BS ISO 17458-3:2013

BSI Standards Publication

Road vehicles — FlexRay communications system

Part 3: Data link layer conformance test specification

bsi.

...making excellence a habit.™

Document Sharing Hub

National foreword

This British Standard is the UK implementation of ISO 17458-3:2013.

The UK participation in its preparation was entrusted to Technical Committee AUE/16, Electrical and electronic equipment.

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

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

© The British Standards Institution 2013. Published by BSI Standards Limited 2013

ISBN 978 0 580 76714 2

ICS 43.040.15

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

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 January 2013.

Amendments issued since publication

Date	Text affected

Documents
Document Sharing Hub

Full document details: BS ISO 17458-3:2013

INTERNATIONAL STANDARD

ISO 17458-3

First edition
2013-02-01

Road vehicles— FlexRay communications system —

Part 3: Data link layer conformance test specification

Véhicules routiers — Système de communications FlexRay —

Partie 3: Spécification d'essai de conformité de la couche de liaison de données

Reference number
ISO 17458-3:2013(E)

Documents
Document Sharing Hub
© ISO 2013

Full document details: BS ISO 17458-3:2013

COPYRIGHT PROTECTED DOCUMENT

© ISO 2013

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub

Full document details: BS ISO 17458-3:2013

Contents

Page

Foreword ... iv

Introduction ... v

1 Scope ... 1

2 Normative references ... 1

3 Terms, definitions, symbols and abbreviated terms ... 1
3.1 Terms and definitions ... 1
3.2 Symbols ... 1
3.3 Abbreviated terms ... 2
3.4 Functions ... 4

4 Conventions ... 4

5 Document overview ... 5

6 General ... 6
6.1 Test architecture ... 6
6.2 Test implementation ... 7
6.3 Internal RxDelay ... 9
6.4 Analog delays ... 9
6.5 Accepted deviations ... 10
6.6 Testability requirements ... 11
6.7 Test execution ... 14

7 Conformance test cases ... 16
7.1 General statements and test case structure ... 16
7.2 Receive data ... 23
7.3 CHI ... 125
7.4 Clock synchronisation ... 440
7.5 Wakeup ... 530
7.6 Startup ... 568
7.7 Miscellaneous ... 672
7.8 Optional TT-E feature ... 820
7.9 Preambles ... 846

8 Configuration ... 854
8.1 Basic configuration ... 854
8.2 Standard modifications ... 859

9 Static test cases ... 861
9.1 Electrical interface ... 861
9.2 Protocol parameter ... 861

Annex A (normative) **Technical data for the electrical interface of a FlexRay Communication Controller V3.0** ... 862

Annex B (normative) **Technical data for the protocol parameter of a FlexRay Communication Controller V3.0** ... 864

Bibliography ... 865

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Foreword

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

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. 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.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 17458-3 was prepared by Technical Committee ISO/TC 22, *Road vehicles*, Subcommittee SC 3, *Electrical and electronic equipment*.

ISO 17458 consists of the following parts, under the general title *Road vehicles — FlexRay communications system*:

— *Part 1: General information and use case definition*

— *Part 2: Data link layer specification*

— *Part 3: Data link layer conformance test specification*

— *Part 4: Electrical physical layer specification*

— *Part 5: Electrical physical layer conformance test specification*

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Introduction

The FlexRay communications system is an automotive focused high speed network and was developed with several main objectives which were defined beyond the capabilities of established standardized bus systems like CAN and some other proprietary bus systems. Some of the basic characteristics of the FlexRay protocol are synchronous and asynchronous frame transfer, guaranteed frame latency and jitter during synchronous transfer, prioritization of frames during asynchronous transfer, single or multi-master clock synchronization, time synchronization across multiple networks, error detection and signalling, and scalable fault tolerance.

The FlexRay communications system is defined for advanced automotive control applications. It serves as a communication infrastructure for future generation high-speed control applications in vehicles by providing:

— A message exchange service that provides deterministic cycle based message transport;

— Synchronization service that provides a common time base to all nodes;

— Start-up service that provides an autonomous start-up procedure;

— Error management service that provides error handling and error signalling;

— Wakeup service that addresses the power management needs.

Since start of development the automotive industry world wide supported the specification development. The FlexRay communications system has been successfully implemented in production vehicles today.

The ISO 17458 series specifies the use cases, the communication protocol and physical layer requirements of an in-vehicle communication network called "FlexRay communications system".

This part of ISO 17458 has been established in order to define the protocol conformance test case requirements.

To achieve this, it is based on the Open Systems Interconnection (OSI) Basic Reference Model specified in ISO/IEC 7498-1 and ISO/IEC 10731, which structures communication systems into seven layers. When mapped on this model, the protocol and physical layer requirements specified by ISO 17458 are broken into:

— Diagnostic services (layer 7), specified in ISO 14229-1 [14], ISO 14229-4 [16];

— Presentation layer (layer 6), vehicle manufacturer specific;

— Session layer services (layer 5), specified in ISO°14229-2 [15];

— Transport layer services (layer 4), specified in ISO 10681-2 [5];

— Network layer services (layer 3), specified in ISO 10681-2 [5];

— Data link layer (layer 2), specified in ISO 17458-2, ISO 17458-3;

— Physical layer (layer 1), specified in ISO 17458-4, ISO 17458-5;

in accordance with Table 1.

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Table 1 — FlexRay communications system specifications applicable to the OSI layers

Applicability	**OSI 7 layers**	ISO 17458 **FlexRay communications system**	**Vehicle manufacturer enhanced diagnostics**
Seven layer according to ISO 7498-1 and ISO/IEC 10731	Application (layer 7)	vehicle manufacturer specific	ISO 14229-1, ISO 14229-4
	Presentation (layer 6)	vehicle manufacturer specific	vehicle manufacturer specific
	Session (layer 5)	vehicle manufacturer specific	ISO 14229-2
	Transport (layer 4)	vehicle manufacturer specific	ISO 10681-2
	Network (layer 3)	vehicle manufacturer specific	
	Data link (layer 2)	ISO 17458-2, ISO 17458-3	
	Physical (layer 1)	ISO 17458-4, ISO 17458-5	

Table 1 shows ISO 17458 Parts 2 – 5 being the common standards for the OSI layers 1 and 2 for the FlexRay communications system and the vehicle manufacturer enhanced diagnostics.

The FlexRay communications system column shows vehicle manufacturer specific definitions for OSI layers 3 – 7.

The vehicle manufacturer enhanced diagnostics column shows application layer services covered by ISO 14229-4 which have been defined in compliance with diagnostic services established in ISO 14229-1, but are not limited to use only with them. ISO 14229-4 is also compatible with most diagnostic services defined in national standards or vehicle manufacturer's specifications. The presentation layer is defined vehicle manufacturer specific. The session layer services are covered by ISO 14229-2. The transport protocol and network layer services are specified in ISO 10681.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

INTERNATIONAL STANDARD ISO 17458-3:2013(E)

Road vehicles — FlexRay communications system — Part 3: Data link layer conformance test specification

1 Scope

This part of ISO 17458 specifies the FlexRay protocol conformance test. This test verifies the conformance of FlexRay communication controllers with respect to ISO 17458-2.

Some testability requirements are given in 6.2.2.3 and 6.6 and are applicable for FlexRay communication controllers to pass the conformance test

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 17458-2, *Road vehicles — FlexRay communications system — Part 2: Data link layer specification*

ISO 17458-4, *Road vehicles — FlexRay communications system — Part 4: Electrical physical layer specification*

3 Terms, definitions, symbols and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions defined in ISO 17458-2 and ISO 17458-4 apply.

3.2 Symbols

Δ	delta
$\in$	Element, lower-case epsilon
ξ	Xsi
µT	microtick
σT	*gdSampleClockPeriod* (= Sampletick)
t_{RC}	modification of cycle length due to calculated rate correction (equal to *zRateCorrection*, used in figures of "Clock synchronisation")
t_{OC}	modification of cycle length due to calculated offset correction (equal to *zOffsetCorrection*, used in figures of "Clock synchronisation")

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

φ_{Rx}, φ_{Tx} analogue delays (see 6.4)

ξ, ξ_{IUT} allowed deviation from theoretical results due to LT-IUT jitter (see 6.5)

3.3 Abbreviated terms

BC basic configuration

BD bus driver

BSS byte start sequence

CAS collision avoidance symbol

CC communication controller

CE communication element

CHI controller host interface

CHIRP channel idle recognition point

CRC cyclic redundancy code

DC dual channel

DTS dynamic trailing sequence

FES frame end sequence

FIFO first in first out

FPGA field programmable gate array

FSS frame start sequence

ID identifier

IP intellectual property

IUT implementation under test

LT lower tester

MT macrotick

MTS media access test symbol

NIT network idle time

NM network management

PE protocol engine

POC protocol operation control

RTL register transfer level

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

RxD	receive data signal from bus driver
SC	single channel
TE	test execution
TSS	transmission start sequence
TT-D	time triggered distributed
TT-E	time triggered external
TxD	transmit data signal from CC
TxEN	transmit data enable not signal from CC
UT	upper tester
WUDOP	wakeup during operation pattern
WUP	wakeup pattern
WUS	wakeup symbol

POC states:

C	*POC:config*
CSCC	*POC:coldstart consistency check*
CSCR	*POC:coldstart collision resolution*
CSG	*POC:coldstart gap*
CSJ	*POC:coldstart join*
CSL	*POC:coldstart listen*
DC	*POC:default config*
H	*POC:halt*
ICC	*POC:integration consistency check*
ICSC	*POC:integration coldstart check*
IL	*POC:integration listen*
IS	*POC:initialize schedule*
NA	*POC:normal active*
NP	*POC:normal passive*
R	*POC:ready*

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

3.4 Functions

TruncateTowardsZero: function returns the integer part of a number without its fractional digits (= sign(x) * floor(|x|))

4 Conventions

ISO 17458 are based on the conventions specified in the OSI Service Conventions (ISO/IEC 10731) as they apply for physical and data link layer (protocol).

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

5 Document overview

Figure 1 depicts the FlexRay document reference according to OSI model.

