DRAFT INTERNATIONAL STANDARD ISO/DIS 16845

ISO/TC **22**/SC **3** Secretariat: **DIN**

Voting begins on
2000-10-05

Voting terminates on
2001-03-05

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION

Road vehicles — Controller aera network (CAN) — Conformance test plan

Véhicules routiers — Gestionnaire de réseau de communication (CAN) — Plan d'essai de conformité

ICS 43.040.10

To expedite distribution, this document is circulated as received from the committee secretariat. ISO Central Secretariat work of editing and text composition will be undertaken at publication stage.

Pour accélérer la distribution, le présent document est distribué tel qu'il est parvenu du secrétariat du comité. Le travail de rédaction et de composition de texte sera effectué au Secrétariat central de l'ISO au stade de publication.

THIS DOCUMENT IS A DRAFT CIRCULATED FOR COMMENT AND APPROVAL. IT IS THEREFORE SUBJECT TO CHANGE AND MAY NOT BE REFERRED TO AS AN INTERNATIONAL STANDARD UNTIL PUBLISHED AS SUCH.

IN ADDITION TO THEIR EVALUATION AS BEING ACCEPTABLE FOR INDUSTRIAL, TECHNOLOGICAL, COMMERCIAL AND USER PURPOSES, DRAFT INTERNATIONAL STANDARDS MAY ON OCCASION HAVE TO BE CONSIDERED IN THE LIGHT OF THEIR POTENTIAL TO BECOME STANDARDS TO WHICH REFERENCE MAY BE MADE IN NATIONAL REGULATIONS.

Contents

Foreword

ISO (the International Organization for Standardization) is a world-wide 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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Road vehicles — Controller area network (CAN) — Conformance test plan

1 Scope

This International Standard specifies the methodology and the abstract test suite necessary to check the conformance of any CAN implementation to the harmonised CAN specifications in ISO 11898 and CAN Specification - Bosch - Version 2.0.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 11898:1993, Road vehicles - Interchange of digital information - Controller area network (CAN) for high-speed communication

ISO 11898:1993/Amd.1:1995, Road vehicles - Interchange of digital information - Controller area network (CAN) for high-speed communication - AMENDMENT 1

ISO 11519-2:1994, Road vehicles - Low-speed serial data communication - Part 2 : Low-speed Controller area network (CAN)

ISO 11519-2:1994/Amd.1:1995, Road vehicles - Low-speed serial data communication - Part 2 : Low-speed Controller area network (CAN) - AMENDMENT 1

ISO 9646-1:1994, Information technology - Open Systems interconnection - Conformance testing methodology and framework - Part 1: General concepts

Bosch reference C model revision 2.2

CAN Specification - Bosch - Version 2.0

3 Terms and definitions

For the purpose of this International Standard following terms and definitions apply.

All expression starting with capital letters are listed.

3.1 **ACK Delimiter:** The second bit of the ACK Field.

3.2 **ACK Field :** The last field before the EOF used for message validation.

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

3.4 **ACK Slot:** The first bit of the ACK Field.

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

3.6 **Active Error Frame:** Error frame that starts with an Active (Dominant) Error Flag.

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

3.8 **Arbitration Field:** The field starting after the SOF bit and finished with the RTR bit.

3.9 **Bit Error:** Error condition encountered when the received bit does not correspond to the transmitted or to the expected bit.

3.10 **Conformance Testing:** Applying the Test Plan to an IUT.

3.11 **CRC Delimiter:** Last bit of the CRC Field.

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

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

3.14 **Default State:** State of the IUT described in the paragraph.

3.15 **Dominant:** see Dominant State.

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

3.17 **Elementary Test:** See definition in the paragraph.

3.18 **End Of Frame :** The last field of a data or remote frame before the Intermission Field.

3.19 **Error Active:** see Active State.

3.20 **Error Delimiter:** Second field of an Error Frame.

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

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

3.23 **Error Passive:** see Passive State.

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

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

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

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

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

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

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

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

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

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

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

3.35 Recessive: see Recessive State.

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

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

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

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

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

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

3.42 Test Class: Each Test Type is divided in 7 Test Classes.

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

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

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

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

3.47 Time Quantum: Elementary time unit of the CAN bit time derived from the oscillator clock and the prescaler.

3.48 Upper Tester: The Upper Tester acts as an user of the IUT.

4 Abbreviated terms

All abbreviations are written in upper case letters.

ACK	Acknowledgement
CAN	Controller Area Network
CRC	Cyclic Redundancy Check
DLC	Data Length Code
EOF	End Of Frame

DEN	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

5 General

5.1 Architecture of the test plan

The architecture of the test plan is shown in figure 1.

This methodology and the associated abstract test suites are hereafter referred to as test plan (TP).

The TP is a specific application of ISO 9646-1 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 shall rely either on the «Co-ordinated test method» or the «Remote test method».

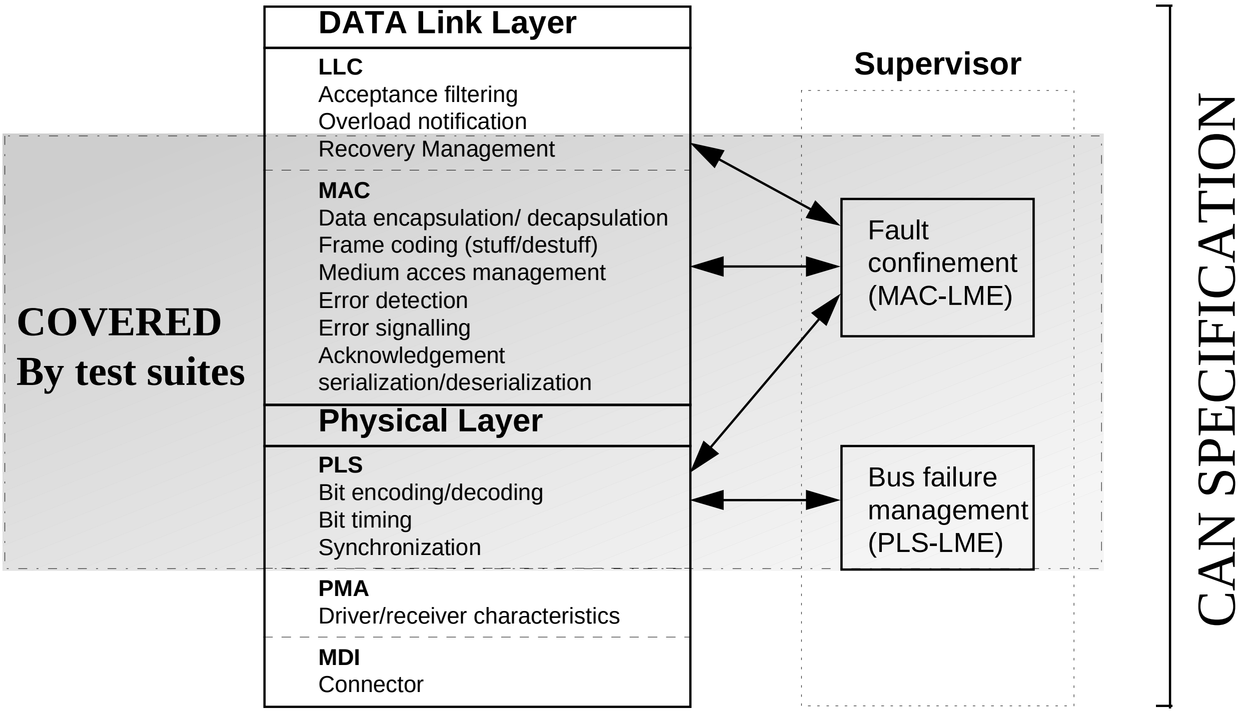

Figure 1 - Architecture of the test plan

Depending on the test method applied, the TP shall 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 standardized lower service boundary of the IUT (PCO) through the two (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 2.

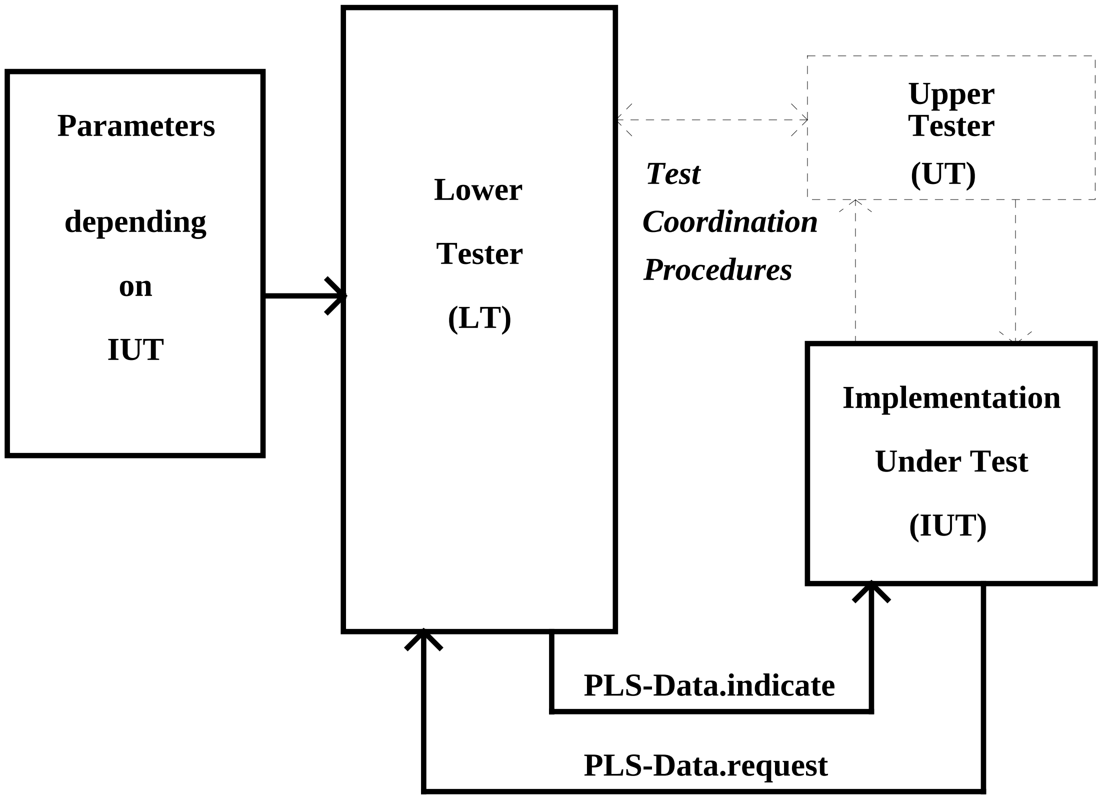

Figure 2 - CAN conformance TP environment

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 specify 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.

5.2 Organisation

5.2.1 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.

5.2.2 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.

5.2.2.1 Set-up state

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

5.2.2.2 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.

5.2.2.3 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.

5.2.2.4 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.

5.3 Hierarchical structure of tests

5.3.1 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.

5.3.2 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.

5.3.3 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).

5.3.4 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.

6 LT parameters

6.1 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 specifies which tests can be executed for the IUT, and which test method shall be applied.

- **Application parameters:** This category specifies 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.

6.2 Description of parameters

6.2.1 Communication parameters

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

6.2.1.1 Implementation parameters

Some parameters depending on the IUT shall 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].

6.2.1.2 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.

Phase_Seg2:
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. In all tests, the Re-synchronisation Jump Width shall be programmed to its full range, up to its maximum value which is the minimum of Phase_Seg1 and 4 tq.

IPT:
Information Processing Time.

IUT Delay Time:
This parameter shall be considered for Bit Timing Class tests. It indicates the time difference between the response of the IUT and the response of an ideal IUT (without internal delays) to an edge causing synchronisation. The IUT Delay Time is the sum of the IUT's input delay time and the IUT's output delay time, it is measured according to ISO 11898-2 (chapter 5.4.6).

