3.5.8	Transmission of a frame without suspend transmission .....	56
3.5.9	No transmission of a frame between the third bit of intermission field and eighth bit of suspend transmission.....	56
3.5.10	Bus-off state.....	56
3.5.11	Bus-off recovery .....	56
3.5.12	Completion condition for a passive error flag .....	57
3.5.13	Form error in passive error delimiter .....	57
3.5.14	Maximum recovery time after a corrupted frame .....	57
3.5.15	Transition from active to passive ERROR FLAG .....	57

3.6	Test class 6, error counter management .....	58
3.6.1	TEC increment on bit error during active error flag .....	58
3.6.2	TEC increment on bit error during overload flag .....	58
3.6.3	TEC increment when active error flag is followed by dominant bits .....	58
3.6.4	TEC increment when passive error flag is followed by dominant bits .....	58
3.6.5	TEC increment when overload flag is followed by dominant bits .....	58
3.6.6	TEC increment on bit error in data frame .....	59
3.6.7	TEC increment on form error in a frame .....	59
3.6.8	TEC increment on acknowledgement error .....	59
3.6.9	TEC increment on form error in error delimiter .....	59
3.6.10	TEC increment on form error in overload delimiter .....	59
3.6.11	TEC decrement on successful frame transmission for TEC < 128 .....	59
3.6.12	TEC decrement on successful frame transmission for TEC > 127 .....	60
3.6.13	TEC non-increment on 13-bit long overload flag .....	60
3.6.14	TEC non-increment on 13-bit long error flag .....	60
3.6.15	TEC non-increment on form error at last bit of overload delimiter .....	60
3.6.16	TEC non-increment on form error at last bit of error delimiter .....	60
3.6.17	TEC non-increment on acknowledgement error in passive state .....	60
3.6.18	TEC increment after acknowledgement error in passive state .....	61
3.6.19	TEC non-increment on stuff error during arbitration .....	61
3.6.20	TEC non-decrement on transmission while arbitration lost .....	61
3.6.21	TEC non-increment after arbitration lost and error .....	61
3.7	Test class 7, bit timing .....	62
3.7.1	Sample point test .....	62
3.7.2	Hard synchronization on SOF reception before sample point .....	62
3.7.3	Hard synchronization on SOF reception after sample point .....	62
3.7.4	Synchronization when $e < 0$ and $ e  \leq \text{SJW}(N)$ .....	62
3.7.5	Synchronization for $e < 0$ and $ e  > \text{SJW}(N)$ .....	62
3.7.6	Glitch filtering test on negative phase error .....	63
3.7.7	Non-synchronization on dominant bit transmission .....	63
3.7.8	Synchronization before information processing time .....	63
3.7.9	Synchronization after sample point while sending a dominant bit .....	63
3.7.10	C&S, Synchronisation for $e > 0$ and $e > \text{SJW}$ in first bit of intermission field after successful transmission (C&S Add-on) .....	63
3.8	Test class 8, bit timing in FD frame format .....	65
3.8.1	Sample point test .....	65
3.8.2	Secondary sample point test .....	66
3.8.3	No synchronization within data phase bits when $e < 0$ ; $ e  \leq \text{SJW}(D)$ .....	66
3.8.4	Glitch filtering test on negative phase error within FD frame bits .....	67
3.8.5	No synchronization on dominant bit transmission in FD frames .....	67
4	Test type 3, bi-directional frame type .....	68

4.1	Test class 1, valid frame format class.....	68
4.1.1	C&S, receive standard remote frame and number of data .....	68
4.1.2	C&S, receive extended remote frame and number of data .....	70
4.1.3	C&S, receive standard remote frame DLC greater than 8 .....	72
4.1.4	C&S, transmit standard remote frame and number of data .....	74
4.1.5	C&S, transmit extended remote frame and number of data .....	76
4.1.6	C&S, transmit standard remote frame DLC greater 8 .....	78
4.1.7	C&S, arbitration in standard remote frame .....	80
4.1.8	C&S, arbitration in extended remote frame .....	82
4.2	Test class 2, error detection .....	84
4.3	Test class 3, active error frame management .....	84
4.4	Test class 4, overload frame management .....	84
4.5	Test class 5, passive-error state and bus-off.....	84
4.6	Test class 6, error counter management .....	85
4.6.1	C&S, REC unaffected when increasing TEC .....	85
4.6.2	C&S, TEC unaffected when increasing REC .....	86



# 1 Organisation and methods

## 1.1 References and Conventions

### 1.1.1 References

- [1] CAN Specification - Bosch - Version 2.0
- [2] ISO CD 11898-1, 2014-11-20 Road vehicles - Interchange of digital information - Controller area network (CAN) for high-speed communication
- [4] ISO 11519-2 Road vehicles - Low-speed serial data communication - Part 2: Low-speed Controller area network (CAN)
- [5] ISO 11519-2 Road vehicles - Low-speed serial data communication - Part 2: Low-speed Controller area network (CAN) - AMENDMENT 1
- [6] OSI Conformance Testing General Concepts - 2nd DP 9646-1 - Version 5.10 dated December 07th, 1987
- [7] Bosch reference C model revision 2.2
- [8] ISO CAN Conformance Tests according to "ISO 16845-1:2016 Road vehicles - Controller area network (CAN) - Conformance test plan
- [9] [ISO IEC 9646-1](#) We expect for the first error: TEC +8 +128 +8 (-1 for the next frame) and for the second error TEC +16 (-1 for the next frame) ==> TEC should be 0x9e but we got 0xa6. 1994-12
- [10] [ISO IEC 9646-2](#) Information technology - Open Systems Interconnection - Conformance testing methodology and framework - Abstract Test Suite specification 1994-12
- [11] [ISO IEC 9646-4](#) Information technology - Open Systems Interconnection - Conformance testing methodology and framework - Test realization 1994-12

## 1.1.2 Conventions

### 1.1.2.1 Abbreviations

All abbreviations in this document are written in upper case letters.

ACK	Acknowledgement
CAN	Controller Area Network
CRC	Cyclic Redundancy Check
DLC	Data Length Code
EOF	End Of Frame
IDEN	Identifier
ISO	International Standardisation Organisation
IUT	Implementation Under Test
LLC	Logical Link Control
LME	Layer Management Entity
LT	Lower Tester
MAC	Medium Access Control
MDI	Medium Dependent Interface
OSI	Open System Interface
PCO	Point of Control and Observation
PDU	Protocol Data Unit
PLS	Physical Layer Signalling
PMA	Physical Medium Attachment
REC	Receive Error Counter
RTR	Remote Transmission Request
SOF	Start Of Frame
TEC	Transmit Error Counter
TP	Test Plan
UT	Upper Tester

### 1.1.2.2 Glossary

All expression starting with capital letters are listed in the glossary.

ACK Delimiter:

The second bit of the ACK Field.

ACK Field:

The last field before the EOF used for message validation.

Acknowledgement Error:

Error condition of the transmitter when it does not detect a Dominant bit on the ACK Slot.

ACK Slot:

The first bit of the ACK Field.

Active Error Flag :

First field of an Active Error Frame.

Active Error Frame:

Error frame that starts with an Active (Dominant) Error Flag.

Active State:

A node is in the Active State when it can transmit an Active Error Frame.

Arbitration Field:

The field starting after the SOF bit and finished with the RTR bit.

Bit Error:

Error condition encountered when the received bit does not correspond to the transmitted or to the expected bit.

BRP:

Baud rate prescaler.

Conformance Testing:

Applying the Test Plan to an IUT.

CRC Delimiter:

Last bit of the CRC Field

CRC Error:

Error condition of a receiver when the received CRC code does not match the calculated CRC code.

CRC Field:

The field preceding the ACK Field, consisting of the CRC code and the CRC Delimiter.

Default State:

State of the IUT described in the § 1.3.

Dominant:

see Dominant State.

Dominant State:

The CAN bus is in Dominant State when at least one CAN node drives a Dominant value on the line.

Elementary Test:

See definition in the § 1.3.2.

End Of Frame:

The last field of a data or remote frame before the Intermission Field.

Error Active:

see Active State.

Error Delimiter:

Second field of an Error Frame.

**Error Flag:**  
First field of an Error Frame.

**Error Frame:**  
Formatted sequence of bits indicating an error condition.

**Error Passive:**  
see Passive State.

**Form Error:**  
Error condition encountered in a fixed form field.

**Harmonised CAN Specification:**  
This is the updated ISO 11898 specification which will be harmonised with the CAN Bosch Specification 2.0 as well as the Bosch CAN C Reference Model Rev. 2.2

**Idle State:**  
The CAN bus is in Idle State when no frame is started after Intermission Field.

**Intermission Field:**  
The field after EOF, Error Delimiter or Overload Delimiter.

**Lower Tester:**  
The Lower Tester supervises the Test Suite.

**Overload Delimiter:**  
Second field of an Overload Frame.

**Overload Flag:**  
First field of an Overload Frame.

**Overload Frame:**  
Formatted sequence of bits indicating an overload condition.

**Passive Error Flag:**  
First part of a Passive Error Frame.

**Passive State:**  
The device is in the Passive state because the value of the REC or the TEC has reached the Error Passive limit.