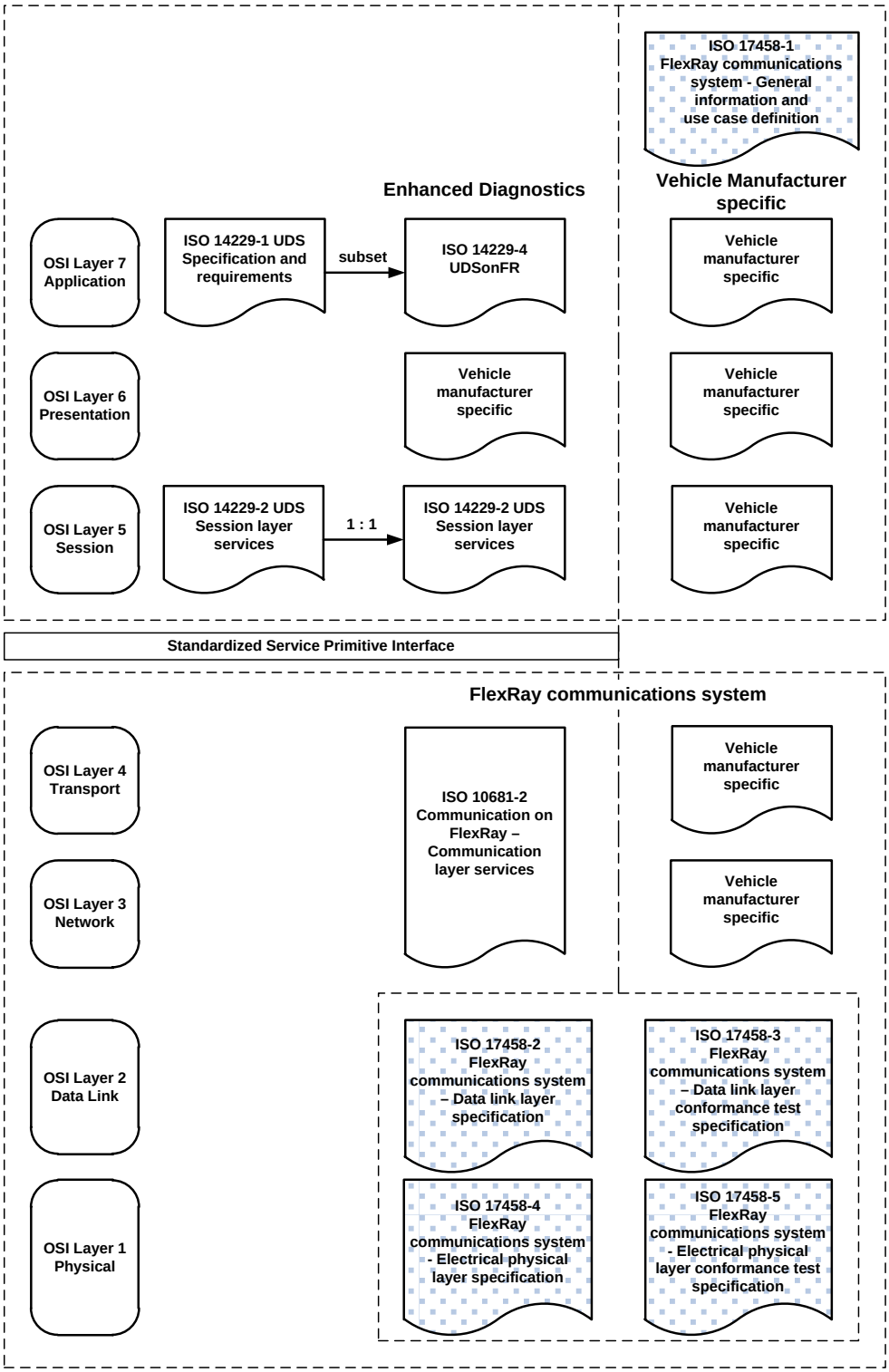

Figure 1 — FlexRay document reference according to OSI model

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

6 General

6.1 Test architecture

This part of ISO 17458 is based on a test architecture as shown in Figure 2, which follows the ISO 9646 standard. The implementation under test (IUT) is the FlexRay CC. The upper tester (UT) is connected to the FlexRay controller host interface (CHI) of the IUT and the CHI is device specific. The lower tester (LT) is connected to the FlexRay physical layer interface of the IUT and ISO 17458-4 describes this interface. The test coordination procedure controls the UT and the LT.

In a hardware-based test environment (see 6.2.2), the FlexRay CC can be an “embedded” FlexRay CC meaning that CC is embedded in a microcontroller. In this case, the CHI (i.e., the upper tester interface) is between the embedded FlexRay CC and the microcontroller and in order to get access to the CHI, the upper tester may be partly distributed to the microcontroller.

The test architecture shown in Figure 2 is suitable for testing one FlexRay CC.

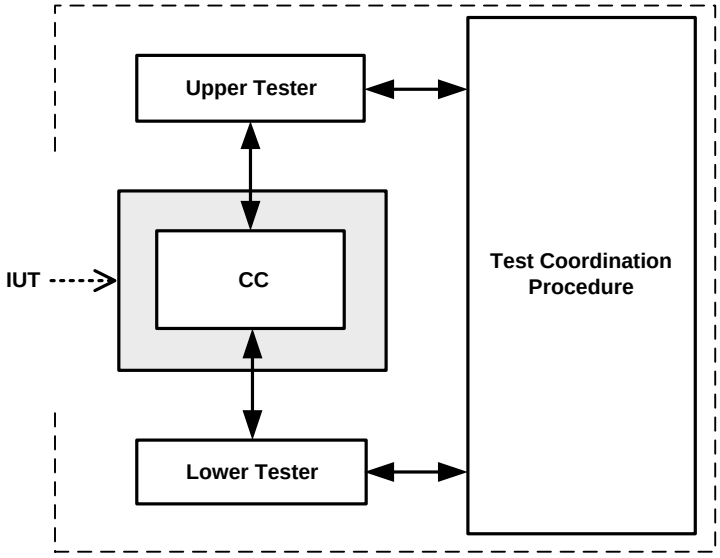

Figure 2 — Standard test architecture

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

There are some optional test cases in this part of ISO 17458 which tests the optional TT-E feature by using a pair of FlexRay CCs, namely a source CC and a sink CC, which are connected via the time gateway interface. Here, the IUT is the pair of connected FlexRay CCs. To test the TT-E feature, the test architecture as shown in Figure 3 is proposed. The upper tester and lower tester are connected to both FlexRay CCs and the test coordination procedure controls the upper and lower tester.

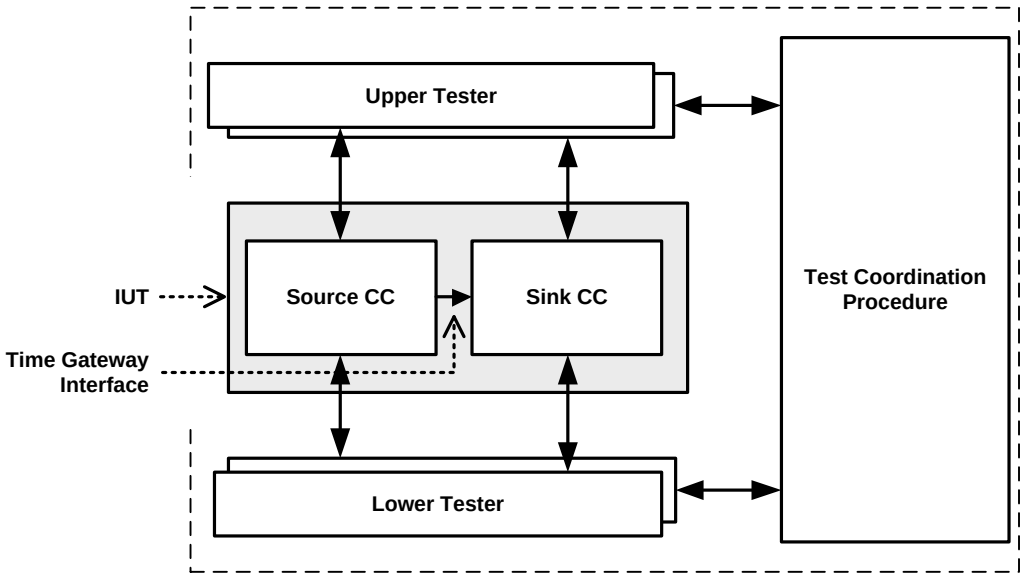

Figure 3 — Test architecture for the TT-E option

6.2 Test implementation

6.2.1 General

The test cases and the proposed test architecture of this specification can be either implemented in a hardware-based environment or in a simulation-based environment. In the following, both environments are described.

6.2.2 Hardware-based test implementation

6.2.2.1 IUT

In a hardware-based test implementation, the IUT is a physical device. The IUT can be either a standalone FlexRay CC, an embedded FlexRay CC, or a FlexRay CC programmed in an FPGA. For testing the optional TT-E feature, the IUT consists of a pair of connected FlexRay CCs.

6.2.2.2 Lower tester

The electrical characteristics of the lower tester shall follow the electrical characteristics of the interface between the FlexRay CC and the FlexRay Bus Driver on the Bus Driver side. This interface is described in ISO 17458-4. The requirements on the lower tester in a hardware-based test environment, including the electrical characteristics, are listed in Table 2. It is not advised to use any extra circuits (e.g., level shifters) between the IUT's FlexRay CC and the lower tester.

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Table 2 — Requirements on the lower tester in a hardware-based test environment

Description	Relevant signal	Parameter name used in ISO 17458-4	Min	Max	Unit
Input capacitance	TxD, TxEN	*C_BDTxD*	-	10	pF
Threshold for detecting logical high	TxD, TxEN	*uBDLogic_1*	-	60	%
Threshold for detecting logical low	TxD, TxEN	*uBDLogic_0*	40	-	%
Voltage reference for logical high and low	TxD, TxEN, RxD	*uVDIG*	same as IUT		V
Sample Rate	TxD, TxEN	N/A	160	-	MHz
Asymmetry	RxD	see ISO 17458-4: measured at 50 % *uVDIG* and 25 pF load	-	2	ns
Sum of rise and fall time @ 15 pF load	RxD	*dBDRxDR15* + *dBDRxDF15*	-	13	ns
Difference of rise and fall time @ 15 pF load	RxD	| *dBDRxDR15* – *dBDRxDF15* |	-	5	ns
Sum of rise and fall time @ 25 pF load	RxD	*dBDRxDR25* + *dBDRxDF25*	-	16,5	ns
Difference of rise and fall time @ 25 pF load	RxD	| *dBDRxDR25* – *dBDRxDF25* |	-	5	ns
Frequency of FlexRay clock, provided to IUT	clk	N/A	-	80	MHz
Precision of FlexRay clock, provided to IUT	clk	N/A	-	500	ppm

6.2.2.3 Clock synchronisation

Several test cases require the synchronisation between the IUT and the test environment such that random clock deviations can be excluded and the occurrence time of a sampletick can be determined within one sampletick accuracy.

Therefore it is required that the LT provides a clock signal (called “FlexRay clock”) to the IUT. The LT shall use the FlexRay clock as a basis for FlexRay bus stimuli of the RxD signal and for sampling the TxD and TxEN signals. The IUT shall also use the FlexRay clock for FlexRay transmission and reception. Existing PLLs in the IUT need to be bypassed or programmed not to multiply. If the IUT uses an own clock source or an active PLL, then there might be the risk of failing some test cases.

Some requirements on the FlexRay clock signal are listed in Table 2. In addition, those frequencies shall be supported, which can be derived from 80 MHz by integer division. The FlexRay clock shall run continuously in order to be able to test embedded FlexRay CCs, where the FlexRay clock signal is also used to for clocking the host microcontroller.

6.2.3 Simulation-based test implementation

In a simulation-based test implementation, the IUT is described in a hardware description language typically on register transfer level (RTL), e.g., in SystemVerilog code or VHDL code. The IUT is a FlexRay CC, including message buffers and FIFO. For testing the optional TT-E feature, the IUT consists of a pair of connected FlexRay CCs. The test environment (upper tester, lower tester, and test coordination procedure) and the test cases exist as software only. Executing a test case in a simulation-based test implementation means to load all necessary parts (IUT, test environment, test case) in an RTL simulator and then to run the simulation.

In the simulation-based test implementation, the clocks of the IUT and the LT have no deviation from the nominal frequency and also have no jitter. Therefore, no requirements on the clock synchronisation between the LT and the IUT (as listed in 6.2.2.3 for the hardware-based test implementation) are given for the

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

simulation-based test implementation. However, ξ_{IUT} and ξ (see 6.5) have to be considered in the simulation-based test implementation.

6.3 Internal RxDelay

The parameter *adInternalRxDelay* is implementation specific and has an allowed range between 1 and 4 sampleticks ISO 17458-4. All basic configurations (see clause 8) assume *adInternalRxDelay* to be 4 sampleticks. In order to compensate between the implementation specific value of *adInternalRxDelay* and the assumed value of 4 sampleticks, a delay compensation shall be integrated in the RxD signal between the LT and the IUT. The delay compensation shall have a delay of 4-dt [sampleticks], and dt is the IUT's actual value of *adInternalRxDelay*. The delay compensation shall be applied in the hardware-based test implementation and in the simulation-based test implementation. Figure 4 gives an example how to implement the delay compensation in a hardware-based test implementation.