6.2.1.3 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.

6.2.1.4 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.

7 Received frame type

7.1 Valid frame format class

7.1.1 Identifier and number of data test in standard format

7.1.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

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

Tested identifiers: ∈ [000h, 7EFh] ∪ [7F0h, 7FFh],

Tested number of data bytes: ∈ [0, 8]

7.1.1.2 Test Case organisation

Test Case organisation see table 1.

Table 1 - 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 shall acknowledge the test frame. The data received by the IUT during the test state must match the data sent in the test frame.

7.1.2 Identifier and number of data test in extended format test 1

7.1.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {B}

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

Tested identifiers: ∈ [00000000, 1FFFFFFFh]

Tested number of data bytes: ∈ [0, 8]

7.1.2.2 Test Case organisation

Test Case organisation see table 2.

Table 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 shall acknowledge the test frame. The data received by the IUT during the test state must match the data sent in the test frame.

7.1.3 Identifier and number of data test in extended format test 2

7.1.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {BP}

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

Tested identifiers:	∈ [00000000, 1FFFFFFFh]
Tested number of data bytes:	∈ [0, 8]

7.1.3.2 Test Case organisation

Test Case organisation see table 3.

Table 3 - 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.

7.1.4 Acceptance of non-nominal «r1,r0» combination in standard format

7.1.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A}

The purpose of this test is to verify that the IUT accepts the non-nominal value of «r1, r0» bits in a valid standard frame.

There are three (3) values to test, see table 4

Table 4 - r1 and r0 bits

r1	r0
1	1
1	0
0	1

7.1.4.2 Test Case organisation

Test Case organisation see table 5.

Table 5 - 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 three Elementary Tests.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must acknowledge the test frame. The data received by the IUT during the test state must match the data sent in the test frame.

7.1.5 Acceptance of non-nominal «IDE, r0» combination in standard format

7.1.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {B, BP}

The purpose of this test is to verify that the IUT accepts the non-nominal value of «IDE, r0» bits in a valid standard frame.

There is 1 value to test, see table 6.

Table 6 - Non.nominal IDE

IDE	r0
0	1

7.1.5.2 Test Case organisation

Test Case organisation see table 7.

Table 7 - 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 the Elementary Test.
Verification	The IUT shall not generate any Error Flag in this test frame. The IUT must acknowledge the test frame. The data received by the IUT during the test state must match the data sent in the test frame.

7.1.6 Acceptance of non-nominal «SRR, r1, r0» in extended format test 1

7.1.6.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {B}

The purpose of this test is to verify that the IUT accepts the non-nominal value of «SRR, r1, r0» bits in a valid extended frame.

The 7 values to test see table 8.

Table 8 - non-nominal value of «SRR, r1, r0» bits

SRR	r1	R0
1	1	1
1	1	0
1	0	1
0	1	1
0	1	0
0	0	1
0	0	0

7.1.6.2 Test Case organisation

Test Case organisation see table 9.

Table 9 - 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 seven Elementary Tests.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must acknowledge the test frame. The data received by the IUT during the test state must match the data sent in the test frame.

7.1.7 Acceptance of non-nominal «SRR, r1, r0» in extended format test 2

7.1.7.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {BP}

The purpose of this test is to verify that the IUT accepts the non-nominal value of «SRR, r1, r0» bits in a valid extended frame.

The seven (7) values to test are shown in table 10.

Table 10 - Non-nominal value of «SRR, r1, r0» bits

SRR	r1	r0
1	1	1
1	1	0
1	0	1
0	1	1
0	1	0
0	0	1
0	0	0

7.1.7.2 Test Case organisation

Test Case organisation see table 11.

Table 11 - 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 seven Elementary Tests.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must acknowledge the test frame.

7.1.8 DLC greater than 8

7.1.8.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

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

There are seven (7) Elementary Tests, for which DLC ∈ [9, Fh].

7.1.8.2 Test Case organisation

Test Case organisation see table 12.

Table 12 - 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. If 8 is an element of NDATA, the data received by the IUT during the test state must match the data sent in the test frame.

7.1.9 Absent bus idle

7.1.9.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies the behaviour of the IUT when receiving two consecutive frames not separated by a bus Idle State.

There are two cases to be tested:
the second frame starts after the second or
after the third Intermission bit of the first frame.

7.1.9.2 Test Case organisation

Test Case organisation see table 13.

Table 13 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	Two test frames are used for each of the two Elementary Tests.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must acknowledge the test frames.

7.1.10 Stuff acceptance test 1

7.1.10.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies the behaviour of the IUT when receiving a correct standard frame with particular data containing critical stuffing bit profiles in the different fields of the frame.

The fields of the tested frame are given in the table14.

Table 14 - Fields of the tested frame

Frame	IDEN	RTR	CTRL	DATA
1	078h	0	8	01h, E1h, E1h, E1h, E1h, E1h, E1h, E1h
2	41Fh	0	1	00
3	707h	0	1Fh	0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh
4	360h	0	10h	-
5	730h	0	10h	-
6	47Fh	0	01h	1Fh
7	758h	0	00h	-
8	777h	0	01h	1Fh
9	7Efh	1	02h	-
10	3Eah	1	1Fh	-

7.1.10.2 Test Case organisation

Test Case organisation see table 15.

Table 15 - 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 data received by the IUT during the test state must match the data sent in the test frame.

7.1.11 Stuff acceptance test 2

7.1.11.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {B, BP}

This test verifies the behaviour of the IUT when receiving a correct extended frame with particular data containing critical stuffing bit profiles in the different fields of the frame.

The fields of the tested frame are given in the table 16.

Table 16 - Fields of the tested frame

Frame	IDEN 1	SRR	IDEN 2	RTR	CTRL	DATA
1	1F0h	1	30F0Fh	0	8	3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch
2	1F0h	1	0F0F0h	0	1	00
3	078h	1	31717h	0	1Fh	0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh
4	078h	1	00FF0h	0	3Ch	1Fh, 0Fh, E0h, F0h, 7Fh, E0h, FFh, 20h
5	7EEh	1	0	0	01h	A0h
6	02Fh	1	0540Fh	1	20h	-
7	557h	1	15557h	1	3Fh	-

7.1.11.2 Test Case organisation

Test Case organisation see table 17.

Table 17 - 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 n Elementary Tests.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must acknowledge the test frame. The data received by the IUT during the test state must match the data sent in the test frame.

7.1.12 Message validation

7.1.12.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify the point of time at which a message is taken to be valid by the IUT.

7.1.12.2 Test Case organisation

Test Case organisation see table 18.

Table 18 - Test Case organisation

State	Description
Set-up	The IUT has to be initialised with data different from those used in the test frame.
Test	A single test frame is used for the Elementary Test. The last bit of the EOF is forced to Dominant State.
Verification	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.

7.1.13 DLC not belonging to NDATA

7.1.13.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies the behaviour of the IUT when receiving a correct frame with a DLC not belonging to NDATA and lower than 9.

7.1.13.2 Test Case organisation

Test Case organisation see table 19.

Table 19 - 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.

7.2 Error detection class

7.2.1 Bit Error in data frame

7.2.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects a Bit Error when the Dominant ACK Slot is forced to Recessive State by LT.

7.2.1.2 Test Case organisation

Test Case organisation see table 20.

Table 20 - 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 the Elementary Test. The Dominant acknowledgement bit sent by the IUT is forced to Recessive State.
Verification	The IUT must generate an Active Error Frame starting at the bit position following the Bit Error.

7.2.2 Stuff Error test 1

7.2.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects a Stuff Error whenever it receives 6 consecutive bits of the same value until the position of the CRC Delimiter in a standard frame.

The frames in table 21 are used in this test:

Table 21 - Stuff Error test 1

Frame	IDEN	RTR	CTRL	DATA
1	078h	0	8	01h, E1h, E1h, E1h, E1h, E1h, E1h, E1h
2	41Fh	0	1	00
3	707h	0	1Fh	0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh
4	360h	0	10h	-
5	730h	0	10h	-
6	47Fh	0	01h	1Fh
7	758h	0	00h	-
8	777h	0	01h	1Fh
9	7EFh	1	02h	-
10	3EAh	1	1Fh	-

7.2.2.2 Test Case organisation

Test Case organisation see table 22.

Table 22 - 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. In each Elementary Test the LT forces one of the Stuff Bits to its complement.
Verification	The IUT must generate an Active Error Frame starting at the bit position following the Stuff Error.

7.2.3 Stuff Error test 2

7.2.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {B, BP}

This test verifies that the IUT detects a Stuff Error whenever it receives 6 consecutive bits of the same value until the position of the CRC Delimiter in an extended frame.

The frames in figure 23 are used in this test:

Table 23 – Frames specifications

Frame	IDEN 1	SRR	IDEN 2	RTR	CTRL	DATA
1	1F0h	1	30F0Fh	0	8	3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch
2	1F0h	1	0F0F0h	0	1	00
3	078h	1	31717h	0	1Fh	0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh, 0Fh
4	078h	1	00FF0h	0	3Ch	1Fh, 0Fh, E0h, F0h, 7Fh, E0h, FFh, 20h
5	7EEh	1	0	0	01h	A0h
6	02Fh	1	0540Fh	1	20h	-
7	557h	1	15557h	1	3Fh	-

7.2.3.2 Test Case organisation

Test Case organisation see table 24.

Table 24 - 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. In each Elementary Test the LT forces one of the Stuff Bits to its complement.
Verification	The IUT must generate an Active Error Frame starting at the bit position following the Stuff Error.

7.2.4 CRC Error test

7.2.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify:

- that the IUT uses the specific CRC mechanism as defined in the Harmonised CAN Specification.
- that an IUT detecting a CRC Error generates an error frame at the correct position.

There are two Elementary Tests to perform:

- A Dominant bit in the CRC Field is changed in a Recessive one,
- A Recessive bit in the CRC Field is changed in a Dominant one.

7.2.4.2 Test Case organisation

Test Case organisation see table 25.

Table 25 - 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. In each Elementary Test the LT modifies the frame according to 7.2.4.1.
Verification	The IUT must not acknowledge the test frame. The IUT must generate an Active Error Frame starting at the bit position following the ACK Delimiter.

7.2.5 Combination of CRC Error and Form Error test

7.2.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an IUT detecting a CRC Error and a Form Error on the CRC Delimiter in the same frame generates only one single 6 bits long Error Flag starting on the bit following the CRC Delimiter.

There is only one Elementary Test to perform.

7.2.5.2 Test Case organisation

Test Case organisation see table 26.

Table 26 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 the Elementary Test.
Verification	The IUT must generate an Active Error Frame.

7.2.6 Form Error in data frame test 1

7.2.6.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects a Form Error when the Recessive bit of CRC Delimiter is forced to Dominant State by LT.

There is only one Elementary Test to perform.

7.2.6.2 Test Case organisation

Test Case organisation see table 27.

Table 27 - 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 the Elementary Test.
Verification	The IUT must generate an Active Error Frame at the bit position following the CRC Delimiter.

7.2.7 Form Error in data frame test 2

7.2.7.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects a Form Error when the Recessive ACK Delimiter is forced to Dominant State by LT.

There is only one Elementary Test to perform.

7.2.7.2 Test Case organisation

Test Case organisation see table 28.

Table 28 - 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 the Elementary Test.
Verification	The IUT must generate an Active Error Frame at the bit position following the ACK Delimiter.

7.2.8 Form Error in data frame test 3

7.2.8.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects a Form Error when one of six first Recessive bits of EOF is forced to Dominant State by LT.

- There are three Elementary Tests to perform, corrupting the first, the last and the 3rd bit position.