**REC Passive State:**  
The device is in the Passive State because the value of the REC has reached the Error Passive limit.

**Recessive:**  
see Recessive State.

**Recessive State:**  
The CAN bus is in the Recessive State when no CAN node drives a Dominant value on the line.

**Stuff Bit:**  
Specific bit inserted into the bit stream to increase the number of edges for synchronisation purpose.

**Stuff Error:**  
Error condition encountered when an expected Stuff Bit is missing.

**Suspend Transmission Field:**  
Waiting time added after Intermission Field for Error Passive transmitters before it can start another transmission.

**TEC Passive State:**

The device is in the Passive State because the value of the TEC has reached the Error Passive limit.

**Test Case:**

Each Test Case is defined by a specific number and a particular name in the Test Suite.

**Test Class:**

Each Test Type is divided in 7 Test Classes.

**Test Frame:**

Test Frames are CAN frames containing the test pattern specified in this document.

**Test Plan:**

Test Plan is a specific application of the « OSI Conformance Testing General Concepts » standard.

**Test Suite:**

Test Suite check the behaviour of the IUT for particular parameters of the Harmonised CAN Specification.

**Test Type:**

Test Types define the direction of the test frames (e.g. behaviour of the IUT if receiving and/or transmitting messages).

**Time Quantum:**

Elementary time unit of the CAN bit time derived from the oscillator clock and the prescaler.

**TSEG2:**

From [2] Phase\_Seg2.

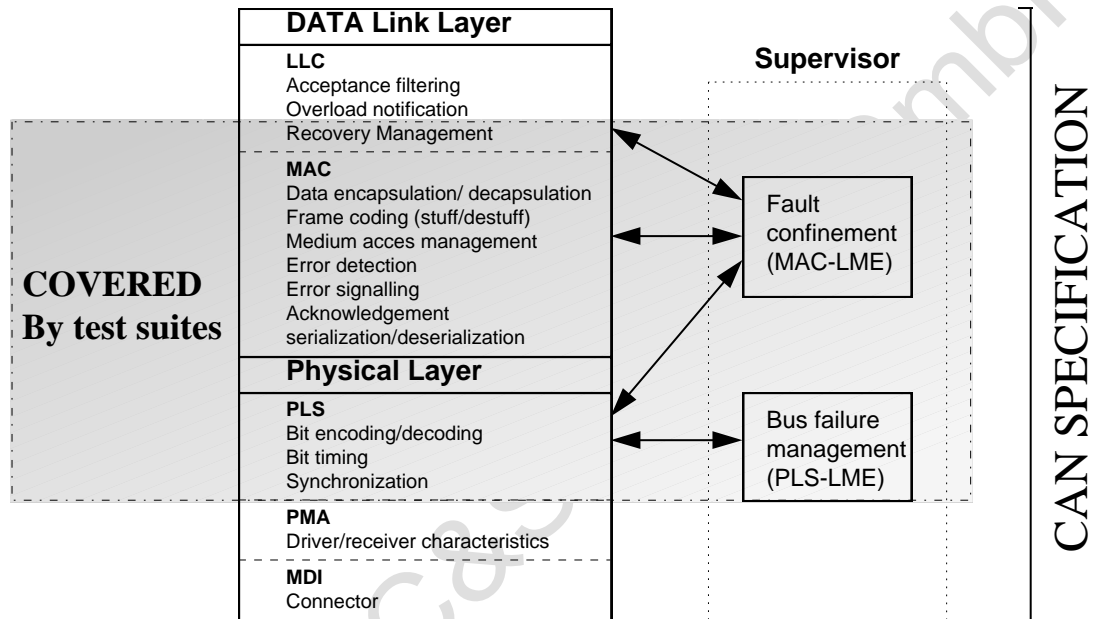
**Upper Tester:**

The Upper Tester acts as an user of the IUT.

## 1.2 Global overview

### 1.2.1 Scope of test plan

The ISO 16845 [8] defines the methodology and the abstract Test Suite necessary to check the conformance of any CAN implementation to the Harmonised CAN specification referring to documents [1], [2].



### 1.2.2 Architecture of the Test Plan

This methodology and the associated abstract Test Suites will be hereafter referred to as Test Plan (TP).

The TP is a specific application of the «OSI Conformance Testing General Concepts» standard [6] and is restricted to the single party testing mode. Since the upper service boundary of a CAN implementation is not standardised and in some cases may not be observed and controlled (due to an application specific behaviour embedded in this implementation, e.g. CAN SLIO (Serial Linked Input/Output), the TP will rely either on the «Co-ordinated test method» or the «Remote test method».

Depending on the test method applied, the TP will involve up to three test functions:

- A Lower Tester (LT) operating in way similar to the CAN implementation to be tested (IUT), running Test Suite and granting test verdict;
- An Upper Tester (UT) acting as user of the IUT (IUT dependant) ;

- A test management protocol between the IUT and the LT. The protocol consists in test co-ordination procedures.

The last two functions are only applicable to the Co-ordinated test procedure.

During test execution, the LT can observe and control the standardised lower service boundary of the IUT (PCO) through the 2 service primitives provided by the CAN physical signalling sub-layer: PLS-Data.indicate and PLS-Data.request in most cases.

The environment that implements the TP is described in the figure 1.

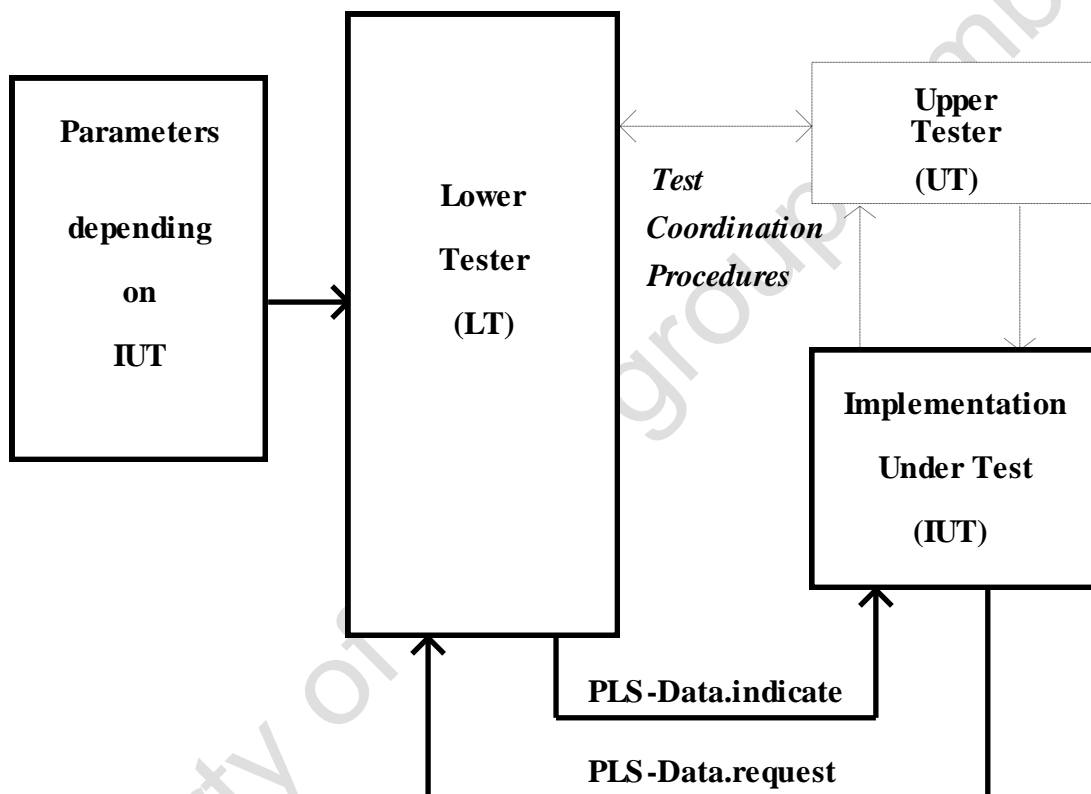


Figure 1. CAN Conformance TP environment

Using the network interface, the LT indicates to the UT the actions to be performed and the UT provides the LT with information concerning the internal behaviour of the IUT.

In order to allow the LT and the UT to communicate, it is necessary to define some test Co-ordination procedures between them. These procedures use the network to the exclusion of any other physical link. They are used to set up the UT and to verify the test results.

## 1.3 Organisation

### 1.3.1 General organisation

The LT verifies if the IUT complies with the MAC, LLC, and PLS sub-layers of Harmonised CAN specification. The LT points out differences between what is expected from the standard and the actual behaviour of the IUT.