Please note that in the test cases, the time interval between IUT's frame and LT's frame are measured at the test points marked in Figure 4. Therefore the delay compensation is not to be considered in test steps like *"It is verified (LT) that the interval between the IUT's frames in slot 1 and LT's frame slot 2 is x µT..."*.

Figure 4 depicts the proposal for compensation of *adInternalRxDelay* in a hardware-based test implementation.

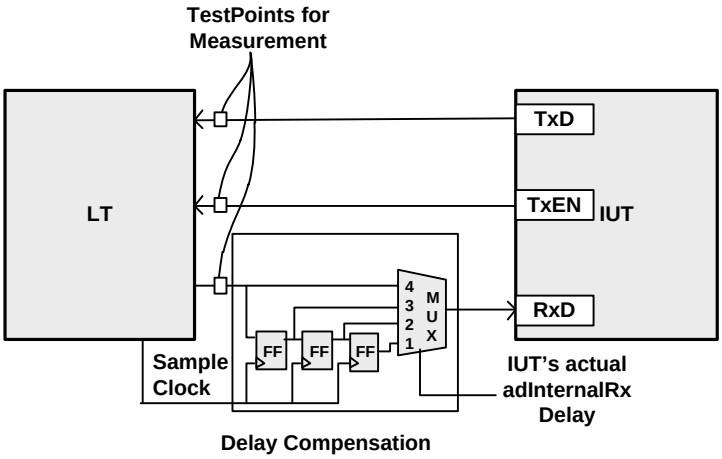

Figure 4 — Proposal for compensation of adInternalRxDelay in a hardware-based test implementation

6.4 Analog delays

Analog delays of the IUT appear in the signal paths between the physical pads of the device and the flip flops of the FlexRay CC. In ISO 17458-4, the analog delays are captured in the following parameters: *dCCRxD01*, *dCCRxD10*, *dCCTxD01*, *dCCTxD10*, *dCCTxEN10* or *dCCTxEN01*. In this conformance test specification, the following two analog delays are defined: φ_{Rx} is the analog delay on the reception path, and φ_{Tx} is the analog delay on the transmission path. In the hardware-based test implementation, the analog delays are determined as follows:

— $\varphi_{Rx} \in [\ 0;\ \max\{dCCRxD01, dCCRxD10\}\ ]$

— $\varphi_{Tx} \in [\ 0;\ \max\{dCCTxD01, dCCTxD10, dCCTxEN10, dCCTxEN01\}\ ]$

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Here, the parameters *dCCRxD01*, *dCCRxD10*, *dCCTxD01*, *dCCTxD10*, *dCCTxEN10* or *dCCTxEN01* have the maximal values as defined in ISO 17458-4.

In the simulation-based test environment, there are no physical pads but only the FlexRay CC itself simulated. Therefore, the analog delays can be set to 0 here:

— $\varphi_{Rx} = 0$

— $\varphi_{Tx} = 0$

6.5 Accepted deviations

Despite the synchronisation of the IUT and the test environment described in 6.5 there is a remaining deviation between the LT and the IUT. A sample point at the LT does not necessarily cause a sample point at the IUT at the same moment since the clock signal stemming from the test environment can be delayed within the IUT due to the clock tree. Due to this we have to assume that the IUT's sample point occurs anywhere within two sample points of the LT and could also have a small amount of jitter.

Figure 5 depicts the shift of IUT sampletick relative to LT's signal.

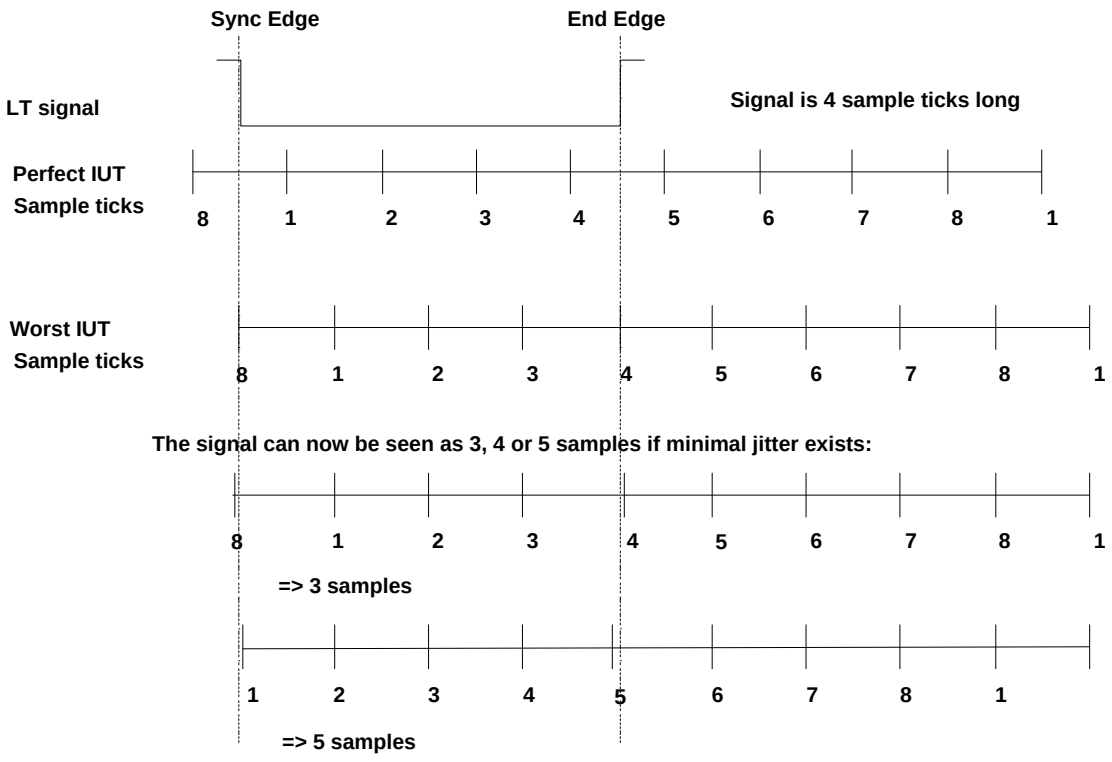

Figure 5 — Shift of IUT sampletick relative to LT's signal

Figure 5 shows that in the worst case this can lead to a situation where the IUT samples directly on the edges of the LT's signal. In that case even a small amount of jitter can cause a 4 sample signal of the LT to be seen as a 5 sample signal (line 3 of Figure 5) or a 3 sample signal (line 4 of Figure 5) by the IUT. This of course has consequences for the precision test cases can achieve and also on the results of the IUT's clock synchronisation algorithm.

The imprecision seen in Figure 5 prevents exact testing of the bit strobing point. Sending a 4 sample long pattern can, as has been shown, either be seen at the bit strobing point (sample point 5) or not. Similarly a 5

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

sample long pattern could not be seen at the bit strobing point. Thus 4 and 5 sample long LOW patterns can not be used to verify if bit strobing correctly happens at sample 5.

Similar the clock synchronisation mechanism of FlexRay can be disturbed by this sample point deviations and cause a slight de-synchronisation of IUT and test environment. Figure 5 shows that a sync edge which arrives at the same time as a sampletick can either be seen already by this sampletick (line 4) or only by the next one (line 3). If this happens to the FSS/BSS sync edge, the time reference point of the frame shifts (see ISO 17458-2 Figure "Time reference point definitions"). If due to jitter this time reference point shifts into different directions in consecutive cycles, the IUT will erroneously determine the need to correct its macrotick clock rate and phase (=offset). This means that all higher level events (e.g. slot starts, cycle starts, ...) that depend on macroticks can be slightly unaligned between IUT and test environment. This has to be taken into account in the test cases.

To address the deviations introduced by the unaligned FlexRay clock correction mechanism, ξ_{IUT} is introduced, with the allowed range of:

$$\xi_{IUT} \in [-6;5]\mu T$$

ξ_{IUT} expresses the range of deviation that can be introduced in signals coming from the IUT to the LT as well as the status variables related to clock correction sent from the IUT to the UT. The maximum tolerances for IUT signals and clock correction related variables as seen by the LT and UT shall be limited to. The test environment and setup shall minimize those tolerances as much as possible. Please note that ξ_{IUT} is only a valid deviation if the FlexRay clock correction is involved in the measurement in question. For example it is not to be applied to tests of the decoder or to tests within the WAKEUP period.

In addition, the LT introduces a deviation when measuring due to the granularity of measurements as well as some inherent inexactness. Thus every time measurement of the IUT by the LT is assumed to have a possible deviation of:

$$\xi \in [-1;1]\mu T$$

ξ is to be added as tolerance to each traffic measurement at the IUT's interface to the LT.

6.6 Testability requirements

6.6.1 General

Since ISO 17458-2 lacks some requirements which are necessary for defining precise test cases (e.g., CHI delays), this subclause specifies such testability requirements.

NOTE subclause 6.2.2.3 also gives a testability requirement in a hardware-based test implementation. A FlexRay CC, which does not meet the testability requirements, runs the risk of failing the test cases.

6.6.2 CHI delay constraints

ISO 17458-2 states for the controller host interface (CHI) that *"due to implementation constraints the CHI may add product specific delays for data or control signals exchanged between the host and the protocol engine"*. The maximum value of some CHI delays are to be known for testing a FlexRay CC. Therefore, Table 3 specifies the necessary maximum values and the maximum values are based on the time t_0 at which the event / the variable change happens in the protocol engine. It is assumed that if a host requests one of these values from the CHI anywhere between (t_0 + Max. delay) and the earliest visibility time of the next change of the value, the host will receive the correct value.

NOTE the delay values in Table 3 do not include the time required to transfer the data from the CHI to the host. They define the earliest point in time when a host might access the data and is guaranteed to receive the updated value in

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

response. It is expected that the CHI transfers the requested data immediately with no delay[1)] except that inherent in the CHI-host interface (e.g. due to round trip times, bandwidth restrictions, parallel ongoing transfers, ...). Should the underlying value changes during the transfer (e.g. during the transfer of payload a new frame arrives for the slot) it is the task of the CHI to ensure the atomicity of the transfer.

In order to be able to pass the test cases of this specification, the CHI delays of a FlexRay CC shall follow the maximum values and it is highly recommended that the delays are much below the maximum values.