7.2.8.2 Test Case organisation

Test Case organisation see table 29.

Table 29 - 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 the Elementary Test.
Verification	The IUT must generate an Active Error Frame at the bit position following the corrupted bit.

7.2.9 Message non-validation

7.2.9.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify the point of time at which a message is still considered as non-valid by the IUT.

There is only one Elementary Test to perform.

7.2.9.2 Test Case organisation

Test Case organisation see table 30

Table 30 - Test Case organisation

State	Description
Set-up	The IUT has to be initialised with data different from those used in the test frame.
Test	A single test frame is used for the Elementary Test. The sixth bit of the EOF is forced to Dominant. A single test frame is used for the Elementary Test.
Verification	The IUT must generate an Active Error Frame. The data initialised during the Set-up state must remain unchanged. No frame reception is indicated to the upper layers of the IUT.

7.3 Error Frame management class

7.3.1 Error Flag longer than 6 bits

7.3.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT tolerates up to seven consecutive Dominant bits after sending an Active Error Flag.

There are three Elementary Tests to perform, lengthening the Error Flag by one, four, and seven Dominant bits.

7.3.1.2 Test Case organisation

Test Case organisation see table 31.

Table 31 - 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 generate an Error Frame. The LT lengthens the Error Flag generated by the IUT.
Verification	After sending the Active Error Flag, the IUT sends Recessive bits.

7.3.2 Data frame starting on the third bit of Intermission Field

7.3.2.1 Purpose and limits of this Test Case

CAN_VERSION ∊ {A, B, BP}

The purpose of this test is to verify that an IUT accepts a frame starting after the second bit of the intermission following the Error Frame it has transmitted.

There is one Elementary Test to perform:

7.3.2.2 Test Case organisation

Test Case organisation see table 32.

Table 32 - 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 generate an Error Frame. A single test frame is started 2 bits after the end of the Error Delimiter.
Verification	The IUT must acknowledge the test frame. The data received by the IUT during the test state must match the data sent in the test frame.

7.3.3 Bit error in Error Flag

7.3.3.1 Purpose and limits of this Test Case

CAN_VERSION ∊ {A, B, BP}

This test verifies that the IUT detects a bit Error when one of the six Dominant bits of the Error Flag it transmits is forced to Recessive State by LT.

There are three Elementary Tests to perform, corrupting the first, the third, and the sixth bit of the Error Flag.

7.3.3.2 Test Case organisation

Test Case organisation see table 33.

Table 33 - 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 generate an Error Frame. The LT forces one of the above mentioned bits of the Error Frame generated by the IUT to Recessive State.
Verification	The IUT must restart with an Active Error Frame at the bit position following the corrupted bit.

7.3.4 Form Error in Error Delimiter

7.3.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects a Form Error when receiving an invalid Error Delimiter.

The LT replaces one of the eight Recessive bits of the Error Delimiter by a Dominant bit.

There are three Elementary Tests to perform, corrupting the second, the fourth, and the seventh bit of the Error Delimiter.

7.3.4.2 Test Case organisation

Test Case organisation see table 34.

Table 34 - 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 generate an Error Frame. The LT replaces one of the above mentioned Recessive bits of the Error Delimiter with a Dominant bit.
Verification	The IUT must restart with an Active Error Frame at the bit position following the replaced bit.

7.4 Overload Frame management class

7.4.1 MAC overload generation during Intermission Field

7.4.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT generates an Overload Frame when detecting a Dominant bit on one of the two first Recessive bits of the Intermission Field.

There are two Elementary Tests to perform.

7.4.1.2 Test Case organisation

Test Case organisation see table 35.

Table 35 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	One test frame is used for each of the two Elementary Tests. The LT forces one of the two first bits of the Intermission Field of the test frame to a Dominant value.
Verification	The IUT generates an Overload Frame at the bit position following the Dominant bit.

7.4.2 Last bit of EOF

7.4.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT generates an Overload Frame when detecting a Dominant State on the last bit of EOF.

There is one Elementary Test to perform.

7.4.2.2 Test Case organisation

Test Case organisation see table 36.

Table 36 Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT forces the last bit of the EOF to a Dominant State.
Verification	The IUT generates an Overload Frame at the bit position following the Dominant bit.

7.4.3 Eighth bit of an Error and Overload Delimiter

7.4.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT generates an Overload Frame when detecting a Dominant bit on the eighth bit of an Error and Overload Delimiter it is transmitting.

There are two Elementary Tests to perform.

7.4.3.2 Test Case organisation

Test Case organisation see table 37.

Table 37 - 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 generate an Error Frame or an Overload Frame. The LT forces to a Dominant State the eighth bit of the Delimiter.
Verification	The IUT generates an Overload Frame starting at the bit position following the Dominant bit forced by LT.

7.4.4 Bit error in Overload Flag

7.4.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects a Bit Error when one of the six Dominant bits of the Overload Flag it transmits is forced to Recessive State by LT.

There are three Elementary Tests to perform, corrupting the first, the third, and the sixth bit of the Overload Flag.

7.4.4.2 Test Case organisation

Test Case organisation see table 38.

Table 38 - 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 generate an Overload Frame. The LT forces one of the above mentioned bits of the Overload Flag to the Recessive State.
Verification	The IUT must generate an Error Frame at the bit position following the corrupted bit.

7.4.5 Form Error in Overload Delimiter

7.4.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects a Form Error when receiving an invalid Overload Delimiter.

The LT replaces one of the eight Recessive bits of the Overload Delimiter by a Dominant bit.

There are three Elementary Tests to perform, corrupting the second, the fourth, and the seventh bit of the Overload Delimiter.

7.4.5.2 Test Case organisation

Test Case organisation see table 39.

Table 39 - 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 generate an Overload Frame. The LT replaces one of the above mentioned Recessive bits of the Overload Delimiter with a Dominant bit.
Verification	The IUT generates an Error Frame starting at the bit position following the replaced bit.

7.5 Passive error state class

7.5.1 Passive Error Flag completion test 1

7.5.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an Error Passive IUT considers the Passive Error Flag as completed after the detection of six consecutive bits of the same value.

There are three Elementary Tests to perform, superimposing the Passive Error Flag by an Active Error Flag starting at the first, the third and the sixth bits.

7.5.1.2 Test Case organisation

Test Case organisation see table 40.

Table 40 - Test Case organisation

State	Description
Set-up	The IUT is set in Passive State.
Test	The LT causes the IUT to generate a Passive Error Frame. During the Passive Error Flag sent by the IUT, the LT sends an Active Error Flag. At the end of the Active Error Flag the LT waits for (8+2) bit time before sending a valid test frame.
Verification	The IUT must acknowledge the test frame.

7.5.2 Data frame acceptance after Passive Error Frame transmission

7.5.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an Error Passive IUT accepts a frame starting after the second bit of the intermission following the Error Frame it has transmitted.

There is one Elementary Test to perform.

7.5.2.2 Test Case organisation

Test Case organisation see table 41.

Table 41 - Test Case organisation

State	Description
Set-up	The IUT is set in Passive State.
Test	The LT causes the IUT to generate a Passive Error Frame. At the end of the Passive Error Flag the LT waits for (8+2) bit time before sending a valid test frame.
Verification	The IUT must acknowledge the test frame.

7.5.3 Acceptance of 7 consecutive Dominant bits after Passive Error Flag

7.5.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an Error Passive IUT does not detect any error when detecting up to 7 consecutive Dominant bits starting at the bit position following the last bit of the Passive Error Flag.

There are three Elementary Tests to perform, transmitting one, four, or seven consecutive Dominant bits.

7.5.3.2 Test Case organisation

Test Case organisation see table 42.

Table 42 - Test Case organisation

State	Description
Set-up	The IUT is set in Passive State.
Test	The LT causes the IUT to generate a Passive Error Frame. After the Passive Error Flag, the LT starts transmitting Dominant bits according to 7.5.3.1. After the Dominant bit sequence the LT waits for (8+2) bit time before sending a valid test frame.
Verification	The IUT must acknowledge the test frame.

7.5.4 Passive State unchanged on further errors

7.5.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an Error Passive IUT does not become Error Active on any error detection.

There is one Elementary Test to perform.

7.5.4.2 Test Case organisation

Test Case organisation see table 43.

Table 43 - Test Case organisation

State	Description
Set-up	The IUT is set to Passive State.
Test	The LT sends at least nine invalid test frames.
Verification	The IUT shall not generate any Active Error Frame.

7.5.5 Passive Error Flag completion

7.5.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an Error Passive IUT restarts the Passive Error Flag when detecting up to five consecutive Dominant bits during its own Passive Error Flag.

There are three Elementary Tests to perform, superimposing the Passive Error Flag by the sequence of five Dominant bits starting at the first, the third and the sixth bits of the Passive Error Flag.

7.5.5.2 Test Case organisation

Test Case organisation see table 44.

Table 44 - Test Case organisation

State	Description
Set-up	The IUT is set in Passive State.
Test	The LT causes the IUT to generate a Passive Error Frame. During the passive Error Flag sent by the IUT the LT sends a sequence of five Dominant bits according to 7.5.5.1. After this sequence, the LT waits for (6 + 7) bit time before sending a Dominant bit, corrupting the last bit of the Error Delimiter.
Verification	Following the Dominant bit sent by the LT the IUT must generate an Overload Frame.

7.5.6 Form Error in passive Error Delimiter

7.5.6.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an Error Passive IUT detects a Form Error when receiving an invalid Error Delimiter.

The LT replaces one of the eight Recessive bits of the Error Delimiter by a Dominant bit.

There are three Elementary Tests to perform, corrupting the second, the fourth, and the seventh bit of the Error Delimiter.

7.5.6.2 Test Case organisation

Test Case organisation see table 45.

Table 45 - Test Case organisation

State	Description
Set-up	The IUT is set in Passive State.
Test	The LT causes the IUT to generate a Passive Error Frame. During the Error Delimiter, the LT creates a Form Error according to according to 7.5.6.1. After the Form Error, the LT waits for (6 + 7) bit time before sending a Dominant bit, corrupting the last bit of the Error Delimiter.
Verification	The IUT must generate an Overload Frame starting at the bit position following the last Dominant bit sent by the LT.

7.6 Error counter management class

7.6.1 REC increment on Bit Error in Active Error Flag

7.6.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT increases its REC by 8, when detecting a Bit Error during the transmission of an Active Error Flag.

There are three Elementary Tests to perform, corrupting the first, the third, and the sixth bit of the Active Error Flag.

7.6.1.2 Test Case organisation

Test Case organisation see table 46.

Table 46 - 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 generate an Active Error Frame. The LT corrupts one of the above mentioned Dominant bits of the Error Flag.
Verification	The REC is increased by eight on the corrupted bit.

7.6.2 REC increment on Bit Error in Overload Flag

7.6.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT increases its REC by 8, when detecting a Bit Error during the transmission of an Overload Flag.

There are three Elementary Tests to perform, corrupting the first, the fourth, and the sixth bit of the Overload Flag.

7.6.2.2 Test Case organisation

Test Case organisation see table 47.

Table 47 - 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 generate an Overload Frame. The LT corrupts one of the above mentioned Dominant bits of the Overload Flag.
Verification	The REC is increased by eight on the corrupted bit.

7.6.3 REC increment when Active Error Flag is longer than 13 bits

7.6.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT increases its REC by 8, when detecting the eighth consecutive Dominant bit following the transmission of its Active Error Flag and after each sequence of additional eight consecutive Dominant bits.

There is one Elementary Test to perform.

7.6.3.2 Test Case organisation

Test Case organisation see table 48.

Table 48 - 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 generate an Active Error Frame. After the Error Flag sent by the IUT, the LT sends a sequence of 16 Dominant bits.
Verification	The REC is increased by eight on each eighth Dominant bit after the Error Flag.