The abstract Test Suites of the TP are independent one another. Each abstract Test Suite checks the behaviour of the IUT for a particular parameter of the Harmonised CAN specification. Each Test Case may be executed one after another in any order or alone.

Test Cases requiring variations of individual parameters (identifier, number of data,...) have to be repeated for each value of the parameter. Each repetition is named Elementary Test. A Test Case including different Elementary Tests is valid only if all tests pass.

### 1.3.2 Test Case organisation

Each Elementary Test is made of three states:

- Set-Up state,
- Test state,
- Verification state.

At the PCO, these states involve exchanges of valid sequences of PLS service primitives (CAN frame(s)) or invalid sequences of PLS primitives (invalid CAN frames or noise).

Before the first Elementary Test is started the IUT has to be initialised into the Default State.

#### 1.3.2.1 Set-Up state

The Set-Up state is the state in which the IUT has to be before entering the Test state.

#### 1.3.2.2 Test state

This is the part of the Elementary Test in which the parameter or protocol feature is actually checked. This state needs one or several exchanges or frames. These frames are named test frames.

#### 1.3.2.3 Verification state

Verification state is made of the data reading frames which verify that the data have been handled in accordance with the Harmonised CAN specification. These data have to be checked.



For tests belonging to Classes 1 to 6, the LT must be able to detect the correct value of the bit.  
For bit timing tests (Class 7) the LT must be able to detect a faulty synchronisation of one Time Quantum.

#### 1.3.2.4 Default state

The Default state is characterised by the following default value:

- \* Both REC and TEC must be equal to 0,
- \* No pending transmission must be present,
- \* IUT must be in Idle State,
- \* PLS-Data.indicate and PLS-Data.request must be Recessive.

After the end of each Elementary test, the Default state must be applied.

### 1.3.3 Hierarchical structure of tests

#### 1.3.4 Overview

All the Tests defined in the Test Plan are grouped into categories in order to aid planning, development, understanding or execution of each Test Case. There are three levels of categories:

- \* The Test Types,
- \* The Test Classes,
- \* The Test Cases.

#### 1.3.5 Test Types

The types define the direction of the frames. There are three types:

- **Type 1: Received frame type:**  
It includes all the tests evaluating the behaviour of the IUT for Data frames and Remote frames received by the IUT.
- **Type 2: Transmitted frame type:**  
It includes all the tests evaluating the behaviour of the IUT for Data frames and Remote

frames transmitted by the IUT.

- **Type 3: Bi-directional frame type:**

It includes all the tests with Data frames or Remote frames both received and transmitted by the IUT.

### 1.3.6 Test Classes

Each of the 3 test Types previously defined is divided in 7 classes grouping tests by topic regarding to the Harmonised CAN specification. These 7 classes are:

- **Class 1: Valid frame format class:**

This class includes the tests involving only error free data or remote frames.

- **Class 2: Error detection class:**

This class includes the tests corrupting data or remote frames. These tests check the correct error detection by the IUT.

- **Class 3: Active Error Frame management class:**

This class includes the tests verifying the IUT correct management of error-free and of corrupted Active Error Frames.

- **Class 4: Overload Frame management class:**

This class includes the tests verifying the IUT correct management of error free and of corrupted Overload Frames.

- **Class 5: Passive error state and bus-off class:**

This class includes the tests verifying the IUT behaviour during Passive Error State and bus-off state.

- **Class 6: Error counters management class:**

This class includes the tests verifying the correct management of the TEC and REC by the IUT in both Active and Passive Error State.

- **Class 7: Bit timing class:**

This class includes the tests verifying the correct management of bit timing by the IUT. This class of test must only be applied to components performing only Recessive to Dominant edge synchronisation (if the Dominant to Recessive edge synchronisation exists, it must be disabled).

### 1.3.7 Test Cases

Any basic entry of the test list is intended to check a particular parameter of the Harmonised CAN specification in the IUT.

Each Test Case is defined by a specific number and a particular name in order to differentiate the Test Cases and to easily summarise the goal of the Test Case. Some Test Cases may be subdivided into Elementary Tests which are repetitions of the Test Case for several values of the parameter to test.

The bit timing test variations defined in ISO16845-1 chapter 6.2.3 "Bit rate configuration parameter variation for bit timing tests" will be interpreted as the IUT specific configuration ranges have to match the minimum requirements given in ISO11898-1 chapter 11.3 "PCS specification". Tests which are not applicable because the IUT configuration range did not match to ISO11898-1 requirements will be verdict as FAIL!

Property of C&S group GmbH

## 1.4 LT parameters

### 1.4.1 Overview

The Harmonised CAN specification allows several IUT implementations. Consequently, the user must provide the LT with parameters in order to indicate which kind of IUT is going to be tested.

These parameters can be classified in two categories:

- Communication parameters:

This category defines which tests can be executed for the IUT, and which test method will be applied.

- Application parameters:

This category defines the features of the frames used for each Test Case selected according to the previous parameters.

Note: LT applies to IUT performing only Recessive to Dominant edge synchronisation and operating in single sampling mode.

### 1.4.2 Description of parameters

#### 1.4.2.1 Communication parameters

These parameters are subdivided in three categories. In the text they are always written in upper case letters.

##### 1.4.2.1.1 Implementation parameters

Some parameters depending on the IUT must be specified by the user in order to allow the LT to fit on the IUT. These parameters are:

CAN\_VERSION:

This parameter indicates the version implemented in the IUT. It can take three values:

- A: IUT is handling 11 bit identifiers,
- B: IUT is handling 11 and 29 bit identifiers,
- BP: IUT is handling 11 identifiers and tolerating 29 bit identifiers

#### OPEN/SPECIFIC:

This parameter indicates if the IUT is open regarding to the application layers, or if it includes a specific application. It can take two values:

**OPEN:** open IUT allowing the test Co-ordination procedure to be implemented in an UT. These IUT are tested with the «Co-ordinated test method» of document [6].

**SPECIFIC:** IUT that can be tested only with the help of a specific configuration procedure. These IUT are tested with the «Remote test method» of document [6].

#### 1.4.2.1.2 *Timing parameters*

The LT also needs some timing parameters to be in accordance with the IUT and the UT characteristics. These parameters are:

##### TIMEOUT:

This parameter indicates the minimum duration time for which the LT must wait in order to respect the three following conditions:

- The UT must have enough time to put the IUT into the Set-Up state,
- The IUT must have enough time to transmit a response frame after a Remote frame,
- The LT must consider an optional additional waiting time after the end of the minimum bus-off recovery sequence before the IUT enters Error Active State again.

##### TSYS:

This parameter indicates the duration of the IUT system clock (clock used as input of the prescaler).

##### BRP:

This parameter indicates the value of the prescaler (the duration of a Time Quantum is  $TQ = TSYS * BRP$ ).

##### NTQ:

This parameter indicates the number of time quanta per bit.

##### TSEG2:

This parameter indicates the number of time quanta for the phase «buffer segment 2».

##### SJW:

This parameter indicates the number of time quanta for the Re-synchronisation Jump Width.

IPT:

Information Processing Time.

#### **1.4.2.1.3 Other parameters**

The other parameter related to the IUT is:

NDATA:

This parameter is a set of DLC values which an IUT accepts for data exchange with higher layers. DLC values have to be in the range of 0 to 8.

#### **1.4.2.1.4 The application parameters**

Except the Test Cases for which a particular profile of data is defined by the TP, the content of the data used during the Test Cases must be chosen by the user.

## **2 Test type 1, received frame**

### **2.1 Test class 1, valid frame format**

#### **2.1.1 Identifier and number of data test in base format**

Covered by ISO16845-1:2016

#### **2.1.2 Identifier and number of data test in extended format**

Covered by ISO16845-1:2016

#### **2.1.3 Reception after arbitration lost**

Covered by ISO16845-1:2016

#### **2.1.4 Acceptance of non-nominal bit in base format frame**

Covered by ISO16845-1:2016

### **2.1.5 Acceptance of non-nominal bit in extended format frame**

Covered by ISO16845-1:2016

### **2.1.6 Protocol exception behaviour on non-nominal bit**

Covered by ISO16845-1:2016

### **2.1.7 Minimum time for bus idle after protocol exception handling**

Covered by ISO16845-1:2016

### **2.1.8 DLC greater than 8**

Covered by ISO16845-1:2016

### **2.1.9 Absent bus idle — Valid frame reception**

Covered by ISO16845-1:2016

### **2.1.10 Stuff acceptance test in base format frame**

Covered by ISO16845-1:2016



### **2.1.11 Stuff acceptance test in extended format frame**

Covered by ISO16845-1:2016

### **2.1.12 Message validation**

Covered by ISO16845-1:2016

### **2.1.13 C&S, Form field tolerance test for FD frame format**

Property of C&S group GmbH

### 2.1.13.1 Test Case organisation

Following Table specifies the test of an IUT that accepts a long ACK slot of two bit with dominant value in a valid frame.