Table 3 — Maximum delay allowed to occur in the CHI

Parameter as defined in ISO 17458-2	Earliest visible	Maximum delay in CHI	Remarks
CHI commands like RUN, ALL_SLOTS…	NA	2 000 µT after command submission	Thus for testing purposes it can not be assumed that the PE will react according to the new state until the delay has passed.
Protocol operation control status	earliest allowed state change time	500 µT after latest allowed state change time in PE	Several state changes are defined to happen sometime within the NIT. For these earliest and latest allowed state change time differ (typically to beginning and end of NIT).
Wakeup and startup status	at state change	500 µT after state change	---
vMacrotick	at macrotick	½ MT, minimum 20 µT, after macrotick	The repeated consecutive reads are not guaranteed, i.e. reading at macrotick x and then immediately again does not guarantee a result of x or x + 1.
vCycleCounter	at cycle start	500 µT after cycle start	---
vSlotCounter A/B	at slot start or segment start	50 µT after PE slot counter change	---
vInterimRateCorrection vInterimOffsetCorrection	at NIT start	500 µT after end of cycle	---
various error indicators	at event occurrence	sync frame overflow: 500 µT after slot or segment boundary; *pLatestTx*: 500 µT after dynamic segment end; transmission across boundary: 500 µT after slot or segment boundary	---
Synchronisation frame status	start of NIT	10 MT after the start of offset correction phase	for non TT-E coldstart nodes
Startup frame status	1 µT before cycle start	10 MT after start of cycle	for TT-E coldstart nodes only
Startup frame status	start of NIT	10 MT after the start of offset correction phase	---
Symbol window status vSS	at symbol window end	500 µT after symbol window end	---
NIT status vSS	at NIT end (= start of next cycle)	500 µT after start of cycle	---

1) However if there is a transfer delay greater than a given verification window, the access time shall be considered by the UT.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Parameter as defined in ISO 17458-2	Earliest visible	Maximum delay in CHI	Remarks
Aggregated channel status	at each slot and segment end	500 µT after each boundary	This could mean that it is not checkable for empty dynamic slots. But this is not necessary for the conformance test as that only checks this value during the NIT.
Dynamic segment status	at dynamic segment end	500 µT after dynamic segment end	---
Transmit buffer status	at slot end / segment end	2 * *gdStaticSlot* for static slots or 2 000 µT for dynamic slots, counting from the slot boundary at the end of the respective slot	Definition is to keep it in accordance with payload.
Slot status data	at slot end / segment end	2 * *gdStaticSlot* for static slots or 2 000 µT for dynamic slots, counting from the slot boundary at the end of the respective slot	---
Frame contents data	at complete (valid!) frame arrival	2 * *gdStaticSlot* for frames received in the static segment or 2 000 µT for frames received in the dynamic segment, counting from the slot boundary following the valid frame	2 000 µT is provided for the dynamic segment where slots can become as short as 2 MT, the duration of 2 static slots is used to scale the available time according to the payload length used. Counting from the end of the slot means, that e.g. the payload of a valid frame arriving in slot 1 could be accessed the earliest after the end of slot 3.
Queued receive buffers	at slot and segment end on complete arrival of a valid frame	2 000 µT after complete arrival of a valid frame	---
Interrupt service	at event occurrence	100 µT after event occurrence	---
Accrued network management vector	at each update (see ISO 17458-2)	500 µT after each update (see ISO 17458-2)	---

6.6.3 Consequences for interpretation of test steps

The upper limits of the CHI delays of Table 3 are used in the test cases explicitly to determine the earliest point in time a change of a CHI parameter can be checked / verified by the UT and the earliest point in time the LT can expect the IUT to show appropriate behaviour due to commands from the UT.

In several test cases, esp. in the preambles, the UT issues several CHI commands in short sequence, mostly to configure the IUT. In order to reduce the runtime of the test cases, the upper limits of the CHI delays are not explicitly stated here, but it is assumed that the UT takes care of the CHI delays and delays any further commands to the IUT until the first one was executed.

In the beginning of each test case the IUT is reset by the UT. In this situation it is the responsibility of the test environment to ensure that the IUT has completed its reset before further actions are performed.

6.6.4 Special handling of vMacrotick counter

The delay specification for *vMacrotick* given above results in a 20 µT window for reading the counter in case the macrotick length is configured as 40 µT. But additionally the IUT's macrotick can shift around the LT's macrotick by up to ξ_{IUT} µT with $\xi_{IUT} \in [-6;5]$ (for definition of ξ_{IUT} see 6.5). As such the validity window shrinks

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

by 11 µT, requiring the UT to hit a 9 µT window to read the *vMacrotick* counter correctly in all circumstances, a non-trivial task.

To ease this problem, additional information about the delay and the jitter for accessing the *vMacrotick* counter shall be provided. For example, the additional information could be “the *vMacrotick* counter is valid at the CHI from 5 µT after the IUT's macrotick to 2 µT after the next IUT's macrotick”. For a macrotick length of 40 µT this results in a validity window of 37 µT, which, after reduction for ξ_{IUT}, provides the UT with a 26 µT window.

6.7 Test execution

6.7.1 Single channel CC

For a single channel CC all test cases with “SC” applicability and all test cases “SC, DC” applicability shall be executed in two runs: one run with *pChannels* = A and one run with *pChannels* = B. Subclause 7.1 gives more details on the “applicability” label per test case.

Total number of test cases for a single channel CC: 391 (this is the sum of 5 "SC" test cases and 386 "SC, DC" test cases).

6.7.2 Dual channel CC

For a dual channel CC, all test cases with “DC” applicability and all test cases with “SC, DC” applicability shall be executed in one run with *pChannels* = A&B. In addition, all test cases with “SC” applicability and all test cases with “SC, DC” applicability shall be executed in two runs: one run with *pChannels* = A and one run with *pChannels* = B.

Total number of test cases for a dual channel CC: 420 (this is the sum of 5 "SC" test cases, 386 "SC, DC" test cases, and 29 "DC" test cases).

6.7.3 Optional TT-E feature

TT-E stands for time triggered external synchronisation method. To test the optional TT-E feature requires a pair of CCs, which are connected by the time gateway interface. One CC is the time gateway source, the other CC is the time gateway sink. See clause 6 and Figure 3. Each CC can be either a single channel or a dual channel implementation. If a pair of connected CCs claims to support the TT-E option, all test cases with “TT-E” applicability shall be executed.

Total number of test cases for the TT-E option: 12

All other test cases shall be executed for each of the two CCs according to 6.7.1 and 6.7.2.

Depending on whether the source and sink CC is a single channel or a dual channel CC, Table 4 gives the required runs of the TT-E test cases and the corresponding set “$pChannels_{Source,Sink}$“of configurations of the parameter *pChannels* for both CCs. For example, only row 1 in Table 4 has to be considered if both CCs are single channel CCs, only row 4 has to be considered of both CCs are dual channel CCs. Notation used for the set $pChannels_{Source,Sink}$:*“(pChannels for Source:CC, pChannels for Sink:CC)”*. For example *“(A,A&B)”* means that the source CC is set to single channel mode with *pChannels*=A and the sink CC is set to dual channel mode with *pChannels*=A&B.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Table 4 — Configurations of pChannels for both CCs, depending on whether each CC (Source and Sink) is a single channel or a dual channel one

Row	Source CC	Sink CC	Number of runs of the "TT-E" tests and the set of configurations for pChannels
1	single channel	single channel	Four runs with $pChannels_{Source,Sink}$ = {(A,A),(A,B),(B,A),(B,B)}
2	single channel	dual channel	Six runs with $pChannels_{Source,Sink}$ = {(A,A),(A,B),(B,A),(B,B),(A,A&B),(B,A&B)}
3	dual channel	single channel	Six runs with $pChannels_{Source,Sink}$ = {(A,A),(A,B),(B,A),(B,B),(A&B,A),(A&B,B)}
4	dual channel	dual channel	Nine runs with $pChannels_{Source,Sink}$ = {(A,A),(A,B),(B,A),(B,B),(A,A&B),(B,A&B),(A&B,A),(A&B,B),(A&B,A&B)}

NOTE The following statement is valid for both CCs (source and sink CC): If the CC is a dual channel CC and it is set to single channel mode (*pChannels*=A or *pChannels*=B), the not configured channel shall be tested for inactivity. If *pChannels*=A, channel B is "not configured"; if *pChannels*=B, channel A is "not configured."

6.7.4 Requirements on clock synchronisation and wakeup

The following shall be considered in dual channel test execution (i.e., if *pChannels* = A&B):

— All test cases specified in 7.4 shall be executed in three instances: in instance 1 the LT simulates its frames on channel A, in instance 2 the LT simulates its frames on channel B, and in instance 3 the LT simulates its frames on both channels.

— All test cases specified in 7.5 shall be executed twice, first with *pWakeupChannel* = A and second with *pWakeupChannel* = B.

Exceptions to this and additional instructions in dual channel test executions are specified per test case and at the beginning of 7.4 and 7.5.

6.7.5 Basic configurations

All test cases, which are to be executed according to 6.7.1 through 6.7.3, shall be executed with all five basic configurations 1a, 1b, 2a, 2b, and 3. The five basic configurations represent the combination of mandatory bit rates and microtick lengths.

Table 5 holds the simplified five basic configurations and Clause 8 lists the basic configurations in detail with all relevant parameters.

Table 5 — Simplified basic configurations: bit rates and microtick lengths

Basic Configuration	1a	1b	2a	2b	3
Bit Rate [Mbit/s]	10	10	5	5	2,5
pdMicrotick [µs]	0,025	0,0125	0,025	0,05	0,05

6.7.6 Modifications, variants, instances

A test case can have modifications of the basic configurations, variants, and instances (see 7.1 for more details). For each test case, all defined modifications, variants, and instances shall be executed.

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

7 Conformance test cases

7.1 General statements and test case structure

7.1.1 General statements

All following general statements are valid for all test cases, unless otherwise specified in the individual test case.

— **Single channel test execution:**
A test case is in “single channel test execution” if it is executed with *pChannels*=A or *pChannels*=B. In single channel test execution, the stimulus and all channel dependent verifications shall be applied only for the channel configured by *pChannels*. If the IUT is a dual channel CC, the “not configured” channel shall be tested for inactivity. If *pChannels*=A, channel B is “not configured”; if *pChannels*=B, channel A is “not configured”.

— **Dual channel test execution:**
A test case is in “dual channel test execution” if it is executed with *pChannels*=A&B. In dual channel test execution, the stimulus shall be applied to both channels symmetrically and all channel dependent verifications shall be applied for both channels.
The phrase “For dual channel test execution <instructions>” means that the <instructions> shall be considered if the test case is executed with *pChannels*=A&B. Examples: “For dual channel test execution it is verified (LT) that the non-wakeup channel stays idle throughout the entire test.”, “For dual channel test execution the test has to be performed in three instances.”

— **Respective channel(s)” and “available channel(s):**
The phrases “respective channel(s)” and “available channel(s)” reflect the channel(s) on which the LT applies its stimuli.

— **Reset:**
A statement similar to “The UT resets the IUT” means that the UT resets the IUT and waits at least for the implementation specific delay to allow the IUT finishing the reset process. After reset, the IUT is in POC:default config.

— **Multiple communication elements in a slot:**
If the LT simulates multiple communication elements, e.g. frames and symbols, within a single slot, those communication elements have to be separated by at least 11 * *gdBit* in order to enable the IUT to differentiate them.

— **Default frame content:**
The default frame contents for frames simulated by the LT or transmitted by the IUT are defined as follows unless otherwise specified in the test case or preamble. After the execution of the preamble, the LT continues with the simulation of valid frames in the used slots of the preamble, unless otherwise specified.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Table 6 defines the default frame contents.

Table 6 — Default frame contents

Default frame contents		
Segment	static	dynamic
Reserved bit	0	0
Payload preamble indicator	0	0
Null frame indicator	1	1
Sync frame indicator	0	0
Startup frame indicator	0	0
Payload length	*gPayloadLengthStatic*	1
Payload data on channel A	all bytes set to 0x00	byte 0: 0xCC and byte 1: 0x33
Payload data on channel B	all bytes set to 0x00	byte 0: 0xCC and byte 1: 0x33
Cycle Count	current cycle count	

— **Check / verify:**
Each test case consists of a preamble, a test execution, and a postamble (see 7.2). The phrase “It is checked that ...” is a “check statement” and it is used in the preamble and postamble. The phrase “It is verified that ...” is a “verify statement” and it is used in test execution. If a “check statement” fails, the setup state for the testing part cannot be reached as required. Therefore the test cannot be performed as specified and the test case is not passed. If a “verify statement” fails the test case is failed.

— **Pass criteria:**
The intention of the section “Pass criteria” in the test cases is to summarize the “verify statements” of the test execution. The section does not imply additional “check statements”or “verify statements”.

— **Implicit checks**:
Implicit checks can be optionally performed in all test cases during preamble, test execution and postamble, even if these checks are not specified in the test case. If an implicit check fails, the test case is not prevented from being passed, but it can be treated as a hint that there might be an unintended behaviour within the IUT.
Although the implicit checks are not specified in very detail within this conformance test specification, they need to be implemented conform to the data link layer specification.
Implicit checks include (there can be more implicit checks):

— It is continuously checked (LT) that the IUT transmits nothing (i.e., TxD=high and TxEN=high on any channels supported by the IUT) between reset (as defined in ISO 17458-2) and the first time the UT issues the CHI_RUN command or the CHI_WAKEUP command.