7.6.4 REC increment when Overload Flag is longer than 13 bits

7.6.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT increases its REC by 8, when detecting the eighth consecutive Dominant bit following the transmission of its Overload Flag and after each sequence of additional eight consecutive Dominant bits.

There is one Elementary Test to perform.

7.6.4.2 Test Case organisation

Test Case organisation see table 49.

Table 49 - 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 generate an Overload Frame. After the Overload Flag sent by the IUT, the LT sends a sequence of 16 Dominant bits.
Verification	The REC is increased by eight on each eighth Dominant bit after the Overload Flag.

7.6.5 REC increment on Bit Error in the ACK Field

7.6.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT increases its REC by 1, when detecting a Bit Error on the ACK Slot it transmits.

There is one Elementary Test to perform.

7.6.5.2 Test Case organisation

Test Case organisation see table 50.

Table 50 - 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 send a Dominant acknowledgement, the ACK Slot is corrupted by LT.
Verification	The REC is increased by one on the corrupted bit.

7.6.6 REC increment on Form Error at CRC Delimiter

7.6.6.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT increases its REC by 1, when detecting a Form Error at CRC Delimiter.

There is one Elementary Test to perform.

7.6.6.2 Test Case organisation

Test Case organisation see table 51.

Table 51 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends a frame with the CRC Delimiter changed to a Dominant value.
Verification	The REC is increased by one on the Dominant CRC Delimiter.

7.6.7 REC increment on Form Error at ACK Delimiter

7.6.7.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT increases its REC by 1, when detecting a Form Error on ACK Delimiter.

There is one Elementary Test to perform.

7.6.7.2 Test Case organisation

Test Case organisation see table 52.

Table 52 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends a frame with the ACK Delimiter changed to a Dominant value.
Verification	The REC is increased by one on the Dominant ACK Delimiter.

7.6.8 REC increment on Form Error in EOF Field

7.6.8.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT increases its REC by 1, when detecting a Form Error on the EOF Field during reception of a data frame.

There are three Elementary Tests to perform, corrupting the second, the third, and the fifth bit of the EOF.

7.6.8.2 Test Case organisation

Test Case organisation see table 53.

Table 53 - Test Case organisation

State	**Description**
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends a frame with the EOF modified according to 7.6.8.1.
Verification	The REC is increased by one on the replaced bit of the EOF.

7.6.9 REC increment on Stuff Error

7.6.9.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT increases its REC by 1, when detecting a Stuff Error.

There are eight Elementary Tests to perform, testing both errors on Recessive and Dominant Stuff Bits in the arbitration, control, data, and CRC field.

7.6.9.2 Test Case organisation

Test Case organisation see table 54.

Table 54 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends a sequence of six consecutive bits according to 7.6.9.1.
Verification	The REC is increased by one on the sixth consecutive bit.

7.6.10 REC increment on CRC Error

7.6.10.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT increases its REC by 1, when detecting a CRC Error during reception of a frame.

There is one Elementary Test to perform.

7.6.10.2 Test Case organisation

Test Case organisation see table 55.

Table 55 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends a frame containing a CRC Error.
Verification	The IUT sends a Recessive acknowledge. The IUT starts the transmission of an Active Error Frame after the ACK Delimiter. The REC is increased by one after the ACK Delimiter

7.6.11 REC increment on Dominant bit after end of Error Flag

7.6.11.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an Error Active IUT increases its REC by 8, when detecting a Dominant bit as the first bit after sending an Error Flag.

There is one Elementary Test to perform.

7.6.11.2 Test Case organisation

Test Case organisation see table 56.

Table 56 - 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 generate an Active Error Flag. The LT sends a Dominant bit at the bit position following the end of the Error Flag sent by the IUT.
Verification	The REC is increased by eight after reception of the Dominant bit sent by the LT.

7.6.12 REC increment on Form Error in Error Delimiter

7.6.12.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that a receiver increases its REC by 1, when detecting a Form Error on a bit of the Error Delimiter it is transmitting.

There are two Elementary Tests to perform, the second, and the last bit of the Error Delimiter are corrupted.

7.6.12.2 Test Case organisation

Test Case organisation see table 57.

Table 57 - 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 generate an Active Error Frame. The LT corrupts one bit of the Error Delimiter according to 7.6.12.1.
Verification	The REC is increased by one after reception of the Dominant bit sent by the LT.

7.6.13 REC increment on Form Error in Overload Delimiter

7.6.13.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that a receiver increases its REC by 1, when detecting a Form Error on a bit of the Overload Delimiter it is transmitting.

There are two Elementary Tests to perform, the second, and the last bit of the Overload Delimiter are corrupted.

7.6.13.2 Test Case organisation

Test Case organisation see table 58.

Table 58 - 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 generate an Overload Frame. The LT corrupts one bit of the Overload Delimiter according to 7.6.13.1.
Verification	The REC is increased by one after reception of the Dominant bit sent by the LT.

7.6.14 REC decrement on valid frame reception

7.6.14.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT decreases its REC by 1, when receiving a valid frame.

There is one Elementary Test to perform.

7.6.14.2 Test Case organisation

Test Case organisation see table 59.

Table 59 - Test Case organisation see table

State	Description
Set-up	The LT forces the IUT to increase its REC.
Test	The LT sends one valid test frame.
Verification	The REC is decreased by one after the successful transmission of the ACK Slot.

7.6.15 REC decrement on valid frame reception during Passive State

7.6.15.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT set its REC to a value between 119 and 127, when receiving a valid frame while being Error Passive.

There is one Elementary Test to perform.

7.6.15.2 Test Case organisation

Test Case organisation see table 60.

Table 60 - Test Case organisation

State	Description
Set-up	The LT causes the IUT’s REC value to be at Error Passive level.
Test	The LT sends one valid test frame.
Verification	The REC must be decremented to a value between 119 and 127 after the successful transmission of the ACK Slot.

7.6.16 REC non-increment on last bit of EOF Field

7.6.16.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT does not change the value of its REC, when detecting a Dominant bit at the last bit of the EOF it is receiving.

This test also verifies that the REC is not decremented below zero.

There is one Elementary Test to perform.

7.6.16.2 Test Case organisation

Test Case organisation see table 61.

Table 61 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends one valid test frame with a Dominant bit at the last bit of EOF.
Verification	The REC value is zero.

7.6.17 REC non-increment on 13-bit length Overload Flag

7.6.17.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT does not change the value of its REC when receiving a 13-bit length Overload Flag.

There is one Elementary Test to perform.

7.6.17.2 Test Case organisation

Test Case organisation see table 62

Table 62 - 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 generate an Overload Frame. After the Overload Flag sent by the IUT, the LT sends a sequence of 7 Dominant bits.
Verification	The REC value is zero.

7.6.18 REC non-increment on 13-bit length Error Flag

7.6.18.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT does not increase its REC after the 7th bit of the received Error Flag.

There is one Elementary Test to perform.

7.6.18.2 Test Case organisation

Test Case organisation see table 63.

Table 63 - 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 generate an Active Error Frame. After the Error Flag sent by the IUT, the LT sends additional 7 consecutive Dominant bits.
Verification	The REC is not further incremented after the increment due to the Dominant bit which followed the Error Flag sent by the IUT.

7.6.19 REC non-increment on last bit of Error and Overload Delimiter

7.6.19.1 Purpose and limits of this Test Case

This test verifies that the IUT does not change the value of its REC when detecting a Dominant bit at the last bit of an Error and Overload Delimiter it is transmitting.

There are two Elementary Tests to perform.

7.6.19.2 Test Case organisation

Test Case organisation see table 64.

Table 64 - 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 generate an Error or an Overload Frame. The LT corrupts the last bit of the Error or the Overload Delimiter.
Verification	The REC value is zero.

7.7 Bit timing class

7.7.1 Sample point test

7.7.1.1 Purpose and limits of this Test Case

CAN_VERSION $\in$ {A, B, BP}

The purpose of this test is to verify the position of the sample point of an IUT.

There is one Elementary Test to perform.

7.7.1.2 Test Case organisation

Test Case organisation see table 65.

Table 65 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT shortens a Dominant Stuff Bit by an amount of Phase_Seg2 and then later shortens another Dominant Stuff Bit by an amount of (Phase_Seg2 + 1TQ).
Verification	The IUT must generate an Error Frame on the bit position following the second shortened Stuff Bit.

7.7.2 Hard synchronisation on SOF reception

7.7.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that the IUT makes a hard synchronisation when receiving a SOF delayed by e, e ∈ [1, NTQ].

There are NTQ Elementary Tests to perform.

7.7.2.2 Test Case organisation

Test Case organisation see table 66.

Table 66 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends a first test frame and after the second bit of the Intermission Field it sends a SOF delayed by e time quanta depending on the Elementary Test. The SOF is followed by a sequence of five Dominant bits.
Verification	The IUT must respond with an Error Frame exactly six bit times after the Recessive to Dominant edge at the beginning of the SOF.

7.7.3 Synchronisation when e > 0 and e ≤ SJW

7.7.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify the behaviour of an IUT detecting a positive phase error (e) on a Recessive to Dominant edge with e≤SJW.

The value tested for e are in Time Quantum with e ∈ [1, SJW].

There is one Elementary Test to perform for each possible value of e.

7.7.3.2 Test Case organisation

Test Case organisation see table 67.

Table 67 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT delays a Dominant Stuff Bit by an amount of e time quanta and shortens the same bit by an amount of (Phase_Seg2 + 1TQ - e).
Verification	The IUT must generate an Error Frame 1 bit time after the Recessive to Dominant edge of the delayed Stuff Bit.

7.7.4 Synchronisation when e > 0 and e > SJW

7.7.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify the behaviour of an IUT detecting a positive phase error (e) on a Recessive to Dominant edge with e>SJW.

The value tested for e are in Time Quantum with e ∈ [(SJW + 1), (NTQ - (Phase_Seg2 + 1))].

There is one Elementary Test to perform for each possible value of e.

7.7.4.2 Test Case organisation

Test Case organisation see table 68.

Table 68 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT delays a Dominant Stuff Bit by an amount of e time quanta and shortens the same bit by an amount of (Phase_Seg2 + 1TQ - e).
Verification	The IUT must generate an Error Frame 1 bit time + (e - SJW) time quanta after the Recessive to Dominant edge of the delayed Stuff Bit.

7.7.5 Synchronisation when e < 0 and |e| ≤ SJW

7.7.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify the behaviour of an IUT detecting a negative phase error (e) on a Recessive to Dominant edge with |e|≤SJW.

The value tested for e are in Time Quantum with |e| ∈ [1, SJW].

There is one Elementary Test to perform for each possible value of e.

7.7.5.2 Test Case organisation

Test Case organisation see table 69.

Table 69 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT shortens the last Recessive bit before an expected Dominant Stuff Bit by an amount of |e| time quanta and then sends a Dominant value for one Time Quantum followed by a Recessive State.
Verification	The IUT must generate an Error Frame 1 bit-time after the last Recessive to Dominant edge.

7.7.6 Synchronisation when e < 0 and |e| > SJW

7.7.6.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify the behaviour of an IUT detecting a negative phase error (e) on a Recessive to Dominant edge with |e|>SJW.

The value tested for e are in Time Quantum with |e| ∈ [(SJW + 1), Phase_Seg2].

There is one Elementary Test to perform for each possible value of e.

7.7.6.2 Test Case organisation

Test Case organisation see table 70.

Table 70 - Test Case organisation see table

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT shortens the last Recessive bit before an expected Dominant Stuff Bit by an amount of |e| time quanta and then sends a Dominant value for one Time Quantum followed by a Recessive State.
Verification	The IUT must generate an Error Frame 1 bit-time - (|e| - SJW) time quanta after the last Recessive to Dominant edge.