Item	Description
Purpose	The purpose of this test is to verify the point of time at which a message is taken to be valid by the IUT.
CAN_VERSION	CAN FD enabled
Test variables	EOF, FDF = 1
Elementary test cases	Number of elementary tests: 1 #1 ACK slot of two bit with dominant value.
Set-up	The IUT is left in the default state.
Execution	A single test frame is used for the elementary test.
Response	The IUT shall not generate any error flag during the test. The IUT shall acknowledge the test frame. The IUT shall generate an overload frame. The data received by the IUT during the test state shall match the data sent in the test frame.

## **2.2 Test class 2, error detection**

### **2.2.1 Bit error in data frame**

Covered by ISO16845-1:2016

### **2.2.2 Stuff error for basic frame**

Covered by ISO16845-1:2016

### **2.2.3 Stuff error for extended frame**

Covered by ISO16845-1:2016

### **2.2.4 Stuff error for FD frame payload bytes**

Covered by ISO16845-1:2016

### **2.2.5 CRC error**

Covered by ISO16845-1:2016

### **2.2.6 Combination of CRC error and form error**

Covered by ISO16845-1:2016

### **2.2.7 Form error in data frame at “CRC delimiter” bit position**

Covered by ISO16845-1:2016

### **2.2.8 Form error at fixed stuff bit in FD frames**

Covered by ISO16845-1:2016

### **2.2.9 Form error in data frame at “ACK delimiter” bit position**

Covered by ISO16845-1:2016

### **2.2.10 Form error in data frame at “EOF”**

Covered by ISO16845-1:2016

### **2.2.11 Message non-validation**

Covered by ISO16845-1:2016

## **2.3 Test class 3, error frame management**

### **2.3.1 Error flag longer than 6 bits**

Covered by ISO16845-1:2016

### **2.3.2 Data frame starting on the third bit of intermission field**

Covered by ISO16845-1:2016

### **2.3.3 Bit error in error flag**

Covered by ISO16845-1:2016

### **2.3.4 Form error in error delimiter**

Covered by ISO16845-1:2016

Property of C&S group GmbH

## **2.4 Test class 4, overload frame management**

### **2.4.1 MAC overload generation during intermission field**

Covered by ISO16845-1:2016

### **2.4.2 Last bit of EOF**

Covered by ISO16845-1:2016

### **2.4.3 Eighth bit of an error and overload delimiter**

Covered by ISO16845-1:2016

### **2.4.4 Bit error in overload flag**

Covered by ISO16845-1:2016

### **2.4.5 Form error in overload delimiter**

Covered by ISO16845-1:2016

#### **2.4.6 MAC overload generation during intermission field following an error frame**

Covered by ISO16845-1:2016

#### **2.4.7 MAC overload generation during intermission field following an overload frame**

Covered by ISO16845-1:2016

Property of C&S group GmbH

## **2.5 Test class 5, passive error state class**

### **2.5.1 Passive error flag completion test 1**

Covered by ISO16845-1:2016

### **2.5.2 Data frame acceptance after passive error frame transmission**

Covered by ISO16845-1:2016

### **2.5.3 Acceptance of 7 consecutive dominant bits after passive error flag**

Covered by ISO16845-1:2016

### **2.5.4 Passive state unchanged on further errors**

Covered by ISO16845-1:2016

### **2.5.5 Passive error flag completion — Test case 2**

Covered by ISO16845-1:2016



## 2.5.6 Form error in passive error delimiter

Covered by ISO16845-1:2016

## 2.5.7 Transition from active to passive ERROR FLAG

Covered by ISO16845-1:2016

Property of C&S group GmbH

## **2.6 Test class 6, error counter management**

### **2.6.1 REC increment on bit error in active error flag**

Covered by ISO16845-1:2016

### **2.6.2 REC increment on bit error in overload flag**

Covered by ISO16845-1:2016

### **2.6.3 REC increment when active error flag is longer than 13 bits**

Covered by ISO16845-1:2016

### **2.6.4 REC increment when overload flag is longer than 13 bits**

Covered by ISO16845-1:2016

### **2.6.5 REC increment on bit error in the ACK field**

Covered by ISO16845-1:2016

### **2.6.6 REC increment on form error in CRC delimiter**

Covered by ISO16845-1:2016

### **2.6.7 REC increment on form error in ACK delimiter**

Covered by ISO16845-1:2016

### **2.6.8 REC increment on form error in EOF field**

Covered by ISO16845-1:2016

### **2.6.9 REC increment on stuff error**

Covered by ISO16845-1:2016

### **2.6.10 REC increment on CRC error**

Covered by ISO16845-1:2016

### **2.6.11 REC increment on dominant bit after end of error flag**

Covered by ISO16845-1:2016

### **2.6.12 REC increment on form error in error delimiter**

Covered by ISO16845-1:2016

### **2.6.13 REC increment on form error in overload delimiter**

Covered by ISO16845-1:2016

### **2.6.14 REC decrement on valid frame reception**

Covered by ISO16845-1:2016

### **2.6.15 REC decrement on valid frame reception during passive state**

Covered by ISO16845-1:2016

### **2.6.16 REC non-increment on last bit of EOF field**

Covered by ISO16845-1:2016

### **2.6.17 REC non-increment on 13-bit length overload flag**

Covered by ISO16845-1:2016

### **2.6.18 REC non-increment on 13-bit length error flag**

Covered by ISO16845-1:2016

### **2.6.19 REC non-increment on last bit of error delimiter**

Covered by ISO16845-1:2016

### **2.6.20 REC non-increment on last bit of overload delimiter**

Covered by ISO16845-1:2016

### **2.6.21 REC non-decrement on transmission**

Covered by ISO16845-1:2016

### **2.6.22 REC increment on form error at fixed stuff bit in FD frames**

Covered by ISO16845-1:2016

### **2.6.23 REC non-increment on protocol exception in FD frames**

Covered by ISO16845-1:2016

## 2.7 Test class 7, bit timing CAN classical

### 2.7.1 Sample point test

Covered by ISO16845-1:2016

### 2.7.2 Hard synchronization on SOF reception

Covered by ISO16845-1:2016

### 2.7.3 Synchronization when $e > 0$ and $e \leq \text{SJW}(N)$

Covered by ISO16845-1:2016

### 2.7.4 Synchronization when $e > 0$ and $e > \text{SJW}(N)$

Covered by ISO16845-1:2016

### 2.7.5 Synchronization when $e < 0$ and $|e| \leq \text{SJW}(N)$

Covered by ISO16845-1:2016

### **2.7.6 Synchronization when $e < 0$ and $|e| > SJW(N)$**

Covered by ISO16845-1:2016

### **2.7.7 Glitch filtering test on positive phase error**

Covered by ISO16845-1:2016

### **2.7.8 Glitch filtering test on negative phase error**

Covered by ISO16845-1:2016

Property of C&S group GmbH

## **2.7.9 Glitch filtering test in idle state**

### **2.7.9.1 Glitch filtering test in idle state 1 (single pulse)**

Covered by ISO16845-1:2016

### **2.7.9.2 Glitch filtering test in idle state 2 (multiple pulses)**

Covered by ISO16845-1:2016

## **2.7.10 Non-Synchronization after a dominant sampled bit**

Covered by ISO16845-1:2016

## **2.7.11 Synchronization when $e < 0$ and $|e| \leq \text{SJW}(N)$ at “ACK” bit position**

Covered by ISO16845-1:2016

## **2.7.12 C&S, Synchronisation for $e > 0$ and $e > \text{SJW}$ in first bit of intermission field after successful transmission (C&S)**



### 2.7.12.1 Purpose and limits of this Test Case

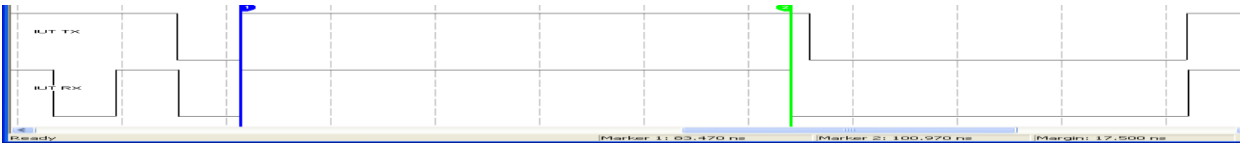
CAN\_VERSION  $\in$  {A, B, BP}

The purpose of this test is to verify the behaviour of an IUT acting as a receiver detecting a positive phase error  $e$  on a Recessive to Dominant edge inserted into first intermission bit with  $e > SJW$ . A soft synchronisation is expected. The values tested for “ $e$ ” are measured in Time Quanta with  $e \in [(SJW + 1), (NTQ - (TSEG2 + 1))]$ . There is one Elementary Test to perform for each possible value of  $e$ . C&S did the test twice, with BRP = 1 and BRP  $\geq 2$ . (typical value is 4)