— It is checked (LT) that the IUT transmits no communication elements in not assigned slots.

— If interrupt requests are globally enabled and interrupt requests for (one ore more) specific interrupt source(s) are enabled, then it is checked that whenever the interrupt requests are generated, the interrupt status indication flags for the enabled source(s) are set. This implicit check shall be performed according to ISO 17458-2 subclause "Interrupt service".

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

— **Not passed or failed test cases:**
Once a test case is not passed or has failed during conformance testing, the test result (failed or not passed) can only be modified after re-running the test case if:

  — the implementation of the test case has been modified so that the implementation follows the specification of the test case, or

  — the implementation of the test environment has been modified, or

  — the IUT has been modified, or

  — the specification of the test case has been modified and this modification has been accepted. Afterwards, the implementation of the test case shall be modified according to the modification of the test case specification.

— **Channel status:**
All channel specific status variables (e.g. flags, indicators, counters etc.) have to be verified on a per-channel basis for the active channel(s).

— **Aggregated channel status:**
Unless otherwise specified, in test cases which use the aggregated channel status for verification purposes, the aggregated channel status has to be cleared either in the NIT of every cycle (respectively after its verification in the NIT) except, for test cases in 7.2.2 test purpose at the end of the last static slot of every cycle.

— **Slot Status:**
To verify the slot status for an specific slot on a channel a receive buffer shall be configured by the UT in the IUT, if a buffer configuration is not explicitly stated in the test case.

— **Message buffers for transmission of frames:**
If a frame shall be transmitted in a specific slot, then a message buffer shall be configured for this slot even if the message buffer is not explicitly configured in the test case.

— **Message buffers for reception of frames:**
If a frame shall be received in a specific slot, then a message buffer shall be configured for this slot even if the message buffer is not explicitly configured in the test case.

— **Tolerance window:**
A statement similar to “The LT starts to simulate startup frames *gdCycle* ± 0,25 * *gdCycle* after the CHI command RUN” defines a tolerance window of ±0,25 * *gdCycle* during which the LT has to start its simulation. The exact point in time at which the LT begins its simulation is is not relevant for the correct test execution and therefor left free for the test implementer.

— **ξ and ξ_{IUT}:**
In a statement like “It is verified (LT) that the interval between the IUT's frames in slot 1 / cycle 12 and slot 1 / cycle 13 is *pMicroPerCycle* + ξ + ξ_{IUT} µT.“, the term “ξ + ξ_{IUT}“ defines a tolerance of [-7; 6] µT. I.e. the time interval between the two frames sent by the IUT, which is measured by the LT is allowed to deviate by [-7; 6] µT.
In contrast, if ξ_{IUT} is used in combination with a verification window, like “[0; 100] µT + ξ_{IUT} µT“, this means that the verification window from the point of view of the test environment has to be reduced accordingly, because of the allowed deviation of IUTs clock and test environments clock. This leads for the given example “[0; 100] µT + ξ_{IUT} µT” to an allowed window of [5; 94] µT in test environments view.

— **Verification window:**
A statement similar to “It is verified (UT) that the IUT is in the *POC:initialize schedule* state *gdCycle* ± 0,1 * *gdCycle* after the CHI command RUN.” defines a verification window during which the UT shall do the specified verification once (here: verification that IUT is in the *POC:initialize schedule*). The exact point in time within the verification window at which the UT does the verification is irrelevant for the correct test execution and therefor is left free to the test implementation.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

A verification window is also be defined by statements like “In cycle 8, within interval tAW after t0 it shall be verified ...” with “$t_{AW} \in$ [0, 500] µT”. In this case, the verification window starts at t0 and ends at t0+500 µT.

— The clock deviation (ξ_{IUT}) has to be considered in all verification windows (for LT and UT) by reducing the windows accordingly.

— **Detecting low simultaneously:**
A phrase like "It is verified (LT) that the TxEN output and TxD output become low simultaneously" allows a time tolerance of 1 σT for detecting low at both outputs due to allowed fall times at the outputs and logic thresholds of the LT. Let t0_TxEN be the time when low of the TxEN output is detected by the LT and let t0_TxD be the time when low of the TxD output is detected by the LT. Then the tolerance of 1σT means that the absolute difference between t0_TxEN and t0_TxD shall be at most 1σT, i.e., |t0_TxEN - t0_TxD| <= 1 σT.

— **Clock of the LT:**
Unless otherwise specified, the LT does not perform a clock correction but statically applies its communication elements as specified in the test cases.
Depending on the startup scenario of the test case either LT or IUT starts the schedule. If LT initiates the communication the IUT aligns with the given schedule. However, if IUT starts the communication the LT is aware of the expected IUT's start of schedule.
In some test cases LT's time is shifted by starting the next cycle earlier or later. In these test cases the order of test steps is important for the referenced time: test steps before the instruction to shift the schedule refer to the LT time before the shift, test steps after this instruction refer to the LT time after the shift. Statements to shift the simulated frames, however, do not influence LT's clock.

— **Protocol related points in time:**
Although the LT does not perform clock correction the clock correction algorithm in the IUT makes sure that the clocks of the LT and the IUT are synchronized. Therefore, for the verification timings relative to the cycle start, to the NIT and to similar protocol-related points in time LT's schedule shall be taken as the references point. The allowed clock deviation between IUT and LT, ξ_{IUT}, has to be considered for the verification timing by shortening the verification windows accordingly.
However, in some testcases LT's schedule is shifted in order to test the FlexRay clock correction algorithm. In these test cases IUT's schedule and LT's schedule differ for a number of cycles. To fit the requested verification timings in these cycles, valid verification windows are specified in both IUT's time and in LT's time. Due to simplification of the test description these verification windows may differ in start point or in size, but are valid by all means.

— **Execution time of test step:**
Each single test step shall be executed at the given time or within the given time window. If now timing is specified for a test step, the test step shall be executed immediately after completion of the previous test step.

— **LT transmission timing**:
Frames and symbols transmitted by the LT shall start at the according action point of the given slot or segment, unless otherwise specified. I.e. a statement like “... the LT simulates a frame in slot 1 ...” means that the LT starts the transmission at the action point of communication slot 1. The same principle applies to symbols transmitted in the symbol window.

— **Parameter value vs. preamble and test case:**
If the value of a parameter is defined in the preamble (Preamble I, II, III, IV) and in the test case itself (e.g., in the Configuration section of the test case), then the value as defined in the test case itself always supersedes the value as defined in the preamble.

— **Sampletick / Microtick / Macrotick:**
The abbreviation “σT” stands for sampletick and “σT” is equivalent to the parameter *gdSampleClockPeriod*. The abbreviation “MT” stands for macrotick and “MT” is equivalent to the parameter *gdMacrotick*. The abbreviation “µT” stands for microtick and “µT” is equivalent to the parameter *pdMicrotick*.

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

7.1.2 Test case structure

Each test case is structured by eight titles.

In the following, details and examples for each of these titles (ordered list) are given.

— **Test name**

Descriptive name of the test.

— **Test purpose**

Short description of which requirement of the FlexRay Protocol Specification is tested.

— **Applicability**

The following three applicability classes are available:

— SC: Single channel test case, i.e. applicable if pChannels = A or pChannels = B

— DC: Dual channel test case, i.e., applicable if pChannels = A&B

— TT-E: Time triggered external test case., i.e., applicable if the pair of CCs (source and sink CCs) offers the TT-E option

NOTE A test case might be applicable to more than one class, e.g., an applicability of “SC,DC” means that the test case is applicable if *pChannels* = A, *pChannels* = B, or *pChannels* = A&B.

— **Configuration**

The configuration section addresses the configuration of the parameters. If parameters need to be modified for the test execution, the modified parameter values are explicitly stated in the test case within a modification table. If modifications are used in a test case, the test case has to be executed for all modifications.

Different modification tables to basic configurations:

Table 7 defines one modification, valid for all basic configurations.

Table 7 — One modification, valid for all basic configurations

Parameter	Modification
pKeySlotID	2

Table 8 defines the set of modifications (numbered with Roman letters I, II, III, …), valid for all basic configurations.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Table 8 — Set of modifications (numbered with Roman letters I, II, III, …), valid for all basic configurations

Parameter	Modification		
	I	II	III
pAllowPassiveToActive	1	16	31
pAllowHaltDueToClock	false		

Table 9 defines the set of modifications, depending on the basic configurations.

Table 9 — Set of modifications, depending on the basic configurations

Parameter	Modification to Basic Configurations		
	1a & 1b	2a & 2b	3
gdStaticSlot [MT]	42	40	68
gNumberOfStaticSlots	70	37	21
gPayloadLengthStatic [two-byte word]	1		
gdNIT [MT]	49	17	33

— **Preamble (setup state)**

In this subclause the necessary actions are defined to drive the IUT from the stable state (i.e. reset state) to the initial state for test execution. Four preambles (I, II, III, and IV) are defined in 7.8.4.4 Test purpose with IUT as following coldstart node, as leading coldstart node, as integrating node and as time gateway source / sink node. In some test cases, the preambles are modified slightly.

— **Test execution**

Test execution is organized in steps of taking actions by UT or LT and the verification of correct IUT's reaction and IUT's behaviour .

Dependency on modifications:

If parameter modifications are used and the execution of a test step depends on a certain modification, this dependency is marked by roman letters in brackets, i.e., “(I)”, (II)” or “(I,II)”.

EXAMPLE

Line 1: 1. In cycle 9, the LT simulates its frame in slot 1 with a payload length in the frame header equal to

Line 2: (I) *gPayloadLengthStatic* + 1

Line 3: (II, III) *gPayloadLengthStatic*

In this example, the test step 1 (lines 1-3) depends on the parameter modifications, i.e., line 2 is to be executed for modification I, line 3 is to be executed for modifications II and III.

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Variants:

If the execution of a test step depends on a certain variant, this dependency is marked by lower Latin letters in brackets, i.e., “(a)”, “(b)”, “(c)”, etc. If variants are used in a test case, the test case has to be executed for all variants.

Line 1: 1. In cycle 16, 500 µT after cycle start and before NIT, it is verified (UT) that the counter *vAllowPassiveToActive* = 0 (*pAllowPassiveToActive* – 1) and that

Line 2: (a, b) the IUT stays in the *POC:normal passive* state (*vPOC!State* = NORMAL_PASSIVE) and

Line 3: (c) the IUT has returned to the *POC:normal active* state (*vPOC!State* = NORMAL_ACTIVE) and has set the *vPOC!ErrorMode* to ACTIVE.

In this example, the test step 1 (lines 1-3) depends on the variants, i.e., line 2 is to be executed for variants (a) and (b), line 3 for variant (c).

Instances:

In some test case, instances are used. In the following, two examples for instances are given.

EXAMPLE 1:

For dual channel test execution the test has to be performed in three instances. The LT generates its stimuli for instance 1 on channel A, for instance 2 on channel B and for instance 3 on both channels.

EXAMPLE 2:

Test instance 1:

— In cycles 7 to 9, the LT simulates its startup frame in slot 2.

— In cycle 7, it is verified (UT) that *vPOC!CHIHaltRequest* = false.

— The UT clears (resets) the interrupt status indication flags of the IUT for all interrupt sources.

— …

Test instance 2:

— In cycles 7 to 9, the LT simulates its startup frame in slot 2.

— In cycle 7, it is verified (UT) that *vPOC!CHIHaltRequest* = false.

— At the beginning of cycle 8, the UT issues the CHI command DEFERRED_HALT.

— etc.

A test case shall be executed for all given instances.

Notes on modifications, variants, and instances:

In some test cases, modifications, variants, and instances are used in combinations in a nested way, e.g., two modifications (I and II) and three variants (a), (b), and (c) which gives in total 2 * 3 = 6 test executions for this example.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

— **Postamble**

Description of the test steps defining how to drive the IUT into a final definite state.

