

MOST

Media Oriented Systems Transport

Multimedia and Control
Networking Technology

MOST Core Compliance Test Specification

Rev. 3.0E7

03/2019



Legal Notice

COPYRIGHT

© Copyright 1999 - 2019 MOST Cooperation. All rights reserved.

LICENSE DISCLAIMER

Nothing on any MOST Cooperation Web Site, or in any MOST Cooperation document, shall be construed as conferring any license under any of the MOST Cooperation or its members or any third party's intellectual property rights, whether by estoppel, implication, or otherwise.

CONTENT AND LIABILITY DISCLAIMER

MOST Cooperation or its members shall not be responsible for any errors or omissions contained at any MOST Cooperation Web Site, or in any MOST Cooperation document, and reserves the right to make changes without notice. Accordingly, all MOST Cooperation and third party information is provided "AS IS". In addition, MOST Cooperation or its members are not responsible for the content of any other Web Site linked to any MOST Cooperation Web Site. Links are provided as Internet navigation tools only.

MOST COOPERATION AND ITS MEMBERS DISCLAIM ALL WARRANTIES WITH REGARD TO THE INFORMATION (INCLUDING ANY SOFTWARE) PROVIDED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

In no event shall MOST Cooperation or its members be liable for any damages whatsoever, and in particular MOST Cooperation or its members shall not be liable for special, indirect, consequential, or incidental damages, or damages for lost profits, loss of revenue, or loss of use, arising out of or related to any MOST Cooperation Web Site, any MOST Cooperation document, or the information contained in it, whether such damages arise in contract, negligence, tort, under statute, in equity, at law or otherwise.

FEEDBACK INFORMATION

Any information provided to MOST Cooperation in connection with any MOST Cooperation Web Site, or any MOST Cooperation document, shall be provided by the submitter and received by MOST Cooperation on a non-confidential basis. MOST Cooperation shall be free to use such information on an unrestricted basis.

TRADEMARKS

MOST Cooperation and its members prohibit the unauthorized use of any of their trademarks. MOST Cooperation specifically prohibits the use of the MOST Cooperation LOGO unless the use is approved by the Steering Committee of MOST Cooperation.

SUPPORT AND FURTHER INFORMATION

For more information on the MOST technology, please contact:

MOST Cooperation
Administration
Emmy-Noether-Str. 14
76131 Karlsruhe
Germany

Tel: (+49) (0) 721 966 50 00
E-mail: contact@mostcooperation.com
Web: www.mostcooperation.com



This Specification is Confidential Information of the MOST Cooperation. It may only be disclosed to member companies. Member companies wishing to discuss these Specifications with suppliers or other third parties must ensure that a commercially standard form of non-disclosure agreement has been previously executed by the party receiving such Specifications. Use of these Specifications may only be for purposes for which they are intended by the MOST Cooperation. Unauthorized use or disclosure is a violation of law.

© Copyright 1999 - 2019 MOST Cooperation.
All rights reserved.