### 2.7.12.2 Test Case organisation

State	Description
Set-Up	No action required, the IUT is left in the Default State.
Test	The LT generates a dominant Bit within the first bit of intermission. The recessive to dominant edge (start of bit) is delayed by an amount of “ $e$ ” time quanta and shortens the same bit by an amount of $e - SJW$ .
Verification	The IUT should generate an Overload Frame 1 bit time – $(e - SJW)$ time quanta after the Recessive to Dominant edge of the delayed dominant Bit (disturbance in intermission 1 bit).
Reference	ISO11898 12.4.2.1 C) “Hard synchronization is performed during interframe space (with the exception of the first bit of intermission) whenever there is a »recessive« to »dominant« edge”.
	We cause a falling edge within the middle of first bit of intermission to set a dominant bit and check if the following overload frame start according soft sync rules +- a possible tolerance because of the 9 bits without a synchronisation. = Pass In case the overload frame starts exact one bit time after the falling edge on RX we have a hard synchronisation. = Fail

2.7.12.3 Test Scenario:



Frame | ACK | ACK | EOF + e | X + Overload Frame |  
 Del  
 $X = 1\text{Bit} - (e\text{-SJW})$

## **2.8 Test class 8, bit timing CAN FD**

### **2.8.1 Sample point test**

#### **2.8.1.1 Sample point test at “BRS” bit position**

Covered by ISO16845-1:2016

#### **2.8.1.2 Sample point test at “DATA” field**

Covered by ISO16845-1:2016

#### **2.8.1.3 Sample point test at “CRC delimiter” bit position**

Covered by ISO16845-1:2016

### **2.8.2 Hard synchronization on “res” bit**

#### **2.8.2.1 Hard synchronization on “res” bit (delayed end of FDF)**

Covered by ISO16845-1:2016

### **2.8.2.2 Hard synchronization on “res” bit (early end of FDF)**

Covered by ISO16845-1:2016

### **2.8.3 Synchronization when $e > 0$ and $e \leq \text{SJW}(D)$**

#### **2.8.3.1 Synchronization when $e > 0$ and $e \leq \text{SJW}(D)$ at ESI bit position**

Covered by ISO16845-1:2016

#### **2.8.3.2 Synchronization when $e > 0$ and $e \leq \text{SJW}(D)$ at DATA field**

Covered by ISO16845-1:2016

#### **2.8.3.3 Synchronization when $e > 0$ and $e \leq \text{SJW}(D)$ at CRC delimiter bit position**

Covered by ISO16845-1:2016

### **2.8.4 Synchronization when $e > 0$ and $e > \text{SJW}(D)$**

#### **2.8.4.1 Synchronization when $e > 0$ and $e > \text{SJW}(D)$ at ESI bit position**

Covered by ISO16845-1:2016

#### **2.8.4.2 Synchronization when $e > 0$ and $e > SJW(D)$ at DATA field**

Covered by ISO16845-1:2016

#### **2.8.4.3 Synchronization when $e > 0$ and $e > SJW(D)$ at “CRC delimiter” bit position**

Covered by ISO16845-1:2016

#### **2.8.5 Synchronization when $e < 0$ and $|e| \leq SJW$**

##### **2.8.5.1 Synchronization when $e < 0$ and $|e| \leq SJW(D)$ at “ESI” bit position**

Covered by ISO16845-1:2016

##### **2.8.5.2 Synchronization when $e < 0$ and $|e| \leq SJW(D)$ at “DATA” field**

Covered by ISO16845-1:2016

##### **2.8.5.3 Synchronization when $e < 0$ and $|e| \leq SJW(N)$ at “ACK” bit position**

Covered by ISO16845-1:2016

## **2.8.6 Synchronization when $e < 0$ and $|e| > SJW$**

### **2.8.6.1 Synchronization when $e < 0$ and $|e| > SJW(D)$ at “ESI” bit position**

Covered by ISO16845-1:2016

### **2.8.6.2 Synchronization when $e < 0$ and $|e| > SJW(D)$ at “DATA” field**

Covered by ISO16845-1:2016

### **2.8.6.3 Synchronization when $e < 0$ and $|e| > SJW(N)$ at “ACK” bit position**

Covered by ISO16845-1:2016

## **2.8.7 Glitch filtering test on positive phase error**

### **2.8.7.1 Glitch filtering test on positive phase error at “res” bit position**

Covered by ISO16845-1:2016

### **2.8.7.2 Glitch filtering test on positive phase error at “DATA” field**

Covered by ISO16845-1:2016

### **2.8.7.3 Glitch filtering test on positive phase error at “ACK” bit position**

Covered by ISO16845-1:2016

## **2.8.8 Glitch filtering test on negative phase error**

### **2.8.8.1 Glitch filtering test on negative phase error at “ESI” bit position**

Covered by ISO16845-1:2016

### **2.8.8.2 Glitch filtering test on negative phase error at “DATA” field**

Covered by ISO16845-1:2016

### **2.8.8.3 Glitch filtering test on negative phase error at “ACK” bit position**

Covered by ISO16845-1:2016

## **2.8.9 No synchronization after a dominant sampled bit**

### **2.8.9.1 No synchronization after a dominant sampled bit at “BRS” bit position**

Covered by ISO16845-1:2016

### **2.8.9.2 No synchronization after a dominant sampled bit at “DATA” field**

Covered by ISO16845-1:2016

### **2.8.9.3 No synchronization after a dominant sampled bit at “CRC delimiter” bit position**

Covered by ISO16845-1:2016

Property of C&S group GmbH



### **3 Test type 2, transmitted frame**

#### **3.1 Test class 1, valid frame format**

##### **3.1.1 Identifier and number of data bytes test in base format**

Covered by ISO16845-1:2016

##### **3.1.2 Identifier and number of data bytes test in extended format**

Covered by ISO16845-1:2016

##### **3.1.3 Arbitration in base format frame**

Covered by ISO16845-1:2016

##### **3.1.4 Arbitration in extended format frame test**

Covered by ISO16845-1:2016

### **3.1.5 Message validation**

Covered by ISO16845-1:2016

### **3.1.6 Stuff bit generation capability in base format frame**

Covered by ISO16845-1:2016

### **3.1.7 Stuff bit generation capability in extended frame**

Covered by ISO16845-1:2016

### **3.1.8 Transmission on the third bit of intermission field after arbitration lost**

Covered by ISO16845-1:2016

Property of C&S group GmbH

## **3.2 Test class 2, error detection**

### **3.2.1 Bit error test in base format frame**

Covered by ISO16845-1:2016

### **3.2.2 Bit error in extended format frame**

Covered by ISO16845-1:2016

### **3.2.3 Stuff error test in base format frame**

Covered by ISO16845-1:2016

### **3.2.4 Stuff error test in extended frame format**

Covered by ISO16845-1:2016

### **3.2.5 Form error test**

Covered by ISO16845-1:2016

### **3.2.6 Acknowledgement error**

Covered by ISO16845-1:2016

### **3.2.7 Form field tolerance test for FD frame format**

Covered by ISO16845-1:2016

### **3.2.8 Bit error at stuff bit position for FD frame payload Bytes**

Covered by ISO16845-1:2016

Property of C&S group GmbH

### **3.3 Test class 3, error frame management**

#### **3.3.1 Error flag longer than 6 bits**

Covered by ISO16845-1:2016

#### **3.3.2 Transmission on the third bit of intermission field after error frame**

Covered by ISO16845-1:2016

#### **3.3.3 Bit error in error flag**

Covered by ISO16845-1:2016

#### **3.3.4 Form error in error delimiter**

Covered by ISO16845-1:2016

Property of C&S group GmbH

## **3.4 Test class 4, overload frame management**

### **3.4.1 MAC overload generation in intermission field**

Covered by ISO16845-1:2016

### **3.4.2 Eighth bit of an error and overload delimiter**

Covered by ISO16845-1:2016

### **3.4.3 Transmission on the third bit of intermission after overload frame**

Covered by ISO16845-1:2016

### **3.4.4 Bit error in overload flag**

Covered by ISO16845-1:2016

### **3.4.5 Form error in overload delimiter**

Covered by ISO16845-1:2016

## **3.5 Test class 5, passive error state and bus-off**

### **3.5.1 Acceptance of active error flag overwriting passive error flag**

Covered by ISO16845-1:2016

### **3.5.2 Frame acceptance after passive error frame transmission**

Covered by ISO16845-1:2016

### **3.5.3 Acceptance of 7 consecutive dominant bits after passive error flag**

Covered by ISO16845-1:2016

### **3.5.4 Reception of a frame during suspend transmission**

Covered by ISO16845-1:2016

### **3.5.5 Transmission of a frame after suspend transmission — Test case 1**

Covered by ISO16845-1:2016

### **3.5.6 Transmission of a frame after suspend transmission — Test case 2**

Covered by ISO16845-1:2016

### **3.5.7 Transmission of a frame after suspend transmission — Test case 3**

Covered by ISO16845-1:2016

### **3.5.8 Transmission of a frame without suspend transmission**

Covered by ISO16845-1:2016

### **3.5.9 No transmission of a frame between the third bit of intermission field and eighth bit of suspend transmission**