— **Pass criteria**

Short description of the test pass criteria.

7.2 Receive data

7.2.1 Static segment

7.2.1.1 TSS

— **Test purpose**

Verify recognition of the transmission start sequence TSS in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations using the modifications in Table 10, Table 11 and Table 12.

Table 10 — Modification to basic configuration 1a and 1b for static segment – receive data

Parameter	Modification to basic configurations					
	1a			1b		
	I	II	III	I	II	III
gdCASRxLowMax [gdBit]	77	97	105	77	97	105
gdTSSTransmitter [gdBit]	1[a]	11	15	1[a]	11	15
pDecodingCorrection [µT]	12	52	68	24	104	136
pMacroInitialOffset[A,B] [MT]	5	6		5	6	
pMicroInitialOffset[A,B] [µT]	23		7	46		14

[a] *gdTSSTransmitter* = 1 [*gdBit*] which is an intentional protocol constraints violation

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Table 11 — Modification to basic configuration 2a and 2b for static segment – receive data

Parameter	Modification to basic configurations					
	2a			2b		
	I	II	III	I	II	III
gdCASRxLowMax [gdBit]	70	80	84	70	80	84
gdTSSTransmitter [gdBit]	1[a]	6	8	1[a]	6	8
pDecodingCorrection [µT]	24	64	80	12	32	40
pMacroInitialOffset[A,B] [MT]	4		5	4		5
pMicroInitialOffset[A,B] [µT]	46	6	70	23	3	35
[a] *gdTSSTransmitter* = 1 [*gdBit*] which is an intentional protocol constraints violation						

Table 12 — Modification to basic configuration 3 for static segment – receive data

Parameter	Modification to basic configuration 3		
	I	II	III
gdCASRxLowMax [gdBit]	66	72	74
gdTSSTransmitter [gdBit]	1[a]	4	5
pDecodingCorrection [µT]	24	48	56
pMacroInitialOffset[A,B] [MT]	4	5	
pMicroInitialOffset[A,B] [µT]	6	22	14
[a] *gdTSSTransmitter* = 1 [*gdBit*] which is an intentional protocol constraints violation			

— **Preamble (setup state)**

Preamble I.

— **Test execution**

1) The LT simulates its frame in slot 1 with a TSS length of
3 / 8 [*gdBit*] in cycle 8,
1 [*gdBit*] in cycle 9,
gdTSSTransmitter [*gdBit*] in cycle 10,
gdTSSTransmitter + 2 + 3 / 8 [*gdBit*] in cycle 11, and
gdTSSTransmitter + 3 [*gdBit*] in cycle 12.

2) In the NIT of cycle 8, it is verified (UT) that the syntax error flag / indicator is true and that the valid frame flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status.

3) In the NIT of cycles 9, 10 and 11, it is verified (UT) that the syntax error flag / indicator is false and that the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status.

4) In the NIT of cycle 12, it is verified (UT) that the syntax error flag / indicator is true and that the valid frame flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The syntax error flag / indicator and the valid frame flag / indicator are set accordingly in the slot status data and in the aggregated channel status.

7.2.1.2 Reserved bit

— **Test purpose**

Verify reception of the reserved bit in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations.

— **Preamble (setup state)**

Preamble I.

— **Test execution**

1) In cycle 8, the LT simulates its frame in slot 1 with the reserved bit set to '0' and it is verified (UT) that the reserved bit *vRF!Header!Reserved* is '0' in the frame contents data in the NIT of this cycle.

2) In cycle 9, the LT simulates its frame in slot 1 with the reserved bit set to '1' and it is verified (UT) that the reserved bit *vRF!Header!Reserved* is '1' in the frame contents data in the NIT of this cycle.

3) In cycle 10, the LT simulates its frame in slot 1 with the reserved bit set to '0' and it is verified (UT) that the reserved bit *vRF!Header!Reserved* is '0' in the frame contents data in the NIT of this cycle.

Figure 6 depicts the reserved bit.

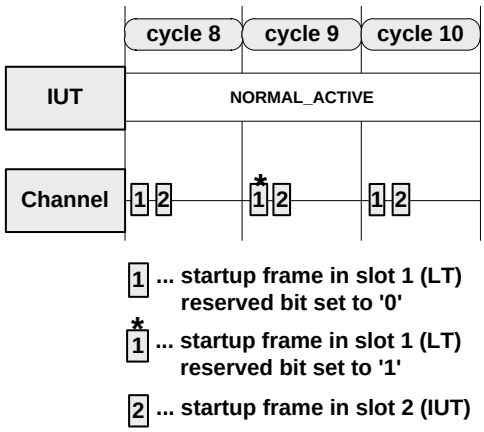

Figure 6 — Reserved bit

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The reserved bit *vRF!Header!Reserved* in the frame contents data is set accordingly.

7.2.1.3 Payload preamble indicator

— **Test purpose**

Verify reception of the payload preamble indicator in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations.

— **Preamble (setup state)**

Preamble I.

— **Test execution**

1) In cycle 8, the LT simulates its frame in slot 1 with the payload preamble indicator set to '0' and the null frame indicator set to '1'. It is verified (UT) that the payload preamble indicator *vRF!Header!PPIndicator* is '0' in the frame contents data in the NIT of this cycle. It is also verified (UT) that the valid frame flag / indicator is true and the content error flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of the cycle.

2) In cycle 9, the LT simulates its frame in slot 1 with the payload preamble indicator set to '1' and the null frame indicator set to '1'. It is verified (UT) that the payload preamble indicator *vRF!Header!PPIndicator* is '1' in the frame contents data in the NIT of this cycle. It is also verified (UT) that the valid frame flag / indicator is true and the content error flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of the cycle.

3) In cycle 10, the LT simulates its frame in slot 1 with the payload preamble indicator set to '0' and the null frame indicator set to '1'. It is verified (UT) that the payload preamble indicator *vRF!Header!PPIndicator* is '0' in the frame contents data in the NIT of this cycle. It is also verified (UT) that the valid frame flag / indicator is true and the content error flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of the cycle. It is also verified (UT) that the frame contents data corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 10.

4) In cycle 11, the LT simulates its frame in slot 1 with the payload preamble indicator set to '1', the null frame indicator set to '0' and all payload bytes set to 0x00. It is verified (UT) that the valid frame flag / indicator is false and the content error flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of the cycle. It is also verified (UT) that the frame contents data corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 10.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

Figure 7 depicts the payload preamble indicator.

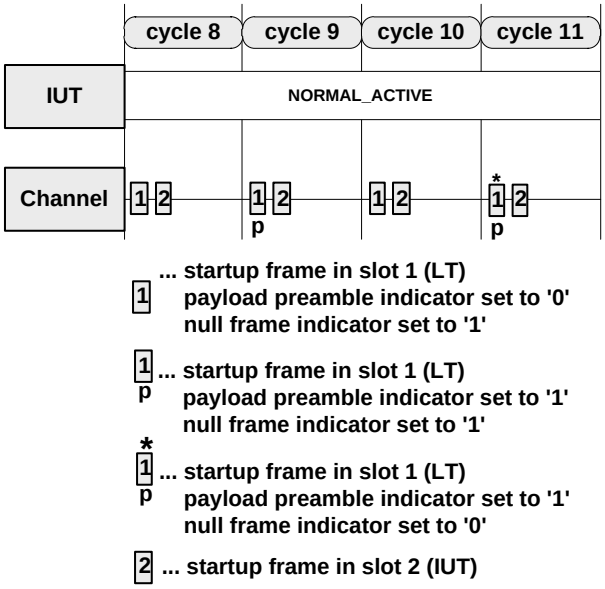

Figure 7 — Payload preamble indicator

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The payload preamble indicator *vRF!Header!PPIndicator* in the frame contents data is set accordingly and additionally the content error flag / indicator and the valid frame flag / indicator are set accordingly in the slot status data and in the aggregated channel status. The IUT does not update the frame contents data upon reception of an invalid frame.

7.2.1.4 Null frame indicator

— **Test purpose**

Verify reception of the null frame indicator in the static segment. Verify that the IUT does not update frame contents data upon reception of a null frame.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations.

— **Preamble (setup state)**

Preamble I.

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

— **Test execution**

1) In cycle 8, the LT simulates its frame in slot 1 with the null frame indicator set to '1'. It is verified (UT) that the flag *vSS!NFIndicator* is '1' in the slot status data of slot 1 in the NIT of this cycle. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

2) In cycle 9, the LT simulates its frame in slot 1 with the null frame indicator set to '0' and all payload bytes set to 0x00. It is verified (UT) that the flag *vSS!NFIndicator* is '0' in the slot status data of slot 1 in the NIT of this cycle. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

3) In cycle 10, the LT simulates its frame in slot 1 with the null frame indicator set to '1'. It is verified (UT) that the flag *vSS!NFIndicator* is '1' in the slot status data of slot 1 in the NIT of this cycle. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 10.

Figure 8 depicts the null frame indicator.

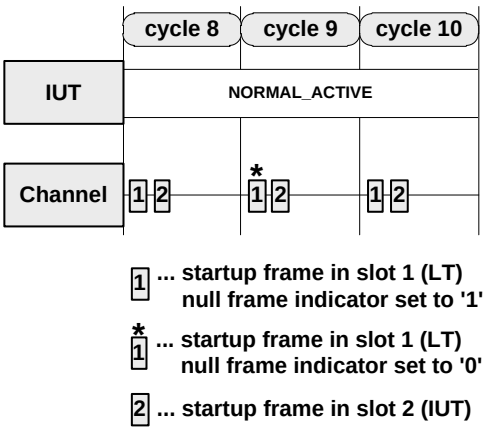

Figure 8 — Null frame indicator

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The flag *vSS!NFIndicator* in the slot status data is set accordingly. The IUT does not update frame contents data upon reception of a null frame.

7.2.1.5 Sync frame indicator

— **Test purpose**

Verify reception of the sync frame indicator in the static segment and the correct error indication in case of the reception of a startup frame having its sync frame indicator set to '0'.

— **Applicability**

SC, DC.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

— **Configuration**

All basic configurations using the modifications in Table 13.

Table 13 — Modification to basic configuration for static segment – sync frame indicator

Parameter	Modification
gMaxWithoutClockCorrectionFatal	15
gMaxWithoutClockCorrectionPassive	15

— **Preamble (setup state)**

Preamble I.

— **Test execution**

$t_{AW} \in [0, 500]$ µT

Depending on the considered channel(s) the respective received or transmitted sync frames lists shall be verified in the test execution.

1) In cycle 8, the LT simulates its startup frame with the sync frame indicator set to '1' and matching header CRC.

2) In the symbol window of cycle 8, it is verified (UT) that the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 2 (the IUT's startup frame ID) and the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 1 (the LT's startup frame ID).

3) In the NIT of cycle 8, it is verified (UT) that the sync frame indicator *vRF!Header!SyFIndicator* in the frame contents data is '1' and the content error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status.

4) The UT clears the aggregated channel status after its verification.

5) In cycle 8, within the interval t_{AW} after the offset correction start +10 * *gdMacrotick / pdMicrotick* µT, it is verified (UT) that the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 2 the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 1 and the number of valid sync frames received and transmitted in the even and in the odd communication cycles is 2.

6) In cycle 9, the LT simulates its startup frame with the sync frame indicator set to '0' and matching header CRC.

7) In the symbol window of cycle 9, it is verified (UT) that the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 2 and the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 1.

8) In the NIT of cycle 9, it is verified (UT) that the sync frame indicator *vRF!Header!SyFIndicator* in the frame contents data is '1' (invalid frame received and therefore not updated) and the content error flag / indicator is true and the valid frame flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status. The UT clears the aggregated channel status after its verification.