7.7.7 Glitch filtering test on positive phase error

7.7.7.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that there is only one synchronisation within one bit time if there are two Dominant to Recessive edges between synchronisation segment and sample point.

The test also verifies that an IUT is able to synchronise on a minimum duration pulse obeying to the synchronisation rules.

There is one Elementary Test to perform.

7.7.7.2 Test Case organisation

Test Case organisation see table 71.

Table 71 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends a frame containing a Dominant Stuff Bit. After the first two time quanta of Dominant value it changes one Time Quantum to Recessive value. This Dominant Stuff Bit is followed by six Recessive bits.
Verification	The IUT must respond with an Error Frame exactly seven bit times after the first Recessive to Dominant edge of the Stuff Bit.

7.7.8 Glitch filtering test on negative phase error

7.7.8.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that there is only one synchronisation within one bit time if there are two Dominant to Recessive edges between two sample points where the first edge comes before the Synchronisation segment.

The test also verifies that an IUT is able to synchronise on a minimum duration pulse obeying to the synchronisation rules.

There is one Elementary Test to perform.

7.7.8.2 Test Case organisation

Test Case organisation see table 72.

Table 72 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends a frame containing a Dominant Stuff Bit. The Recessive bit before the Stuff Bit is shortened by one Time Quantum. After the first two time quanta of Dominant value it changes one Time Quantum to Recessive value. This Dominant Stuff Bit is followed by six Recessive bits.
Verification	The IUT must respond with an Error Frame exactly seven bit times after the first Recessive to Dominant edge of the Stuff Bit.

7.7.9 Glitch filtering test in Idle State

7.7.9.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an IUT shall not detect a SOF when detecting a Dominant level shorter than (propagation segment + phase buffer segment 1 - 1 TQ).

There is one Elementary Test to perform.

7.7.9.2 Test Case organisation

Test Case organisation see table 73.

Table 73 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends a Dominant glitch according to 7.7.8.1 Purpose and limits of this Test Case. Then the LT waits for 8 bit times.
Verification	The IUT must remain in the Idle State..

7.7.10 Non-Re-synchronisation after a Dominant sampled bit

7.7.10.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that no edge shall be used for synchronisation if the value detected at the previous sample point is the same as the bus value immediately after the edge.

There is one Elementary Test to perform.

7.7.10.2 Test Case organisation

Test Case organisation see table 74.

Table 74 - Test Case organisation

State	Description
Set-up	No action required, the IUT is left in the Default State.
Test	The LT sends a frame containing a Dominant Stuff Bit. At the position (NTQ - Phase_Seg2 + 1) time quanta after the edge at the beginning of the Stuff Bit, the LT changes the value to Recessive for one Time Quantum. The Stuff Bit is followed by five additional Dominant bits.
Verification	The IUT must respond with an Error Frame exactly six bit times after the Recessive to Dominant edge at the beginning of the Stuff Bit.

8 Transmitted frame type

8.1 Valid frame format class

8.1.1 Identifier and number of data bytes test in standard format

8.1.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

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

Tested identifiers:	IDEN ∈ [000h, 7EFh] ∪ [7F0h, 7FFh],
Tested number of data bytes:	∈ [0, 8].

8.1.1.2 Test Case organisation

Test Case organisation see table 75.

Table 75 - 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 data frame with the parameters listed in 8.1.1.1.
Verification	The IUT shall not generate any Error Flag during the test. The content of the frame must match the LT request. The number of data bytes must match the DLC if the DLC is less than 9. No more than 8 data bytes may be transmitted.

8.1.2 Identifier and number of data bytes test in extended format

8.1.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {B}

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

Tested identifiers: IDEN ∈ [0, 1FFFFFFFh]

Tested number of data bytes: ∈ [0, 8]

8.1.2.2 Test Case organisation

Test Case organisation see table 76.

Table 76 - 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 data frame with the parameters listed in 8.1.2.1.
Verification	The IUT shall not generate any Error Flag during the test. The content of the frame must match the LT request. The number of data bytes must match the DLC if the DLC is less than 9. No more than 8 data bytes may be transmitted.

8.1.3 Arbitration in standard format frame

8.1.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP},

This test verifies the capability of the IUT to manage the arbitration mechanism on every bit position in a standard format frame it is transmitting.

For an OPEN device, the cases specified in table 77 are tested.

Table 77 - Cases to be tested for an OPEN device

Transmitted frame			Description of the concerned arbitration bit(s)	Number of Elementary Test
IDEN	RTR	DATA Field		
7EFh	0	No data	Collision on all bits equal to 1	10
010h	1	No data	Collision on all bits equal to 1	2

For a SPECIFIC device all possible cases of transmitting a Recessive arbitration bit shall be considered.

There are at most 12 Elementary Tests to perform.

8.1.3.2 Test Case organisation

Test Case organisation see table 78.

Table 78 - 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 a Recessive bit in the Arbitration Field to the Dominant State according to the table in 8.1.3.1 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.

8.1.4 Arbitration in extended format frame test

8.1.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {B}

This test verifies the capacity of the IUT to manage the arbitration mechanism on every bit position in an extended format frame it is transmitting.

For an OPEN device, the cases in table 79 are tested.

Table 79 - Cases to be tested for an OPEN device

Transmitted frame			Description of the concerned bit(s)	Number of Elementary Test
IDEN	RTR	DATA Field		
1FBFFFFFh	0	No data	Collision on all bits equal to 1	28
00400000h	1	No data	Collision on all bits equal to 1	2
00400000h	0	No data	Collision on SRR and IDE bit	2

For a SPECIFIC device all possible cases of transmitting a Recessive arbitration bit shall be considered.

There are at most 32 Elementary Tests to perform.

8.1.4.2 Test Case organisation

Test Case organisation see table 80.

Table 80 - 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. The LT forces a Recessive bit in the Arbitration Field to the Dominant State according to the table in 8.1.4.1 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.

8.1.5 Message validation

8.1.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify the point of time at which a message transmitted by the IUT is taken to be valid.

There is one Elementary Test to perform.

8.1.5.2 Test Case organisation

Test Case organisation see table 81.

Table 81 - 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 data frame. On the first bit of the Intermission Field of the frame sent by the IUT, the LT starts a SOF making the IUT generating an Overload Frame.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must not restart any frame after the Overload Frame.

8.1.6 Stuff Bit generation capability in standard frame

8.1.6.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an IUT correctly generates the Stuff Bits in a standard frame.

For an OPEN device, the cases in table 82 are tested:

Table 82 – Cases to be tested for an OPEN device

Frame	IDEN	RTR	CTRL	DATA
1	078h	0	8	01h, E1h, E1h, E1h, E1h, E1h, E1h, E1h
2	41Fh	0	1	00
3	47Fh	0	01h	1Fh
4	758h	0	00h	-
5	777h	0	01h	1Fh
6	7EFh	1	02h	-

For a SPECIFIC device all possible Dominant and Recessive Stuff Bits inside and at the end of each stuffed field shall be considered.

There are six Elementary Tests to perform.

8.1.6.2 Test Case organisation

Test Case organisation see table 83.

Table 83 - 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 according to the table in 8.1.6.1.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must correctly generate all Stuff Bits.

8.1.7 Stuff Bit generation capability in extended frame

8.1.7.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {B}

The purpose of this test is to verify that an IUT correctly generates the Stuff Bits in an extended frame.

For an OPEN device, the cases in table 84 are tested:

Table 84 – Cases to be tested for an OPEN device

Frame	IDEN 1	IDEN 2	RTR	CTRL	DATA
1	1F0h	30F0Fh	0	8	3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch
2	1F0h	0F0F0h	0	1	00
3	7EEh	0	0	01h	A0h

There are three Elementary Tests to perform.

For a SPECIFIC device all possible Dominant and Recessive Stuff Bits inside and at the end of each stuffed field shall be considered.

8.1.7.2 Test Case organisation

Test Case organisation see table 85.

Table 85 - 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 according to the table in 8.1.7.1.
Verification	The IUT shall not generate any Error Flag during the test. The IUT must correctly generate all Stuff Bits.

8.2 Error detection class

8.2.1 Bit Error in standard frame test

8.2.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects a Bit Error when the bit it is transmitting in a standard frame is different with the bit it receives.

The test shall be at minimum to modify at least one Dominant and one Recessive bit in each field of the frame except for the Arbitration Field for which only Dominant bits shall be modified. The ACK Slot is not tested.

8.2.1.2 Test Case organisation

Test Case organisation see table 86.

Table 86 - 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 the frames and creates a Bit Error according to 8.2.1.1.
Verification	The IUT must generate an Active Error Frame starting at the bit position following the corrupted bit. The IUT must restart the transmission of the data frame as soon as the bus is idle.

8.2.2 Bit Error in extended frame test

8.2.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {B}

This test verifies that the IUT detects a Bit Error when the bit it is transmitting in an extended frame is different with the bit it receives.

The test shall be to modify at least one Dominant extended identifier bit and the two reserved bits.

8.2.2.2 Test Case organisation

Test Case organisation see table 87.

Table 87 - 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 the frames and creates a Bit Error according to 8.2.2.1.
Verification	The IUT must generate an Active Error Frame starting at the bit position following the corrupted bit. The IUT must restart the transmission of the data frame as soon as the bus is idle.

8.2.3 Stuff Error test in standard frame

8.2.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects an error when after the transmission of five identical bits it receives a sixth bit identical to the five precedents. This test is made of a standard format frame.

The frames in table 88 shall be used to do the Elementary Tests:

Table 88 – Frames for Elementary Tests

N° frame	IDEN	RTR	CTRL	DATA
1	078h	0	8	01h, E1h, E1h, E1h, E1h, E1h, E1h, E1h
2	41Fh	0	1	00
3	47Fh	0	01h	1Fh
4	758h	0	00h	-
5	777h	0	01h	1Fh
6	7EFh	1	02h	-

For an OPEN device, at least one Stuff error shall be generated at each stuffed field.

For a SPECIFIC device at least one Stuff error shall be generated at each stuffed field, where a Stuff Bit can occur.

8.2.3.2 Test Case organisation

Test Case organisation see table 89.

Table 89 - 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 the frames and create a Stuff Error according to 8.2.3.1.
Verification	The IUT must generate an Error Frame at the bit position following the corrupted Stuff Bit. The IUT must restart the transmission of the data frame as soon as the bus is idle.

8.2.4 Stuff Error test in extended frame

8.2.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {B}

This test verifies that the IUT detects an error when after the transmission of five identical bits it receives a sixth bit identical to the five precedents. This test is made of an extended frame.

The frames in table 90 shall be used to do the Elementary Tests.

Table 90 - Frames used at Elementary Tests

N°	IDEN 1	IDEN 2	RTR	CTRL	DATA
1	1F0h	30F0Fh	0	8	3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch, 3Ch
2	1F0h	0F0F0h	0	1	00
3	7EEh	0	0	01h	A0h

For an OPEN device, at least one Stuff error shall be generated at each stuffed field.

For a SPECIFIC device at least one Stuff error shall be generated at each stuffed field, where a Stuff Bit can occur.

8.2.4.2 Test Case organisation

Test Case organisation see table 91.

Table 91 - 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 the frames and create a Stuff Error according to 8.2.4.1.
Verification	The IUT must generate an Error Frame at the bit position following the corrupted Stuff Bit. The IUT must restart the transmission of the data frame as soon as the bus is idle.

8.2.5 Form Error

8.2.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that the IUT detects a Form Error if one of the following fields, transmitted by the IUT, contains a Dominant bit:

- CRC Delimiter,
- ACK Delimiter,
- EOF (the first, the fourth and the last one)

There are five Elementary Tests to perform.

8.2.5.2 Test Case organisation

Test Case organisation see table 92.