Covered by ISO16845-1:2016

### **3.5.10 Bus-off state**

Covered by ISO16845-1:2016

### **3.5.11 Bus-off recovery**

Covered by ISO16845-1:2016



### **3.5.12 Completion condition for a passive error flag**

Covered by ISO16845-1:2016

### **3.5.13 Form error in passive error delimiter**

Covered by ISO16845-1:2016

### **3.5.14 Maximum recovery time after a corrupted frame**

Covered by ISO16845-1:2016

### **3.5.15 Transition from active to passive ERROR FLAG**

Covered by ISO16845-1:2016

Property of C&S group GmbH

## **3.6 Test class 6, error counter management**

### **3.6.1 TEC increment on bit error during active error flag**

Covered by ISO16845-1:2016

### **3.6.2 TEC increment on bit error during overload flag**

Covered by ISO16845-1:2016

### **3.6.3 TEC increment when active error flag is followed by dominant bits**

Covered by ISO16845-1:2016

### **3.6.4 TEC increment when passive error flag is followed by dominant bits**

Covered by ISO16845-1:2016

### **3.6.5 TEC increment when overload flag is followed by dominant bits**

Covered by ISO16845-1:2016

### **3.6.6 TEC increment on bit error in data frame**

Covered by ISO16845-1:2016

### **3.6.7 TEC increment on form error in a frame**

Covered by ISO16845-1:2016

### **3.6.8 TEC increment on acknowledgement error**

Covered by ISO16845-1:2016

### **3.6.9 TEC increment on form error in error delimiter**

Covered by ISO16845-1:2016

### **3.6.10 TEC increment on form error in overload delimiter**

Covered by ISO16845-1:2016

### **3.6.11 TEC decrement on successful frame transmission for $TEC < 128$**

Covered by ISO16845-1:2016

### **3.6.12 TEC decrement on successful frame transmission for TEC > 127**

Covered by ISO16845-1:2016

### **3.6.13 TEC non-increment on 13-bit long overload flag**

Covered by ISO16845-1:2016

### **3.6.14 TEC non-increment on 13-bit long error flag**

Covered by ISO16845-1:2016

### **3.6.15 TEC non-increment on form error at last bit of overload delimiter**

Covered by ISO16845-1:2016

### **3.6.16 TEC non-increment on form error at last bit of error delimiter**

Covered by ISO16845-1:2016

### **3.6.17 TEC non-increment on acknowledgement error in passive state**

Covered by ISO16845-1:2016

### **3.6.18 TEC increment after acknowledgement error in passive state**

Covered by ISO16845-1:2016

### **3.6.19 TEC non-increment on stuff error during arbitration**

Covered by ISO16845-1:2016

### **3.6.20 TEC non-decrement on transmission while arbitration lost**

Covered by ISO16845-1:2016

### **3.6.21 TEC non-increment after arbitration lost and error**

Covered by ISO16845-1:2016

Property of C&S group GmbH

## **3.7 Test class 7, bit timing**

### **3.7.1 Sample point test**

Covered by ISO16845-1:2016

### **3.7.2 Hard synchronization on SOF reception before sample point**

Covered by ISO16845-1:2016

### **3.7.3 Hard synchronization on SOF reception after sample point**

Covered by ISO16845-1:2016

### **3.7.4 Synchronization when $e < 0$ and $|e| \leq \text{SJW}(N)$**

Covered by ISO16845-1:2016

### **3.7.5 Synchronization for $e < 0$ and $|e| > \text{SJW}(N)$**

Covered by ISO16845-1:2016

### 3.7.6 Glitch filtering test on negative phase error

Covered by ISO16845-1:2016

### 3.7.7 Non-synchronization on dominant bit transmission

Covered by ISO16845-1:2016

### 3.7.8 Synchronization before information processing time

Covered by ISO16845-1:2016

### 3.7.9 Synchronization after sample point while sending a dominant bit

Covered by ISO16845-1:2016

### 3.7.10 C&S, Synchronisation for $e > 0$ and $e > SJW$ in first bit of intermission field after successful transmission (C&S Add-on)

#### 3.7.10.1 Purpose and limits of this Test Case

CAN\_VERSION  $\in$  {A, B, BP}

The purpose of this test is to verify the behaviour of an IUT acting as a receiver detecting a positive phase error  $e$  on a Recessive to Dominant edge inserted into first intermission bit with  $e > SJW$ . A soft synchronisation is expected. Second test purpose is to check the correct behaviour (being receiver) after being transmitter of the previous frame. The values tested for  $e$  are measured in Time Quanta with  $e \in [(SJW + 1), (NTQ - (TSEG2 + 1))]$ .

There is one Elementary Test to perform for each possible value of e. C&S did the test twice, with BRP = 1 and BRP >= 2. (typical value is 4)

### 3.7.10.2 Test Case organisation

State	Description
Set-Up	No action required. The IUT is left in the Default State.
Test	The LT generates a dominant Bit within the first bit of intermission. The recessive to dominant edge (start of bit) is delayed by an amount of e time quanta and shortens the same bit by an amount of e - SJW.
Verification	The IUT should generate an Overload Frame 1 bit time - (e - SJW) time quanta after the Recessive to Dominant edge of the delayed dominant Bit (disturbance in intermission 1 bit).
Reference	ISO11898 12.4.2.1 C) "Hard synchronization is performed during interframe space (with the exception of the first bit of intermission) whenever there is a »recessive« to »dominant« edge."
Notes	We cause a falling edge within the middle of first bit of intermission to set a dominant bit and check if the following overload frame start according soft sync rules +- a possible tolerance because of the 9 bits without a synchronisation. = Pass  In case the overload frame starts exact one bit time after the falling edge on RX we have a hard synchronisation. = Fail

### 3.7.10.3 Test Scenario:



Frame | ACK | ACK | EOF + e | X + Overload Frame

Del

$$X = 1\text{Bit} - (e - \text{SJW})$$



## **3.8 Test class 8, bit timing in FD frame format**

### **3.8.1 Sample point test**

#### **3.8.1.1 Sample point test at “res” bit**

Covered by ISO16845-1:2016

#### **3.8.1.2 Sample point test at BRS bit**

Covered by ISO16845-1:2016

#### **3.8.1.3 Sample point test in DATA field**

Covered by ISO16845-1:2016

#### **3.8.1.4 Sample point test at CRC delimiter bit**

Covered by ISO16845-1:2016

## **3.8.2 Secondary sample point test**

### **3.8.2.1 No secondary sample point test at “res” bit**

Covered by ISO16845-1:2016

### **3.8.2.2 No secondary sample point at BRS bit test**

Covered by ISO16845-1:2016

### **3.8.2.3 Secondary sample point test in DATA field**

Covered by ISO16845-1:2016

### **3.8.2.4 Secondary sample point test at end of data phase**

Covered by ISO16845-1:2016

## **3.8.3 No synchronization within data phase bits when $e < 0$ ; $|e| \leq \text{SJW}(D)$**

### **3.8.3.1 No synchronization within data phase bits when $e < 0$ ; $|e| \leq \text{SJW}(D)$ (BRS)**

Covered by ISO16845-1:2016

### **3.8.3.2 No synchronization within data phase bits when $e < 0$ ; $|e| \leq \text{SJW}(D)$ (DATA field)**

Covered by ISO16845-1:2016

### **3.8.4 Glitch filtering test on negative phase error within FD frame bits**

#### **3.8.4.1 Glitch filtering test on negative phase error within FD frame at “ESI” bit position**

Covered by ISO16845-1:2016

#### **3.8.4.2 Glitch filtering test on negative phase error within FD frames at “DATA” field position**

Covered by ISO16845-1:2016

### **3.8.5 No synchronization on dominant bit transmission in FD frames**

#### **3.8.5.1 No synchronization on dominant bit transmission in FD frame at “ESI” bit position**

Covered by ISO16845-1:2016

#### **3.8.5.2 No Synchronization on dominant bit transmission in FD frames at “DATA” field position**

Covered by ISO16845-1:2016

## 4 Test type 3, bi-directional frame type

### 4.1 Test class 1, valid frame format class

#### 4.1.1 C&S, receive standard remote frame and number of data

##### 4.1.1.1 Purpose and limits of this Test Case

CAN\_VERSION  $\in$  {A, B, BP}

This test verifies the behaviour of the IUT when receiving a correct remote request frame with different identifiers and different DLC numbers in a standard format frame.