9) In cycle 9, within the interval t_{AW} after the offset correction start +10 * *gdMacrotick / pdMicrotick* µT, it is verified (UT) that the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 2, the received or transmitted sync frames list for even cycle contains the frame ID 1, the

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

received or transmitted sync frames list for odd cycle does not contain the frame ID 1, the number of valid sync frames received and transmitted in the even communication cycle is 2 and the number of valid sync frames received and transmitted in the odd communication cycle is 1.

10) In cycle 10, the LT simulates a static frame instead of a startup frame in slot 1.

11) In the symbol window of cycle 10, it is verified (UT) that the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 2, the received or transmitted sync frames list for even cycle contains the frame ID 1 and the received or transmitted sync frames list for odd cycle does not contain the frame ID 1.

12) In the NIT of cycle 10, it is verified (UT) that the sync frame indicator *vRF!Header!SyFIndicator* in the frame contents data is '0' and the content error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status. The UT clears the aggregated channel status after its verification.

13) In cycle 10, within the interval t_{AW} after the offset correction start +10 * *gdMacrotick / pdMicrotick* µT, it is verified (UT) that the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 2, the received or transmitted sync frames lists for even and for odd cycles do not contain the frame ID 1 and the number of valid sync frames received and transmitted in the even and in the odd communication cycles is 1.

14) In cycle 11, the LT simulates its startup frame with the sync frame indicator set to '1' and matching header CRC.

15) In the symbol window of cycle 10, it is verified (UT) that the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 2 and the received or transmitted sync frames lists for even and for odd cycles do not contain the frame ID 1.

16) In the NIT of cycle 11, it is verified (UT) that the sync frame indicator *vRF!Header!SyFIndicator* in the frame contents data is '1' and the content error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status. The UT clears the aggregated channel status after its verification.

17) In cycle 11, within the interval t_{AW} after the offset correction start +10 * *gdMacrotick / pdMicrotick* µT, it is verified (UT) that the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 2, the received or transmitted sync frames list for odd cycle contains the frame ID 1, the received or transmitted sync frames list for even cycle does not contain the frame ID 1, the number of valid sync frames received and transmitted in the even communication cycle is 1 and the number of valid sync frames received and transmitted in the odd communication cycle is 2.

18) In cycle 12, the LT simulates its startup frame with the sync frame indicator set to '1' and matching header CRC.

19) In the symbol window of cycle 12, it is verified (UT) that the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 2, the received or transmitted sync frames list for odd cycle contains the frame ID 1, the received or transmitted sync frames list for even cycle does not contain the frame ID 1, the number of valid sync frames received and transmitted in the even communication cycle is 1 and the number of valid sync frames received and transmitted in the odd communication cycle is 2.

20) In the NIT of cycle 12, it is verified (UT) that the sync frame indicator *vRF!Header!SyFIndicator* in the frame contents data is '1' and the content error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status.The UT clears the aggregated channel status after its verification.

21) In cycle 12, within the interval t_{AW} after the offset correction start +10 * *gdMacrotick / pdMicrotick* µT, it is verified (UT) that the received or transmitted sync frames lists for even and for odd cycles contain the frame ID 2, the received or transmitted sync frames lists for even and for odd cycles contain the

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

frame ID 1 and the number of valid sync frames received and transmitted in the even and in the odd communication cycles is 2.

Figure 9 depicts the sync frame indicator.

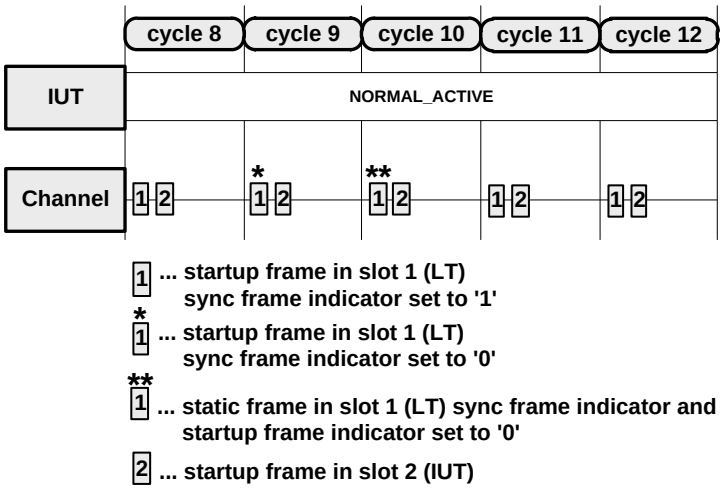

Figure 9 — Sync frame indicator

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The content error flag / indicator and the valid frame flag / indicator in the slot status data and in the aggregated channel status, the number of sync frames, the received or transmitted sync frames lists and the sync frame indicator *vRF!Header!SyFIndicator* in the frame contents data are set accordingly.

7.2.1.6 Startup frame indicator

— **Test purpose**

Verify reception of the startup frame indicator in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations.

— **Preamble (setup state)**

Preamble I.

— **Test execution**

1) In cycle 8, the LT simulates its startup frame with the startup frame indicator set to '1', the sync frame indicator set to '1' and matching header CRC. It is verified (UT) that the startup frame indicator *vRF!Header!SuFIndicator* in the frame contents data is '1', the valid frame flag / indicator is true and

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

the content error flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle.

2) In cycle 9, the LT simulates its startup frame with the startup frame indicator set to '0', the sync frame indicator set to '1' and matching header CRC. It is verified (UT) that the startup frame indicator *vRF!Header!SuFIndicator* in the frame contents data is '0', the valid frame flag / indicator is true and the content error flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle.

3) In cycle 10, the LT simulates its startup frame with the startup frame indicator set to '1', the sync frame indicator set to '1' and matching header CRC. It is verified (UT) that the startup frame indicator *vRF!Header!SuFIndicator* in the frame contents data is '1', the valid frame flag / indicator is true and the content error flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle.

Figure 10 depicts the startup frame indicator.

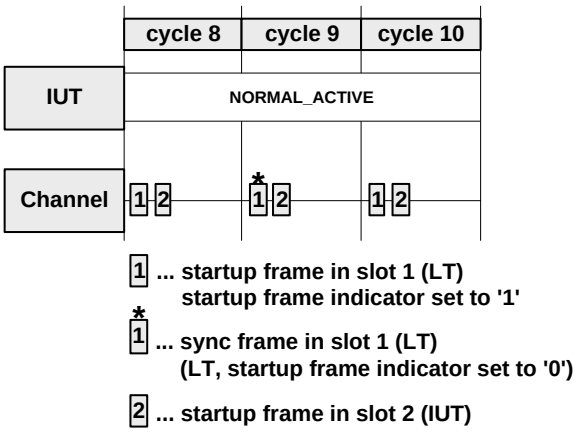

Figure 10 — Startup frame indicator

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The content error flag / indicator in the slot status data and in the aggregated channel status and the startup frame indicator *vRF!Header!SuFIndicator* in the frame contents data are set accordingly.

7.2.1.7 Frame ID

— **Test purpose**

Verify reception of the frame ID in the static segment. Verify that the IUT does not update frame contents data upon reception of invalid frames.

— **Applicability**

SC, DC.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

— **Configuration**

All basic configurations using the modifications as listed in Table 14.

Table 14 — Modification to basic configuration for static segment – frame ID

Parameter	Modification
gMaxWithoutClockCorrectionFatal	3

— **Preamble (setup state)**

Preamble I.

— **Test execution**

1) In cycle 8, the LT simulates its frame with the frame ID set to 1 and matching header CRC. It is verified (UT) that the frame ID *vRF!Header!FrameID* is 1 in the frame contents data, the content error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle. It is also verified (UT) that the remainder of the frame contents data of slot 1 in the IUT corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

2) In cycle 9, the LT simulates its frame with the frame ID set to 0, matching header CRC and with the payload 0x55 in byte 0 and 0xAA in byte 1. It is verified (UT) that the content error flag / indicator is true and the valid frame flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle. It is also verified (UT) that the frame contents data of slot 1 in the IUT is not updated and corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

3) In cycle 10, the LT simulates its frame with the frame ID set to 2, matching header CRC and with the payload 0x55 in byte 0 and 0xAA in byte 1. It is verified (UT) that the content error flag / indicator is true and the valid frame flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle. It is also verified (UT) that the frame contents data of slot 1 in the IUT is not updated and corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

4) In cycle 11, the LT simulates its frame with the frame ID set to 1, matching header CRC and with the payload 0x55 in byte 0 and 0xAA in byte 1. It is verified (UT) that the frame ID *vRF!Header!FrameID* is 1 in the frame contents data, the content error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle. It is also verified (UT) that the remainder of the frame contents data of slot 1 in the IUT corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 11.

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The content error flag / indicator and the valid frame flag / indicator in the slot status data and in the aggregated channel status and the frame ID *vRF!Header!FrameID* in the frame contents data are set accordingly. The IUT does not update frame contents data upon reception of invalid frames.

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

7.2.1.8 Payload length

— **Test purpose**

Verify reception of payload length in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations using the modifications as listed in Table 15.

Table 15 — Modification to basic configurations for static segment – payload length

Parameter	Modification to Basic Configurations								
	1a & 1b			2a & 2b			3		
	I	II	III	I	II	III	I	II	III
gdStaticSlot [MT]	274			271			331		
gNumberOfStaticSlots	10			5			4		
gPayloadLengthStatic [two-byte word]	0	64	127	0	64	127	0	64	76
gdNIT [MT]	249			141			137		
gdActionPointOffset [MT]	4			4			3		
pMacroInitialOffset[A] [MT]	6			5			5		
pMacroInitialOffset[B] [MT]	6			5			5		
gMaxWithoutClockCorrectionFatal	3			3			3		

— **Preamble (setup state)**

Preamble I.

— **Test execution**

1) In cycle 8, the LT simulates its frame in slot 1 with a payload length in the frame header equal to *gPayloadLengthStatic.* It is verified (UT) that the length field *vRF!Header!Length* in the frame contents data equals *gPayloadLengthStatic*, the content error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle.

2) In cycle 9, the LT simulates its frame in slot 1 with a payload length in the frame header equal to:

   (I, II) *gPayloadLengthStatic* + 1.

   (III) *gPayloadLengthStatic.*

3) In the NIT of cycle 9, it is verified (UT)

   (I, II) that the frame contents data of slot 1 was not updated, that the content error flag / indicator is true and that the valid frame flag / indicator is false.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

(III) that the frame contents data was updated, that the content error flag / indicator is false and that the valid frame flag / indicator is true

in the slot status data of slot 1 and in the aggregated channel status.

4) In cycle 10, the LT simulates its frame in slot 1 with a payload length in the frame header equal to

(II, III) *gPayloadLengthStatic* – 1.

(I) *gPayloadLengthStatic*.

5) In the NIT of cycle 10, it is verified (UT)

(II, III) that the frame contents data of slot 1 was not updated, that the content error flag / indicator is true and that the valid frame flag / indicator is false

(I) that the frame contents data of slot 1 was not updated, that the content error flag / indicator is false and that the valid frame flag / indicator is true

in the slot status data of slot 1 and in the aggregated channel status.

6) In cycle 11, the LT simulates its frame in slot 1 with a payload length in the frame header equal to *gPayloadLengthStatic*. It is verified (UT) that the length field *vRF!Header!Length* in the frame contents data equals *gPayloadLengthStatic,* the content error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle.

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The content error flag / indicator and the valid frame flag / indicator in the slot status data and in the aggregated channel status and the length field *vRF!Header!Length* in the frame contents data are set accordingly.

7.2.1.9 Cycle count

— **Test purpose**

Verify reception of the cycle counter in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations.

— **Preamble (setup state)**

Preamble I.