Table 92 - 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 creates a Form Error on the fields listed in 8.2.5.1 by corrupting the Recessive bit of these fields.
Verification	The IUT must generate an Error Frame at the bit position following the corrupted bit. The IUT must restart the transmission of the frame as soon as the bus is idle.

8.2.6 Acknowledgement Error

8.2.6.1 Purpose and limits of this Test Case

CAN_VERSION $\in$ {A, B, BP}

This test verifies that the IUT detects an Acknowledgement Error when the received ACK Slot is recessive.

There is one Elementary Test to perform.

8.2.6.2 Test Case organisation

Test Case organisation see table 93.

Table 93 - 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 standard frame. Then the LT does not send a Dominant bit in the ACK Slot.
Verification	The IUT must generate an Error Frame starting at the bit position following the ACK Slot. The IUT must restart the transmission of the frame as soon as the bus is idle.

8.3 Error Frame management class

8.3.1 Error Flag longer than 6 Bits

8.3.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter tolerates up to seven Dominant bits after sending its own Error Flag.

There are three Elementary Tests to perform, lengthening the Error Flag by one, four, and seven Dominant bits.

8.3.1.2 Test Case organisation

Test Case organisation see table 94.

Table 94 - 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. The LT corrupts this frame causing the IUT to send an Active Error Frame. The LT prolongs the Error Flag send by IUT according to 8.3.1.1.
Verification	The IUT must generate only one Error Frame. The IUT must restart the transmission after the Intermission Field following the Error Frame.

8.3.2 Transmission on the third bit of Intermission Field

8.3.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an IUT is able to transmit a frame on reception of a SOF starting at the third bit of the Intermission Field following the Error Frame it has transmitted.

There is one Elementary Test to perform.

For OPEN devices, the identifier must start with four Dominant bits.

For a SPECIFIC device which can not send such an identifier, any other value may be used.

8.3.2.2 Test Case organisation

Test Case organisation see table 95.

Table 95 - 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 according to 8.3.2.1. The LT corrupts this frame causing the IUT to send an Active Error Frame. At the end of the Error Flag sent by the IUT, the LT waits for (8+2) bit times before sending a SOF.
Verification	The IUT must repeat the frame starting with the identifier without transmitting any SOF.

8.3.3 Bit Error in Error Flag

8.3.3.1 Purpose and limits of this Test Case

CAN_VERSION ∊ {A, B, BP}

This test verifies that an IUT acting as a transmitter detects a Bit Error when one of the six Dominant bits of the Error Flag it transmits is forced to Recessive State by LT.

There are three Elementary Tests to perform, corrupting the first, the fourth and the sixth bit of the Error Flag.

8.3.3.2 Test Case organisation

Test Case organisation see table 96.

Table 96 - 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. The LT corrupts this frame causing the IUT to send an Active Error Frame. Then the LT forces one of the six bits of the Active error flag sent by the IUT to Recessive State.
Verification	The IUT must restart its Active Error Flag at the bit position following the corrupted bit.

8.3.4 Form Error in Error Delimiter

8.3.4.1 Purpose and limits of this Test Case

CAN_VERSION ∊ {A, B, BP}

This test verifies that an IUT acting as a transmitter detects a Form Error when it receives an invalid Error Delimiter.

The LT replaces one of the eight Recessive bits of the Error Delimiter by a Dominant bit.

There are three Elementary Tests to perform, corrupting the second, the fourth and the seventh bit of the Error Delimiter.

8.3.4.2 Test Case organisation

Test Case organisation see table 97.

Table 97 - 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. The LT corrupts this frame causing the IUT to send an Active Error Frame. Then the LT forces one Recessive bit of the Error Delimiter to the Dominant State according to 8.3.4.1.
Verification	The IUT must restart the Error Frame at the bit position following the corrupted bit.

8.4 Overload Frame management class

8.4.1 MAC Overload generation in Intermission Field

8.4.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter generates an Overload Frame when detecting a Dominant bit on one of the two first Recessive bits of the Intermission Field following a data frame it is transmitting.

There are two Elementary Tests to perform.

8.4.1.2 Test Case organisation

Test Case organisation see table 98.

Table 98 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. Then the LT forces one of the first two bits of the Intermission Field to the Dominant State.
Verification	The IUT must generate an Overload Frame starting at the bit position following the Dominant bit generated by the LT.

8.4.2 Eighth bit of an Error and Overload Delimiter

8.4.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter generates an Overload Frame when detecting a Dominant bit on the eighth bit of an Error or an Overload Delimiter it is transmitting.

There are two Elementary Tests to perform.

8.4.2.2 Test Case organisation

Test Case organisation see table 99.

Table 99 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. Then the LT causes the IUT to generate an Error Frame or Overload Frame. Then the LT forces the eighth bit of the Delimiter to a Dominant State.
Verification	The IUT must generate an Overload Frames starting at the bit position following the Dominant bit sent by the LT.

8.4.3 Transmission on the third Bit of Intermission Field

8.4.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an IUT is able to transmit a data frame starting with the identifier and without transmitting SOF, when detecting a Dominant bit on the third bit of the Intermission Field following an Overload Frame.

There is one Elementary Test to perform.

For OPEN devices, the identifier must start with four Dominant bits.

For a SPECIFIC device which can not send such an identifier, any other value may be used.

8.4.3.2 Test Case organisation

Test Case organisation see table 100.

Table 100 - 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 according to 8.4.3.1. Then the LT causes the IUT to generate an Overload Frame. Then the LT forces the third bit of the Intermission following the Overload Delimiter to Dominant State.
Verification	The IUT must repeat the frame starting with the identifier without transmitting any SOF.

8.4.4 Bit Error in Overload Flag

8.4.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter detects a Bit Error when one of the six Dominant bits of the Overload Flag it transmits is forced to Recessive State by LT.

There are three Tests to perform, corrupting the first, the second, and the sixth bit of the Overload Flag.

8.4.4.2 Test Case organisation

Test Case organisation see table 101.

Table 101 - 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 causes the IUT to generate an Overload Frame. Then the LT corrupts one of the six Dominant bits of the Overload Flag to the Recessive State according to 8.4.4.1.
Verification	The IUT must generate an Error Frame starting at the bit position after the corrupted bit.

8.4.5 Form Error in Overload Delimiter

8.4.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter detects a Form Error when it receives an invalid Overload Delimiter.

The LT replaces one of the eight Recessive bits of the Overload Delimiter by a Dominant bit.

There are three Elementary Tests to perform, corrupting the second, the fourth and the seventh bit of the Overload Delimiter.

8.4.5.2 Test Case organisation

Test Case organisation see table 102.

Table 102 - 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 causes the IUT to generate an Overload Frame. The LT corrupts the Overload Delimiter according to 8.4.5.1.
Verification	The IUT must generate an Error Frame starting at the bit position following the corrupted bit.

8.5 Passive error state and bus-off class

8.5.1 Acceptance of Active Error Flag overwriting Passive Error Flag

8.5.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that a Passive State IUT acting as a transmitter does not detect any error when detecting an Active Error Flag during its own Passive Error Flag.

There are three Elementary Tests to perform, superposing the Passive Error Flag by an Active Error Flag starting at the first, the third and the sixth bit.

8.5.1.2 Test Case organisation

Test Case organisation see table 103.

Table 103 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. Then the LT causes the IUT to send a passive Error Flag. During the Passive Error Flag sent by the IUT, the LT sends an Active Error Flag according to 8.5.1.1. At the end of the Error Flag the LT waits for (8+3) bit time before sending a frame.
Verification	The IUT must acknowledge the last frame transmitted by the LT.

8.5.2 Frame acceptance after Passive Error Frame transmission

8.5.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that a Passive State IUT acting as a transmitter accepts to receive a frame starting after the second bit of the intermission following the Error Frame it has transmitted.

There is one Elementary Test to perform:

8.5.2.2 Test Case organisation

Test Case organisation see table 104.

Table 104 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. Then the LT causes the IUT to send a passive Error Flag. During the Passive Error Flag sent by the IUT, the LT sends an Active Error Flag. At the end of the Error Flag the LT waits for (8+2) bit time before sending a frame.
Verification	The IUT must acknowledge the frame.

8.5.3 Acceptance of 7 consecutive Dominant bits after Passive Error Flag

8.5.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that a Passive State IUT acting as a transmitter does not detect any error when detecting Dominant bits during the seven first bit of the Error Delimiter.

There are three Elementary Tests to perform, transmitting one, four or seven consecutive Dominant bits.

8.5.3.2 Test Case organisation

Test Case organisation see table 105.

Table 105 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a data frame. Then the LT causes the IUT to send a passive Error Flag. At the end of Error Flag, the LT continues transmitting Dominant bits according to 8.5.3.1. At this step the LT waits for (8+3) bit time before sending a frame.
Verification	The IUT must acknowledge the frame transmitted by the LT. The IUT must restart the transmission of the corrupted frame (1+7+3+8) bit time after its ACK bit.

8.5.4 Reception of a frame during Suspend Transmission Field

8.5.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that a Passive State IUT acting as a transmitter is able to receive a frame during the Suspend Transmission Field.

There are three Elementary Tests to perform, for which the received frame starts on the first, the fourth and the eighth bit of the Suspend Transmission Field.

8.5.4.2 Test Case organisation

Test Case organisation see table 106.

Table 106 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. At the end of the EOF and Intermission Fields the LT sends a frame according to 8.5.4.1.
Verification	The IUT must acknowledge the last frame transmitted by the LT.

8.5.5 Transmission of a frame after Suspend Transmission Field test 1

8.5.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that a Passive State IUT acting as a transmitter does not transmit any frame before the end of the Suspend Transmission Field following an Error Frame.

There is one Elementary Test to perform.

8.5.5.2 Test Case organisation

Test Case organisation see table 107.

Table 107 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. Then the LT causes the IUT to send a Passive Error Frame.
Verification	After the Intermission following the Error Frame, the LT verifies that the IUT waits eight more bits before re-transmitting the frame.

8.5.6 Transmission of a frame after Suspend Transmission Field test 2

8.5.6.1 Purpose and limits of this Test Case

CAN_VERSION ∊ {A, B, BP}

The purpose of this test is to verify that a Passive State IUT being transmitter does not transmit any data frame before the end of the Suspend Transmission field following an Overload Frame.

There is one Elementary Test to perform.

8.5.6.2 Test Case organisation

Test Case organisation see table 108.

Table 108 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit 2 frames. The LT lets the IUT transmit the first frame and causes the IUT to generate an Overload Frame.
Verification	After the Intermission following the Overload Frame, the LT verifies that the IUT waits eight more bits before transmitting the second frame.

8.5.7 Transmission of a frame after Suspend Transmission Field test 3

8.5.7.1 Purpose and limits of this Test Case

CAN_VERSION ∊ {A, B, BP}

The purpose of this test is to verify that a Passive State IUT acting as a transmitter does not transmit any frame before the end of the Suspend Transmission Field following a frame.

There is one Elementary Test to perform.

8.5.7.2 Test Case organisation

Test Case organisation see table 109.

Table 109 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit 2 frames. The LT lets the IUT transmit the first frame. This frame must end with EOF Field followed by the Intermission Field.
Verification	After the Intermission following the first frame, the LT verifies that the IUT waits eight more bits before transmitting the second frame.

8.5.8 Transmission of a frame without Suspend Transmission Field

8.5.8.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that a Passive State IUT, after loosing arbitration, repeats the frame without inserting any Suspend Transmission Field.

There is one Elementary Test to perform.

8.5.8.2 Test Case organisation

Test Case organisation see table 110.

Table 110 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. The LT causes the IUT to loose arbitration by sending a frame of higher priority.
Verification	The LT verifies that the IUT re-transmits its frame (1+7+3) bit times after acknowledging the received frame.