Tested identifiers:  $\in$  [000h, 7EFh]  $\cup$  [7F0h, 7FFh],

Tested DLC values:  $\in$  [0, 8]

### 4.1.1.2 Test Case organisation

State	Description
Set-Up	No action required. The IUT is left in the Default State.
Test	A single test frame is used for each Elementary Test.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must acknowledge the test frame. The received DLC value by the IUT during the test state must match the DLC sent in the test frame.
Reference	ISO 11898 [2]. Section 10.4.3.1
Notes	C&S Add-on test. Not defined by ISO 16845

### 4.1.1.3 Test Script

30101001\_CS - 30101009\_CS

LowerTesterSendsSOT(ISO\_TST\_FCT\_RMT\_1, TST\_FRM\_0, RTR\_ DLC\_X, REPEAT\_1, 0, 0)

LowerTesterWaitsForEvent(BIT\_TIME, WAIT\_ISO)

LowerTesterSends(SOF, 0x100, RTR\_1, IDE\_0, r0\_0, DLC\_X, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceives(SOF, 0x100, RTR\_0, IDE\_0, r0\_0, DLC\_X, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceivesEOT(ISO\_TST\_FCT\_RMT\_1, 0)

## 4.1.2 C&S, receive extended remote frame and number of data

### 4.1.2.1 Purpose and limits of this Test Case

CAN\_VERSION  $\in$  {B, BP}

This test verifies the behaviour of the IUT when receiving a correct remote request frame with different identifiers and different DLC numbers in a extended format frame.

Tested identifiers:  $\in$  [00000000, 1FFFFFFFh]

Tested DLC values:  $\in$  [0, 8]

### 4.1.2.2 Test Case organisation

State	Description
Set-Up	No action required, the IUT is left in the Default State.
Test	A single test frame is used for each Elementary Test.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must acknowledge the test frame. The received DLC value by the IUT during the test state must match the DLC sent in the test frame.
Reference	ISO 11898 [2]. Section 10.4.3.1
Notes	C&S Add-on test. Not defined by ISO 16845

### 4.1.2.3 Test Script

30102001\_CS - 30102009\_CS

LowerTesterSendsSOT(ISO\_TST\_FCT\_RMT\_1, TST\_FRM\_10, RTR\_DLC\_X, REPEAT\_1, 0, 0)

LowerTesterWaitsForEvent(BIT\_TIME, WAIT\_ISO)

LowerTesterSends(SOF, 0x000, SRR\_1, IDE\_1, 0x0100, RTR\_1, r1\_0, r0\_0, DLC\_X, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceives(SOF, 0x000, SRR\_1, IDE\_1, 0x0100, RTR\_0, r1\_0, r0\_0, DLC\_X, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceivesEOT(ISO\_TST\_FCT\_RMT\_1, 0)

Property of C&S group GmbH

### 4.1.3 C&S, receive standard remote frame DLC greater than 8

#### 4.1.3.1 Purpose and limits of this Test Case

CAN\_VERSION  $\in$  {A, B, BP}

This test verifies the behaviour of the IUT when receiving a correct remote frame with a DLC Field greater than 8.

There are 7 Elementary Tests, for which DLC  $\in$  [9, Fh].

#### 4.1.3.2 Test Case organisation

State	Description
Set-Up	No action required. The IUT is left in the Default State.
Test	A single test frame is used for each of the Elementary Tests.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must acknowledge the test frame. The received DLC value by the IUT during the test state must match the DLC sent in the test frame.
Reference	ISO 11898 [2]. Section 10.4.3.1
Notes	C&S Add-on test. Not defined by ISO 16845



### 4.1.3.3 Test Script

30103001\_CS - 30103007\_CS

LowerTesterSendsSOT(ISO\_TST\_FCT\_RMT\_1, TST\_FRM\_0, RTR\_DLC\_X, REPEAT\_2, 0, 0)

LowerTesterWaitsForEvent(BIT\_TIME, WAIT\_ISO)

LowerTesterSends(SOF, 0x100, RTR\_1, IDE\_0, r0\_0, DLC\_X, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceives(SOF, 0x100, RTR\_0, IDE\_0, r0\_0, DLC\_X, 0xA55AF00F55AA3CC3, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceivesEOT(ISO\_TST\_FCT\_RMT\_1, 0)

Property of C&S group GmbH

## 4.1.4 C&S, transmit standard remote frame and number of data

### 4.1.4.1 Purpose and limits of this Test Case

CAN\_VERSION  $\in$  {A, B, BP}

This test verifies the capacity of the IUT to transmit a remote frame with different identifiers and different numbers of DLC in a standard format frame.

Tested identifiers: IDEN  $\in$  [000h, 7EFh]  $\cup$  [7F0h, 7FFh],

Tested number of DLC:  $\in$  [0, 8].

### 4.1.4.2 Test Case organisation

State	Description
Set-Up	No action required. The IUT is left in the Default State.
Test	A single test frame is used for each Elementary Test. The LT causes the IUT to transmit a remote frame with the parameters listed below.
Verification	The IUT shall not generate any Error Flag during the test. The content of the frame must match the LT request. The value of DLC must match to the requested value.
Reference	ISO 11898 [2]. Section 10.4.3.1
Notes	C&S Add-on test. Not defined by ISO 16845

#### 4.1.4.3 Test Script

30104001\_CS - 30104009\_CS

LowerTesterSendsSOT(ISO\_TST\_FCT\_RMT\_2, TST\_FRM\_0, RTR\_DLC\_X, REPEAT\_1, 0, 0)

LowerTesterReceives(SOF, 0x100, RTR\_1, IDE\_0, r0\_0, DLC\_X, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterWaitsForEvent(BIT\_TIME, WAIT\_ISO)

LowerTesterSends(SOF, 0x100, RTR\_0, IDE\_0, r0\_0, DLC\_X, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceivesEOT(ISO\_TST\_FCT\_RMT\_2, 0)

Property of C&S group GmbH

## 4.1.5 C&S, transmit extended remote frame and number of data

### 4.1.5.1 Purpose and limits of this Test Case

CAN\_VERSION ∈ {B}

This test verifies the capacity of the IUT to transmit a remote frame with different identifiers and different numbers of DLC in an extended format frame.

Tested identifiers: IDEN ∈ [00000000, 1FFFFFFFh]

Tested number of DLC: ∈ [0, 8].

### 4.1.5.2 Test Case organisation

State	Description
Set-Up	No action required. The IUT is left in the Default State.
Test	A single test frame is used for each Elementary Test. The LT causes the IUT to transmit a remote frame with the parameters listed below.
Verification	The IUT shall not generate any Error Flag during the test. The content of the frame must match the LT request. The value of DLC must match to the requested value.
Reference	ISO 11898 [2]. Section 10.4.3.1
Notes	C&S Add-on test. Not defined by ISO 16845

### 4.1.5.3 Test Script

30105001\_CS - 30105009\_CS.

LowerTesterSendsSOT(ISO\_TST\_FCT\_RMT\_2, TST\_FRM\_10, RTR\_DLC\_X, REPEAT\_1, 0, 0)

LowerTesterReceives(SOF, 0x000, SRR\_1, IDE\_1, 0x100, RTR\_1, r1\_0, r0\_0, DLC\_X, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterWaitsForEvent(BIT\_TIME, WAIT\_ISO)

LowerTesterSends(SOF, 0x000, SRR\_1, IDE\_1, 0x100, RTR\_0, r1\_0, r0\_0, DLC\_X, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceivesEOT(ISO\_TST\_FCT\_RMT\_2, 0)

Property of C&S group GmbH

## 4.1.6 C&S, transmit standard remote frame DLC greater 8

### 4.1.6.1 Purpose and limits of this Test Case

CAN\_VERSION  $\in$  {A, B, BP}

This test verifies the capacity of the IUT to transmit a correct remote frame with a DLC Field greater than 8.

There are 7 Elementary Tests, for which DLC  $\in$  [9, Fh].

Tested number of DLC:  $\in$  [0, 8].

### 4.1.6.2 Test Case organisation

State	Description
Set-Up	No action required. The IUT is left in the Default State.
Test	A single test frame is used for each Elementary Test. The LT causes the IUT to transmit a remote frame with the parameters listed below.
Verification	The IUT shall not generate any Error Flag during the test. The content of the frame must match the LT request. The value of DLC must match to the requested value.
Reference	ISO 11898 [2]. Section 10.4.3.1
Notes	C&S Add-on test. Not defined by ISO 16845. REMARK: this test only apply if the device has this option

### 4.1.6.3 Test Script

30106001\_CS - 30106007\_CS

LowerTesterSendsSOT(ISO\_TST\_FCT\_RMT\_2, TST\_FRM\_0, RTR\_ DLC\_X, REPEAT\_1, 0, 0)

LowerTesterReceives(SOF, 0x100, RTR\_1, IDE\_0, r0\_0, DLC\_X, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3 )

LowerTesterWaitsForEvent(BIT\_TIME, WAIT\_ISO)

LowerTesterSends(SOF, 0x100, RTR\_0, IDE\_0, r0\_0, DLC\_X, 0xA55AF00F55AA3CC3, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceivesEOT(ISO\_TST\_FCT\_RMT\_2, 0)

Property of C&S group GmbH

## 4.1.7 C&S, arbitration in standard remote frame

### 4.1.7.1 Purpose and limits of this Test Case

CAN\_VERSION  $\in$  {A, B, BP},

This test verifies the capability of the IUT to manage the arbitration mechanism the RTR bit.