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

— **Test execution**

1) In cycle 8, the LT simulates its frame in slot 1 with the cycle counter set to 8. It is verified (UT) that the cycle counter *vRF!Header!CycleCount* is 8 in the frame contents data of slot 1, the content error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle.

2) In cycle 9, the LT simulates its frame in slot 1 with the cycle counter set to 7. It is verified (UT) that the cycle counter *vRF!Header!CycleCount* is 8 (holds the value from the previous cycle) in the frame contents data of slot 1, the content error flag / indicator is true and the valid frame flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle.

3) In cycle 10, the LT simulates its frame in slot 1 with the cycle counter set to 10. It is verified (UT) that the cycle counter *vRF!Header!CycleCount* is 10 in the frame contents data of slot 1, the content error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle.

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The content error flag / indicator and the valid frame flag / indicator in the slot status data and in the aggregated channel status and the cycle counter *vRF!Header!CycleCount* in the frame contents data are set accordingly.

7.2.1.10 Header CRC

— **Test purpose**

Verify reception of the header CRC in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations.

— **Preamble (setup state)**

Preamble I.

— **Test execution**

1) In cycle 8, the LT simulates its frame in slot 1 with a correct header CRC. It is verified (UT) that the header CRC *vRF!Header!HeaderCRC* in the frame contents data equals the header CRC send by the LT in slot 1 of cycle 8, and that the syntax error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

2) In cycle 9, the LT simulates its frame in slot 1 with the payload 0x55 in byte 0 and 0xAA in byte 1 and wrong header CRC (bit flip in the LSB of the header CRC). It is verified (UT) that the syntax error flag / indicator is true and the valid frame flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of this cycle. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

3) In cycle 10, the LT simulates its frame in slot 1 with a correct header CRC. It is verified (UT) that the header CRC *vRF!Header!HeaderCRC* in the frame contents data equals the header CRC send by the LT in slot 1 of cycle 10, and that the syntax error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1and in the aggregated channel status in the NIT of this cycle. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 10.

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The syntax error flag / indicator and the valid frame flag / indicator in the slot status data and in the aggregated channel status and the header CRC *vRF!Header!HeaderCRC* in the frame contents data are set accordingly. If the header CRC of a received frame is wrong, the syntax error flag / indicator has to be true and the frame contents data shall not be updated.

7.2.1.11 FSS / BSS

— **Test purpose**

Verify recognition of frame start sequence FSS and first byte start sequence BSS in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations.

— **Preamble (setup state)**

Preamble I.

— **Test execution**

The LT simulates an additional frame in static slot 3 with the reserved bit, the payload preamble indicator, the null frame indicator, the sync frame indicator and the startup frame indicator set to '0'.

1) In cycle 8, the LT simulates an additional frame in static slot 3 with FSS / BSS being the bit sequence 1-0. It is verified (UT) that the syntax error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status in the NIT of the cycle.

2) In cycle 9, the LT simulates an additional frame in static slot 3 with FSS / BSS being the bit sequence 1-1-0. It is verified (UT) that the syntax error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status in the NIT of the cycle.

3) In cycle 10, the LT simulates an additional frame in static slot 3 with FSS / BSS being the bit sequence 1-1-1-0. It is verified (UT) that the syntax error flag / indicator is false that the valid frame flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status in the NIT of the cycle.

4) In cycle 11, the LT simulates an additional frame in static slot 3 with FSS / BSS being the bit sequence 1-1-1-1-0. It is verified (UT) that the syntax error flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status and that the valid frame flag is false in the slot status data of slot 3 in the NIT of the cycle.

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

5) In cycle 12, the LT simulates an additional frame in static slot 3 with FSS / BSS as a single bit set to 0. It is verified (UT) that the syntax error flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status and that the valid frame flag is false in the slot status data of slot 3 in the NIT of the cycle.

6) In cycle 13, the LT simulates an additional frame in static slot 3 with FSS / BSS as a single bit set to 1. It is verified (UT) that the syntax error flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status and that the valid frame flag is false in the slot status data of slot 3 in the NIT of the cycle.

7) In cycle 14, the LT simulates an additional frame in static slot 3 with FSS / BSS being the bit sequence 0-1. It is verified (UT) that the syntax error flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status and that the valid frame flag is false in the slot status data of slot 3 in the NIT of the cycle.

8) In cycle 15, the LT simulates an additional frame in static slot 3 with FSS / BSS being the bit sequence 1-1. It is verified (UT) that the syntax error flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status and that the valid frame flag is false in the slot status data of slot 3 in the NIT of the cycle.

9) In cycle 16, the LT simulates an additional frame in static slot 3 with FSS / BSS being the bit sequence 1-0-0. It is verified (UT) that the syntax error flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status and that the valid frame flag is false in the slot status data of slot 3 in the NIT of the cycle.

10) In cycle 17, the LT simulates an additional frame in static slot 3 with FSS / BSS being the bit sequence 1-1-0-0. It is verified (UT) that the syntax error flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status and that the valid frame flag is false in the slot status data of slot 3 in the NIT of the cycle.

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The syntax error flag / indicator is set accordingly in the slot status data and in the aggregated channel status. The valid frame flag is set accordingly in the slot status data.

7.2.1.12 Payload

— **Test purpose**

Verify reception of payload data in the static segment.

— **Applicability**

SC, DC.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

— **Configuration**

All basic configurations using the modifications as listed in Table 16.

Table 16 — Modification to basic configurations for static segment – payload

Parameter	Modification to Basic Configurations														
	1a & 1b					2a & 2b					3				
	I	II	III	IV	V	I	II	III	IV	V	I	II	III	IV	V
gdStaticSlot [MT]	274					271					331				
gNumberOfStaticSlots	10					5					4				
gPayloadLengthStatic [two-byte word]	0	1	32	64	127	0	1	32	64	127	0	1	32	64	76
gdNIT [MT]	249					141					137				
gdActionPointOffset [MT]	4					4					3				
pMacroInitialOffset[A,B] [MT]	6					5					5				

— **Preamble (setup state)**

Preamble I.

Deviating from the preamble the LT's payload data is 0x01 in byte 0, 0x02 in byte 1, ..., 0xFE in byte 253. According to the applied payload length *gPayloadLengthStatic*, the LT has to accommodate its frames with respect to the payload length in the frame header and the number of payload bytes in the payload. E.g., if *gPayloadLengthStatic* = 32 two-byte words, the LT simulates its frames with the payload length set to 32 and the payload comprising byte 0 up to byte 63.

— **Test execution**

1) In the NIT of cycle 8, it is verified (UT) that the syntax error flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status and that the received payload data *vRF!Payload* in the frame contents data equals the one transmitted by the LT. If the IUT is not able to receive the whole payload of the frame due to size restrictions of receive buffers, the received bytes – starting with byte 0 – are compared to the transmitted ones. Bytes at the end of the frame exceeding the buffer size may be dropped.

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The syntax error flag / indicator is set accordingly in the slot status data and in the aggregated channel status and the received payload in the frame contents data equals the payload transmitted by the LT.

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

7.2.1.13 BSS

— **Test purpose**

Verify recognition of byte start sequence BSS in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations.

— **Preamble (setup state)**

Preamble I.

— **Test execution**

The LT simulates an additional frame in static slot 3 with the reserved bit, the payload preamble indicator, the null frame indicator, the sync frame indicator and the startup frame indicator set to '0'.

1) In cycle 8, the LT simulates an additional frame in static slot 3 with:

   (a) the BSS after the frame ID in the header segment,

   (b) the BSS after the first data byte Data 0 in the payload segment and

   (c) the BSS after the first CRC byte in the trailer segment

   being the bit sequence 1-0. It is verified (UT) that the syntax error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status in the NIT of the cycle.

2) In cycle 9, the LT simulates an additional frame in static slot 3 with:

   (a) the BSS after the frame ID in the header segment,

   (b) the BSS after the first data byte Data 0 in the payload segment and

   (c) the BSS after the first CRC byte in the trailer segment

   being the bit sequence 0-1. It is verified (UT) that the syntax error flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status and that the valid frame flag is false in the slot status data of slot 3 in the NIT of the cycle.

3) In cycle 10, the LT simulates an additional frame in static slot 3 with:

   (a) the BSS after the frame ID in the header segment,

   (b) the BSS after the first data byte Data 0 in the payload segment and

   (c) the BSS after the first CRC byte in the trailer segment

   being the bit sequence 0-0. It is verified (UT) that the syntax error flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status and that the valid frame flag is false in the slot status data of slot 3 in the NIT of the cycle.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

4) In cycle 11, the LT simulates an additional frame in static slot 3 with:

   (a) the BSS after the frame ID in the header segment,

   (b) the BSS after the first data byte Data 0 in the payload segment and

   (c) the BSS after the first CRC byte in the trailer segment

   being the bit sequence 1-1. It is verified (UT) that the syntax error flag / indicator is true in the slot status data of slot 3 and in the aggregated channel status and that the valid frame flag is false in the slot status data of slot 3 in the NIT of the cycle.

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The syntax error flag / indicator is set accordingly in the slot status data and in the aggregated channel status. The valid frame flag is set accordingly in the slot status data.

7.2.1.14 Frame CRC

— **Test purpose**

Verify reception of frame CRC in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations using the modifications as listed in Table 17.

Table 17 — Modification to basic configuration for static segment – frame CRC

Parameter	Modification
gMaxWithoutClockCorrectionFatal	3

— **Preamble (setup state)**

Preamble I.

Deviating from the Preamble the LT's payload data is 0xAA in byte 0 and 0x55 in byte 1 and 0x00 in all other bytes.

— **Test execution**

1) In cycle 8, the LT simulates its frame in slot 1 correctly. It is verified (UT) that the syntax error flag / indicator is false and that the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of the cycle.

2) In cycle 9, the LT simulates its frame in slot 1 with a bit flip in the first bit of the payload data within the frame without adaption of the frame CRC. It is verified (UT) that the syntax error flag / indicator is true and the valid frame flag / indicator is false in the slot status data of slot 1 and in the aggregated

© ISO 2013 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013

channel status in the NIT of the cycle. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

3) In cycle 10, the LT simulates its frame in slot 1 with the payload 0x55 in byte 0, 0xAA in byte 1 and 0x00 in all other bytes and a bit flip in the first bit of the frame CRC within the frame. In the NIT of this cycle it is verified (UT) that the syntax error flag / indicator is true and the valid frame flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

4) In cycle 11, the LT simulates its frame in slot 1 correctly. It is verified (UT) that the syntax error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of the cycle. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 11.

— **Postamble**

The UT sets the IUT into *POC:halt* state with the CHI command FREEZE.

— **Pass criteria**

The syntax error flag / indicator and the valid frame flag / indicator are set accordingly in the slot status data and in the aggregated channel status. If the frame CRC of a received frame is wrong, the syntax error flag / indicator has to be true, the valid frame flag / indicator has to be false and the frame contents data shall not be updated.

7.2.1.15 FES

— **Test purpose**

Verify recognition of frame end sequence FES in the static segment.

— **Applicability**

SC, DC.

— **Configuration**

All basic configurations.

— **Preamble (setup state)**

Preamble I.

— **Test execution**

1) In cycle 8, the LT simulates its frame in slot 1 with FES being the bit sequence 0-1. It is verified (UT) that the syntax error flag / indicator is false and the valid frame flag / indicator is true in the slot status data of slot 1 and in the aggregated channel status in the NIT of the cycle. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

2) In cycle 9, the LT simulates its frame in slot 1 with FES being the bit sequence 1-0. It is verified (UT) that the syntax error flag / indicator is true and the valid frame flag / indicator is false in the slot status data of slot 1 and in the aggregated channel status in the NIT of the cycle. It is also verified (UT) that the frame contents data of slot 1 corresponds to the contents of the frame as simulated by the LT in slot 1 of cycle 8.

© ISO 2013 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 17458-3:2013