8.5.9 No transmission of a frame on the third bit of Intermission Field

8.5.9.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that a Passive State IUT does not transmit a frame starting with an identifier and without transmitting SOF, when detecting a Dominant bit on the third bit of the Intermission Field.

There is one Elementary Test to perform.

8.5.9.2 Test Case organisation

Test Case organisation see table 111.

Table 111 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit 2 frames. The LT lets the IUT transmit the first frame. This frame must end with EOF Field followed by the Intermission Field. At the third bit of the Intermission Field, the LT starts sending a frame with the lowest priority.
Verification	The IUT must not starts the second transmission before the end of the frame sent by the LT.

8.5.10 Bus-off State

8.5.10.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an IUT switching to Bus-off State no longer sends Dominant bits.

There is one Elementary Test to perform.

8.5.10.2 Test Case organisation

Test Case organisation see table 112.

Table 112 - 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 twice. The LT causes the IUT to generate an Error Frame. During the Error Flag transmitted by the IUT, the LT forces Recessive State during 16 bit times and then Dominant State for 112 bit times. Then the IUT transmits its first frame. The LT acknowledges the frame and immediately causes the IUT to generate an Overload Frame. The LT forces the first bit of this Overload Flag to Recessive State creating a Bit Error. (6+7) bit times later, the LT generates a Dominant bit to cause the IUT to generate a new Overload Frame; the LT forces the first bit of this new Overload Flag to Recessive State causing the IUT to increment its TEC to the Bus-off limit. (6+8+3+8) bit times later, the LT sends a frame.
Verification	Only one frame must be transmitted by the IUT. The IUT must not acknowledge the frame sent by the LT.

8.5.11 Bus-off recovery

8.5.11.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an IUT which is bus-off is not permitted to become Error Active (no longer bus off) before 128 occurrences of 11 consecutive Recessive bits.

There are two Elementary Tests to perform. In a first Elementary Test the LT sends Recessive bus level for at least 1408 bit times until the IUT becomes Active again. In a second Elementary Test the LT sends 1 group of 10 Recessive bits, 1 group of 21 Recessive bits followed by at least 127 groups of 11 Recessive bits, each group separated by one Dominant bit.

8.5.11.2 Test Case organisation

Test Case organisation see table 113.

Table 113 - Test Case organisation

State	Description
Set-up	No action required. The IUT is left in the Default State.
Test	The LT ask the IUT to send a frame and sets it in the Bus-off State . The LT sends the profiles defined in 8.5.11.1.
Verification	The IUT must not transmit the frame before the end of the profiles sent by the LT and must send it before the end of the TIMEOUT.

8.5.12 Completion condition for a Passive Error Flag

8.5.12.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that a Passive State IUT acting as a transmitter waits for 6 consecutive identical bit to complete its Passive Error Flag.

There is only one Elementary Test to perform.

8.5.12.2 Test Case organisation

Test Case organisation see table 114.

Table 114 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. Then the LT causes the IUT to generate a Passive Error Flag. During the Error Flag, the LT sends 5 Dominant bits, 5 Recessive bits and then 6 Dominant bits. After the 6 Dominant bits, the LT waits for 8 bit time before sending a Dominant bit.
Verification	The IUT must generate an Overload Frame starting at the bit position following the last Dominant bit generated by the LT.

8.5.13 Form Error in passive Error Delimiter

8.5.13.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an Error Passive IUT acting as a transmitter detects a Form Error when monitoring a corruption in the Error Delimiter.

The LT forces one of the eight Recessive bits of the Error Delimiter to a Dominant State.

There are three Elementary Tests to perform, corrupting the second, the fourth, and the seventh bit of the

Error Delimiter.

8.5.13.2 Test Case organisation

Test Case organisation see table 115.

Table 115 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a data frame. Then the LT causes the IUT to generate a Passive Error Frame. The LT creates a Form Error according to 8.6.13.1 After the Form Error, the LT waits for (6 + 7) bit time before sending a Dominant bit.
Verification	The IUT must generate an Overload Frame starting at the bit position following the last Dominant bit generated by the LT.

8.5.14 Maximum recovery time after a corrupted frame

8.5.14.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that the recovery time of an Error Passive IUT detecting an error is at most 31 bit times.

There is one elementary test to perform.

8.5.14.2 Test Case organisation

Test Case organisation see table 116.

Table 116 - Test Case organisation

State	Description
Set-up	The IUT is set to the Passive State.
Test	The LT causes the IUT to transmit a frame. Then the LT corrupts a dominant bit of this frame, causing the IUT to generate a Passive Error Flag. At the bit position following the end of the Passive Error Flag, the LT starts to send 6 Dominant bits.
Verification	The IUT must re-transmit the same frame 31 bit times after the detection of the corrupted bit.

8.6 Error counter management class

8.6.1 TEC increment on Bit Error during Active Error Flag

8.6.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter increases its TEC by 8 when detecting a Bit Error during the transmission of an Active Error Flag.

There are three Elementary Tests to perform, corrupting the first, the third, and the sixth bit of the Active Error

Flag.

8.6.1.2 Test Case organisation

Test Case organisation see table 117.

Table 117 - 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 causes the IUT to generate an Active Error Frame. The LT corrupts one of the above mentioned Dominant bits of the Error Flag.
Verification	The TEC is increased by eight on the corrupted bit.

8.6.2 TEC increment on Bit Error during Overload Flag

8.6.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter increases its TEC by 8 when detecting a Bit Error during the transmission of an Overload Flag.

There are three Elementary Tests to perform, corrupting the first, the fourth, and the sixth bit of the Overload Flag.

8.6.2.2 Test Case organisation

Test Case organisation see table 118.

Table 118 - 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 causes the IUT to generate an Overload Frame. The LT corrupts one of the above mentioned Dominant bits of the Overload Flag.
Verification	The TEC is increased by eight at the corrupted bit.

8.6.3 TEC increment when Active Error Flag is followed by Dominant bits

8.6.3.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter increases its TEC by 8 when detecting eight consecutive Dominant bits following the transmission of its Active Error Flag and after each sequence of additional eight consecutive Dominant bits.

There is one Elementary Test to perform.

8.6.3.2 Test Case organisation

Test Case organisation see table 119.

Table 119 - 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 causes the IUT to generate an Active Error Frame. After the Error Flag sent by the IUT, the LT sends a sequence of 16 Dominant bits.
Verification	The TEC is increased by eight on each eighth Dominant bit after the Error Flag.

8.6.4 TEC increment when Passive Error Flag is followed by Dominant bits

8.6.4.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter increases its TEC by 8 when detecting eight consecutive Dominant bits following the transmission of its Passive Error Flag and after each sequence of additional eight consecutive Dominant bits.

There is one Elementary Test to perform.

8.6.4.2 Test Case organisation

Test Case organisation see table 120.

Table 120 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. Then the LT causes the IUT to generate a Passive Error Frame. After the Error Flag sent by the IUT, the LT sends a sequence of 16 Dominant bits.
Verification	The TEC is increased by eight on each eighth Dominant bit after the Error Flag.

8.6.5 TEC increment when Overload Flag is followed by Dominant bits

8.6.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter increases its TEC by 8 when detecting eight consecutive Dominant bits following the transmission of its Overload Flag and after each sequence of additional eight consecutive Dominant bits.

There is one Elementary Test to perform.

8.6.5.2 Test Case organisation

Test Case organisation see table 121.

Table 121 - 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 causes the IUT to generate an Overload Frame. After the Overload Flag sent by the IUT, the LT sends a sequence of 23 Dominant bits.
Verification	The TEC is increased by eight on each eighth Dominant bit after the Error Flag.

8.6.6 TEC increment on Bit Error in data frame

8.6.6.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter increases its TEC by 8 when detecting a Bit Error in a data frame on one of the following fields it transmits:

- SOF, Arbitration, Control, Data and CRC Fields.

In the Arbitration field, only bit error on Dominant bits shall be considered.

There are five Elementary Tests to perform.

8.6.6.2 Test Case organisation

Test Case organisation see table 122.

Table 122 - 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 corrupts a bit according to 8.6.6.1.
Verification	The TEC is increased by eight at the Bit Error detection.

8.6.7 TEC increment on Form Error in a frame

8.6.7.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter increases its TEC by 8 when detecting a Form Error in a frame on one of the following fields it transmits:

- CRC Delimiter, Acknowledge Delimiter, EOF (first bit, fourth bit and last bit).

There are five Elementary Tests to perform.

8.6.7.2 Test Case organisation

Test Case organisation see table 123.

Table 123 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 corrupts a bit according to 8.6.7.1.
Verification	The TEC is increased by eight at the Form Error detection.

8.6.8 TEC increment on Acknowledgement Error

8.6.8.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an Active State IUT acting as a transmitter increases its TEC by 8 when detecting an Acknowledgement Error in a frame.

There is one Elementary Test to perform.

8.6.8.2 Test Case organisation

Test Case organisation see table 124.

Table 124 - 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 does not acknowledge the frame.
Verification	The TEC is increased by eight at the Acknowledgement Error detection.

8.6.9 TEC increment on Form Error in Error Delimiter

8.6.9.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter increases its TEC by 8 when detecting a Form Error on a bit of the Error Delimiter it is transmitting.

There are three Elementary Tests to perform, corrupting the first, the fourth, and the sixth bit of the Error Delimiter.

8.6.9.2 Test Case organisation

Test Case organisation see table 125.

Table 125 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 causes the IUT to generate an Error Frame. The LT corrupts the Error Delimiter according to 8.6.9.1.
Verification	The TEC is increased by eight at the corrupted bit.

8.6.10 TEC increment on Form Error in Overload Delimiter

8.6.10.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter increases its TEC by 8 when detecting a Form Error during the transmission of an Overload Delimiter.

There are two Elementary Tests to perform, corrupting the second, and the seventh bit of the Overload Delimiter.

8.6.10.2 Test Case organisation

Test Case organisation see table 126.

Table 126 - 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 causes the IUT to generate an Overload Frame. The LT corrupts one of the above mentioned Recessive bits of the Overload Delimiter.
Verification	The TEC is increased by eight at the corrupted bit.

8.6.11 TEC decrement on successful frame transmission for TEC < 128

8.6.11.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an Active State IUT decreases its TEC by 1, when transmitting a valid frame.

There is one Elementary Test to perform.

8.6.11.2 Test Case organisation

Test Case organisation see table 127.

Table 127 - Test Case organisation

State	Description
Set-up	The LT forces the IUT to increase its TEC
Test	The LT causes the IUT to transmit a frame. The LT acknowledges this frame.
Verification	The TEC is decreased by 1 after the frame is completed.

8.6.12 TEC decrement on successful frame transmission for TEC > 127

8.6.12.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that a Passive State IUT decreases its TEC by 1 when transmitting a valid data frame.

There is one Elementary Test to perform.

8.6.12.2 Test Case organisation

Test Case organisation see table 128.

Table 128 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. The LT acknowledges this frame.
Verification	The TEC is decreased by 1 after the frame is completed.

8.6.13 TEC non-increment on 13 bit long Overload Flag

8.6.13.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter does not change the value of its TEC when receiving a 13 bit long Overload Flag.

There is one Elementary Test to perform.

8.6.13.2 Test Case organisation

Test Case organisation see table 129.

Table 129 - Test Case organisation

State	Description
Set-up	The LT forces the IUT to increase its TEC.
Test	The LT causes the IUT to transmit a frame. After the last bit of the EOF, the LT sends a sequence of 14 Dominant bits.
Verification	The TEC must equal the Set-up value decreased by 1.

8.6.14 TEC non-increment on 13 bit long Error Flag

8.6.14.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter does not change the value of its TEC when monitoring an Error Flag with 13 bit length.

There is one Elementary Test to perform.