There is one Elementary Test to perform.

C&S perform an additional test to verify the correct reception of arbitration winning frame.

### 4.1.7.2 Test Case organisation

State	Description
Set-Up	No action required. The IUT is left in the Default State.
Test	The LT causes the IUT to transmit a frame. Then the LT forces the RTR bit in the Arbitration Field to the Dominant State and continues to send a valid frame.
Verification	The IUT must become receiver when sampling the Dominant bit sent by the LT. The data received by the IUT must match the data sent by the LT. As soon as the bus is idle the IUT must restart the transmission of the frame. The IUT shall not generate any Error Flag during the test. The content of the frame must match the LT request.
Reference	ISO 11898 [2]. Section 10.4.3.1
Notes	C&S Add-on test. Not defined by ISO 16845. It is possible that the CAN will receive the arbitration winning data frame into the RTR Message buffer instead retransmitting the RTR Request frame. (RTR Handling within the same buffer) Second test Purpose: (30107001_CS)



The CAN have to receive the arbitration winning data frame and should store it into one RX Message buffer so one receive message buffer have to be enabled already before the CAN transmit the RTR Frame.
---

### 4.1.7.3 Test Script

30107000\_CS

```
LowerTesterSendsSOT(ISO_TST_FCT_RMT_2, TST_FRM_29, RTR_DLC_0, REPEAT_1, 0, 0)
LowerTesterSends(<SYNC>, SOF, 0x7EF, RTR_0, IDE_0, r0_0, DLC_0, CRC, CRC_DEL, ACK_0, ACK_DEL, EOF, INTERM_#3)
LowerTesterReceives(SOF, 0x7EF, RTR_1, IDE_0, r0_0, DLC_0, CRC, CRC_DEL, ACK_0, ACK_DEL, EOF, INTERM_#3)
LowerTesterWaitsForEvent(BIT_TIME, WAIT_ISO)
LowerTesterSends( SOF, 0x7EF, RTR_0, IDE_0, r0_0, DLC_0, CRC, CRC_DEL, ACK_0, ACK_DEL, EOF, INTERM_#3)
LowerTesterReceivesEOT(ISO_TST_FCT_RMT_2, 0)
```

30107001\_CS

```
LowerTesterSendsSOT(ISO_TST_FCT_RMT_2, TST_FRM_29, RTR_DLC_0, REPEAT_1, 1, 0)
LowerTesterSends(<SYNC>, SOF, 0x7EF, RTR_0, IDE_0, r0_0, DLC_0, CRC, CRC_DEL, ACK_0, ACK_DEL, EOF, INTERM_#3)
LowerTesterReceives(SOF, 0x7EF, RTR_1, IDE_0, r0_0, DLC_0, CRC, CRC_DEL, ACK_0, ACK_DEL, EOF, INTERM_#3)
LowerTesterWaitsForEvent(BIT_TIME, WAIT_ISO)
LowerTesterSends( SOF, 0x7EF, RTR_0, IDE_0, r0_0, DLC_0, CRC, CRC_DEL, ACK_0, ACK_DEL, EOF, INTERM_#3)
LowerTesterReceivesEOT(ISO_TST_FCT_RMT_2, 0)
```

## 4.1.8 C&S, arbitration in extended remote frame

### 4.1.8.1 Purpose and limits of this Test Case

CAN\_VERSION ∈ {B}

This test verifies the capability of the IUT to manage the arbitration mechanism the RTR bit.

There is one Elementary Test to perform.

C&S perform an additional test to verify the correct reception of arbitration winning frame.

### 4.1.8.2 Test Case organisation

State	Description
Set-Up	No action required. The IUT is left in the Default State.
Test	The LT causes the IUT to transmit a frame. Then the LT forces the RTR bit in the Arbitration Field to the Dominant State and continues to send a valid frame.
Verification	The IUT must become receiver when sampling the Dominant bit sent by the LT. The data received by the IUT must match the data sent by the LT. As soon as the bus is idle the IUT must restart the transmission of the frame. The IUT shall not generate any Error Flag during the test. The content of the frame must match the LT request.
Reference	ISO 11898 [2]. Section 10.4.3.1
Notes	C&S Add-on test. Not defined by ISO 16845. It is possible that the CAN will receive the arbitration winning data frame into the RTR Message buffer instead retransmitting the RTR Request frame. (RTR Handling within the same buffer) Second test Purpose: (30108001_CS)

The CAN have to receive the arbitration winning data frame and should store it into one RX Message buffer so one receive message buffer have to be enabled already before the CAN transmit the RTR Frame.
---

### 4.1.8.3 Test Script

30108000\_CS

LowerTesterSendsSOT(ISO\_TST\_FCT\_RMT\_2, TST\_FRM\_40, RTR\_DLC\_0, REPEAT\_1, 0, 0)

LowerTesterSends(<SYNC>, SOF, 0x7EF, SRR\_1, IDE\_1, 0x3FFFF, RTR\_0, r1\_0, r0\_0, DLC\_0, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceives(SOF, 0x7EF, SRR\_1, IDE\_1, 0x3FFFF, RTR\_1, r1\_0, r0\_0, DLC\_0, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterWaitsForEvent(BIT\_TIME, WAIT\_ISO)

LowerTesterSends( SOF, 0x7EF, SRR\_1, IDE\_1, 0x3FFFF, RTR\_0, r1\_0, r0\_0, DLC\_0, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceivesEOT(ISO\_TST\_FCT\_RMT\_2, 0)

30108001\_CS

LowerTesterSendsSOT(ISO\_TST\_FCT\_RMT\_2, TST\_FRM\_40, RTR\_DLC\_0, REPEAT\_1, 0, 0)

LowerTesterSends(<SYNC>, SOF, 0x7EF, SRR\_1, IDE\_1, 0x3FFFF, RTR\_0, r1\_0, r0\_0, DLC\_0, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceives(SOF, 0x7EF, SRR\_1, IDE\_1, 0x3FFFF, RTR\_1, r1\_0, r0\_0, DLC\_0, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterWaitsForEvent(BIT\_TIME, WAIT\_ISO)

LowerTesterSends( SOF, 0x7EF, SRR\_1, IDE\_1, 0x3FFFF, RTR\_0, r1\_0, r0\_0, DLC\_0, CRC, CRC\_DEL, ACK\_0, ACK\_DEL, EOF, INTERM\_#3)

LowerTesterReceivesEOT(ISO\_TST\_FCT\_RMT\_2, 0)

## **4.2 Test class 2, error detection**

No tests specified.

## **4.3 Test class 3, active error frame management**

No tests specified.

## **4.4 Test class 4, overload frame management**

No tests specified.

## **4.5 Test class 5, passive-error state and bus-off**

No tests specified.

Property of C&S group GmbH

## 4.6 Test class 6, error counter management

### 4.6.1 C&S, REC unaffected when increasing TEC

#### 4.6.1.1 Test Case organisation

Following table specifies the test of REC will remain unaffected when increasing TEC.

Item	Description
Purpose	This test verifies that increasing REC and TEC are independent operations.
CAN_VERSION	Classical CAN, CAN FD tolerant, CAN FD enabled
Test variables	REC, TEC, FDF = 0
Elementary test cases	There is one elementary test to perform.
Set-up	The IUT is left in the default state.
Execution	The LT causes the IUT to increase its REC up to 127. Then, LT causes the IUT to increase its TEC up to 128. Then, the LT sends a frame containing a stuff error in data field.
Response	Each increment of the TEC shall be responded by an active error flag. The IUT responds to the stuff error with a passive error flag.

## 4.6.2 C&S, TEC unaffected when increasing REC

### 4.6.2.1 Test Case organisation

Following table specifies the test of TEC will remain unaffected when increasing REC.

Item	Description	
Purpose	This test verifies that increasing REC and TEC are independent operations.	
CAN_VERSION	Classical CAN, CAN FD tolerant, CAN FD enabled	CAN FD enabled
Test variables	REC, TEC, FDF = 0	REC, TEC, FDF = 1
Elementary test cases	There is one elementary test to perform.	
Set-up	The IUT is left in the default state.	
Execution	The LT causes the IUT to increase its TEC up to 127. Then, LT causes the IUT to increase its REC up to 128. Then, the LT causes the IUT to send a frame and corrupts this frame in data field.	
Response	Each increment of the REC shall be responded by an active error flag. The IUT responds to the corrupted bit with a passive error flag.	