8.6.14.2 Test Case organisation

Test Case organisation see table 130.

Table 130 - 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 and causes the IUT to send an Active Error Flag. After the last bit of the Error Flag, the LT sends a sequence of 7 Dominant bits.
Verification	The TEC value must be 8.

8.6.15 TEC non-increment on Form Error at last bit of Overload Delimiter

8.6.15.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter does not change the value of its TEC when detecting a Form Error on the last bit of the Overload Delimiter it is transmitting.

There is one Elementary Test to perform.

8.6.15.2 Test Case organisation

Test Case organisation see table 131.

Table 131 - Test Case organisation

State	Description
Set-up	The LT forces the IUT to increase its TEC.
Test	The LT causes the IUT to transmit a frame. Then the LT causes the IUT to generate an Overload Frame. At the last bit of the Overload Delimiter, the LT sends one Dominant bit.
Verification	The TEC must equal the Set-up value.

8.6.16 TEC non-increment on Form Error at last bit of Error Delimiter

8.6.16.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter does not change the value of its TEC when detecting a Form Error on the last bit of the Error Delimiter it is transmitting.

There is one Elementary Test to perform.

8.6.16.2 Test Case organisation

Test Case organisation see table 132.

Table 132 - Test Case organisation

State	Description
Set-up	The LT forces the IUT to increase its TEC.
Test	The LT causes the IUT to transmit a frame. Then the LT causes the IUT to generate an Error Frame. At the last bit of the Error Delimiter, the LT sends one Dominant bit.
Verification	The TEC must equal the Set-up value increased by 8.

8.6.17 TEC non-increment on Acknowledgement Error in Passive State

8.6.17.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that a Passive State IUT acting as a transmitter does not increase its TEC, when detecting an Acknowledgement Error followed by a Passive Error Flag.

There is one Elementary Test to perform.

8.6.17.2 Test Case organisation

Test Case organisation see table 133.

Table 133 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. The LT does not acknowledge this frame. After the Acknowledgement Error the LT sends a Passive Error Frame.
Verification	The TEC must equal the Set-up value.

8.6.18 TEC increment on Acknowledgement Error in Passive State

8.6.18.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that a Passive State IUT acting as a transmitter increases its TEC, when detecting an Acknowledgement Error followed by at least one Dominant bit during the Passive error Flag.

There is one Elementary Test to perform.

8.6.18.2 Test Case organisation

Test Case organisation see table 134.

Table 134 - Test Case organisation

State	Description
Set-up	The IUT is set to the TEC Passive State.
Test	The LT causes the IUT to transmit a frame. The LT does not acknowledge this frame. After the Acknowledgement Error the LT sends a Dominant bit at the sixth bit position of the Passive Error Flag.
Verification	The TEC must equal the Set-up value increased by 8.

8.6.19 TEC non-increment on Stuff Error during arbitration

8.6.19.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

This test verifies that an IUT acting as a transmitter does not increase its TEC when detecting a Stuff Error during arbitration when monitoring a Dominant bit.

There is one Elementary Test to perform.

8.6.19.2 Test Case organisation

Test Case organisation see table 135.

Table 135 - 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 and forces a Recessive Stuff Bit to a Dominant State.
Verification	The TEC must equal the Set-up value.

8.7 Bit timing class

8.7.1 Sample Point Test

8.7.1.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify the sample point of an IUT acting as a transmitter.

There is one Elementary Test to perform.

8.7.1.2 Test Case organisation

Test Case organisation see table 136.

Table 136 - 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. The LT shortens a Dominant bit preceded by a Recessive bit by an amount of (Phase_Seg2 - TSYS) and later shortens another Dominant bit preceded by a Recessive bit by an amount of (Phase_Seg2 + 1TQ + 1TSYS).
Verification	The IUT must generate an Error Frame on the bit position following the second shortened bit.

8.7.2 Hard synchronisation on SOF reception before sample point

8.7.2.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that the IUT, with a pending transmission, makes a hard synchronisation when detecting a Dominant bit before the sample point of the third bit of the Intermission Field.

There is one Elementary Test to perform.

8.7.2.2 Test Case organisation

Test Case organisation see table 137.

Table 137 - 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. While the IUT's transmission is pending the LT generates a Dominant bit starting (1 TQ + 1 TSYS) before the sample point of the third bit of the Intermission Field.
Verification	The IUT must start transmitting the first bit of the identifier one bit time after the recessive-to-Dominant edge of the SOF. The first edge inside the Arbitration Field has to be transmitted an integer number of bit time after the SOF edge.

8.7.3 Hard Synchronisation on SOF Reception after sample point

8.7.3.1 Purpose and limits of this Test Case

CAN_VERSION $\in$ {A, B, BP}

The purpose of this test is to verify that the IUT, with a pending transmission, makes a hard synchronisation when detecting a Dominant bit after the sample point of the third bit of the Intermission Field.

There is one Elementary Test to perform.

8.7.3.2 Test Case organisation

Test Case organisation see table 138.

Table 138 - 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. While the IUT's transmission is pending the LT generates a Dominant bit starting IPT after the sample point of the third bit of the Intermission Field.
Verification	The IUT must start transmitting its SOF at the next TQ following the dominant-to-Recessive edge.

8.7.4 Synchronisation when e < 0 and |e| ≤ SJW

8.7.4.1 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 transmitter, detecting a negative phase error (e) on a recessive-to-Dominant edge with $|e| \leq SJW$.

The value tested for e are in Time Quantum with $|e| \in$ [1, min (SJW, (Phase_Seg2-IPT))].

There are at least (SJW, (Phase_Seg2-IPT)) Elementary Tests to perform.

8.7.4.2 Test Case organisation

Test Case organisation see table 139.

Table 139 - 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. The LT shortens a Recessive bit preceding a Dominant bit by an amount of |e| inside the Arbitration Field.
Verification	The next edge sent by the IUT occurs an integer number of bit times after the edge applied by the LT.

8.7.5 Synchronisation for e < 0 and |e| > SJW

8.7.5.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify the behaviour of an IUT acting as a transmitter detecting a negative phase error (e) on a Recessive to Dominant bit with |e|>SJW.

The value tested for e are in Time Quantum |e| ∈ [(SJW + 1), (Phase_Seg2-IPT))].

There are ((Phase_Seg2-IPT) - SJW) Elementary Tests to perform.

8.7.5.2 Test Case organisation

Test Case organisation see table 140.

Table 140 - 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. The LT shortens a Recessive bit preceding a Dominant bit by an amount of |e| inside the Arbitration Field.
Verification	The next edge sent by the IUT occurs an integer number of bit times plus an amount (|e| - SJW) after the edge applied by the LT.

8.7.6 Glitch filtering test on negative phase error

8.7.6.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that there is only one synchronisation within one bit time if there are two Dominant to Recessive edges between two sample points where the first edge comes before the Synchronisation segment.

The test also verifies that an IUT is able to synchronise on a minimum duration pulse obeying to the synchronisation rules.

There is one Elementary Test to perform.

8.7.6.2 Test Case organisation

Test Case organisation see table 141.

Table 141 - 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. A Recessive bit which is followed by a Dominant bit is shortened by one Time Quantum. After one Time Quantum of Dominant value the LT forces one Time Quantum to Recessive value.
Verification	The IUT must send a dominant-to-Recessive edge an integer number of bit time after the first Recessive to Dominant edge applied by the LT.

8.7.7 Non-synchronisation on Dominant bit transmission

8.7.7.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an IUT transmitting a Dominant bit does not perform any re-synchronisation as a result of a recessive-to-Dominant edge with a positive phase error.

There is one Elementary Test to perform.

8.7.7.2 Test Case organisation

Test Case organisation see table 142.

Table 142 - 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. While the IUT is transmitting the frame, the LT delays each Recessive to Dominant edge by 2 time quanta.
Verification	The IUT must continue transmitting frame without any re-synchronisation.

8.7.8 Synchronisation before Information Processing Time

8.7.8.1 Purpose and limits of this Test Case

CAN_VERSION ∈ {A, B, BP}

The purpose of this test is to verify that an IUT transmitting will synchronise correctly in case of a re-synchronisation as a result of a Recessive-to-Dominant edge that occurs immediately after the Sample point.

There is one Elementary Test to perform.

8.7.8.2 Test Case organisation

Test Case organisation see table 142.

Table 143 - 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 that contains a sequence of 4 alternating recessive and dominant bits. While the IUT is transmitting the frame, the LT shortens the first Recessive bit of the alternating sequence by an amount of Phase_Seg2 and sends a Dominant bit.
Verification	The next edge from Recessive to Dominant sent by the IUT must occur two CAN bit times + (Phase_Seg2 – SJW) after the edge applied by the LT and the IUT must continue transmitting the frame.

9 Bi-directional frame type

9.1 Valid frame format class

There is defined no Test Case.

9.2 Error detection class

There is defined no Test Case.

9.3 Error Frame management class

There is defined no Test Case.

9.4 Overload Frame management class

There is defined no Test Case.

9.5 Passive error state class

There is defined no Test Case.

9.6 Error counter management class

9.6.1 REC unaffected when increasing TEC

CAN_VERSION ∊ {A, B, BP}

This test verifies that increasing REC and TEC are independent operations.

There is one Elementary Test to do.

9.6.1.1 Test Case Organisation

Test Case organisation see table 143.

Table 144 - 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 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.
Verification	Each increment of the TEC must be responded by an Active Error Flag. The IUT responds to the Stuff Error with a Passive Error Flag.

9.6.2 TEC unaffected when increasing REC

CAN_VERSION ∊ {A, B, BP}

This test verifies that increasing REC and TEC are independent operations.

There is one Elementary Test to do.

9.6.2.1 Test Case Organisation

Test Case organisation see table 144.

Table 145 - 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 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.
Verification	Each increment of the REC must be responded by an Active Error Flag. The IUT responds to the corrupted bit with a Passive Error Flag.

9.7 Bit timing class

There is defined no Test Case.

EXPLANATORY REPORT **RAPPORT EXPLICATIF**	**ISO/DIS** 16845
will supersede: remplacera:	
ISO/TC 22 /SC 3	Secretariat DIN/FAKRA

This form should be sent to the ISO Central Secretariat, together with the English and French versions of the committee draft, by the secretariat of the technical committee or subcommittee concerned (see 2.5.9 of part 1 of the ISO/IEC Directives)

Ce formulaire doit être envoyé au Secrétariat central de l'ISO en même temps que les versions anglaise et française du projet de comité, par le secrétariat du comité technique ou du sous-comité concerné (voir 2.5.9 de la partie 1 des Directives ISO/CEI)

The accompanying document is submitted for circulation to member body vote as a DIS, following consensus of the P-*members of the committee obtained*

Le document ci-joint est soumis, pour diffusion comme DIS, au vote comité membre, suite au consensus des membres (P) *du comité obtenu*

on / le 19

□ at the meeting of / à la réunion du TC /SC : see / voir resolution / résolution No. / n° in / dans le document

☒ by postal ballot initiated on / par un vote par correspondance démarré le 1999-01-07

P-members in favour: Membres (P) approuvant le projet:	China, Czech Republic, France, Germany, Italy, Japan, Korea (Republic), The Netherlands, Poland, Romania, Russian Federation, Spain, Sweden, UK
P-members voting against: Membres (P) désapprouvant:	-
P-members abstaining: Membres (P) s'abstenant:	Hungary, Ireland, Switzerland
P-members who did not vote: Membres (P) n'ayant pas voté:	

Remarks/Remarques

I hereby confirm that this draft meets the requirements of part 3 of the ISO/IEC Directives
Je confirme que ce projet satisfait aux prescriptions de la partie 3 des Directives ISO/CEI

Date
2000-05-30

Name and signature of the secretary
Nom et signature du secrétaire
Eisenacher