MOST is a registered trademark

Contents

1	INTRODUCTION.....	11
2	GENERAL ITEMS.....	12
2.1	Definitions	12
2.1.1	DUT	12
2.1.2	FBlock "EnhancedTestability"	12
2.1.3	Changes Of DUT	13
2.1.4	Product Variants Of DUT / Family	13
2.1.5	Multi Node Devices	14
3	PROCEDURES OF MOST CORE COMPLIANCE TESTS.....	15
3.1	General Notes.....	15
3.1.1	Timer Definitions.....	15
3.1.2	Deadlock Prevention	15
3.1.3	Uninitialized Node Address	15
3.1.4	Address Of Tester	16
3.1.5	DUT Manufacturer Information List	16
3.1.6	Further Definitions - States Of DUT	20
3.1.7	Further Definitions - Procedures	22
3.1.8	Preconditions.....	23
3.2	Overview Of The Experimental Set-up	24
3.2.1	Experimental Set-up.....	24
3.3	Static FBlock Behaviour	25
3.3.1.1	Generic FBlock Property Test (2.1.0-1)	25
3.3.1.2	Generic FBlock Method Test (2.1.0-2)	27
3.4	Wake-up	29
3.4.1	Wake-up – General	29
3.4.1.1	Bypass Test (2.1.1-5)	29
3.4.2	Wake-up – Timing Master	31
3.4.2.1	System Lock Flag Test (a) (2.1.1-6a)	31
3.4.2.2	System Lock Flag Test (b) (2.1.1-6b)	33
3.4.2.3	Wake-up / Shutdown Test (2.1.1-4).....	35
3.4.3	Wake-up – Timing Slave	37
3.4.3.1	Slave Lock Detection Test (2.1.2-5).....	37
3.4.3.2	Slave Wake-up (2.1.3-1).....	39
3.4.3.3	Waking Slave Timeout (2.1.3-4)	41
3.5	Normal Operation	43
3.5.1	Normal Operation – Unlock	43
3.5.1.1	Short Unlock Test (2.2.1-3).....	43
3.5.1.2	Accumulated Short Unlock Test (2.2.1-4)	45
3.5.1.3	Critical Unlock Test (2.2.1-5)	47
3.5.1.4	Sudden Signal Off Test (2.2.1-6)	49
3.5.1.5	Shutdown Flag Present Test (2.2.1-7)	51
3.6	Power Management.....	53
3.6.1	Power Management – Power Master.....	53
3.6.1.1	Timeout Execute / Timeout Suspend (2.3.1-3)	53
3.6.1.2	Timeout Suspend (2.3.1-6)	55
3.6.1.3	Temperature ShutDown Reaction Test (2.3.3-5)	57
3.6.2	Power Management – Power Slave	59
3.6.2.1	ShutDown.Start(Query) Test (2.3.2-2)	59
3.6.2.2	ShutDown.Start(Execute) Test (2.3.2-3).....	61
3.6.3	Power Management – General	63
3.6.3.1	Over-/ Undervoltage Test (2.3.3-1)	63
3.6.3.2	Over-temperature Signal Off Test (2.3.3-4)	65
3.7	Error Management.....	67
3.7.1	Error Management – MOST Signal Path	67
3.7.1.1	Restart Stop Test (2.4.1-1)	67
3.7.1.2	Restart Continue Test (2.4.1-2)	69
3.7.1.3	Reaction Of NCE Test (2.4.1-9).....	71

3.7.1.4	Re-enter Delay Test (2.4.1-11)	73
3.7.2	Error Management – Voltage Level	75
3.7.2.1	Critical Voltage Test (2.4.2-1)	75
3.7.2.2	Low Voltage Test (2.4.2-3)	77
3.8	Ring Break Diagnosis	79
3.8.1.1	RBD Procedure Test (2.5-2) MOST50 ePHY	79
3.8.1.2	TM RBD Procedure Test (2.5-3) MOST50 ePHY	81
3.8.1.3	Wakeup After RBD Test (2.5-7) MOST50 ePHY	83
3.8.1.4	RBD Missing TM Test (2.5-8) MOST50 ePHY	85
3.8.1.5	RBD Closed Ring Test (2.5-10) MOST150 oPHY	87
3.8.1.6	RBD Interruption In Front Of DUT Test (2.5-11) MOST150 oPHY	89
3.8.1.7	RBD Interruption Behind DUT Test (2.5-12) MOST150 oPHY	91
3.9	System Configuration	93
3.9.1	Configuration / System Configuration (NWM)	93
3.9.1.1	NWM Address Initialization Test (2.6.1-1)	93
3.9.1.2	FBlock Polling Test (2.6.2-1)	95
3.9.1.3	Device Ignore Test (a) (2.6.2-3a)	97
3.9.1.4	Device Ignore Test (b) (2.6.2-3b)	99
3.9.1.5	Device Integration Test (a) (2.6.2-4a)	101
3.9.1.6	Device Integration Test (b) (2.6.2-4b)	103
3.9.1.7	Device Integration Test (c) (2.6.2-4c)	105
3.9.1.8	Config(Ok) Delay Test (2.6.2-5)	107
3.9.1.9	Double FBlock Test (2.6.2-6)	109
3.9.1.10	Config(New) Order Test (2.6.2-7)	111
3.9.1.11	FBlock Status Change Detection Test (2.6.3-1)	113
3.9.2	Configuration / System Configuration (Slave)	115
3.9.2.1	Address Initialization Test (2.6.4-1)	115
3.9.2.2	NWM Address Storage Test (2.6.4-3)	117
3.9.2.3	Address Re-Initialization Test (2.6.4-4)	119
3.9.2.4	FBlock Response Time Test (2.6.4-8)	121
3.9.2.5	Control Message Suppression Test (2.6.4-9)	123
3.9.3	Configuration / System Configuration (General)	125
3.9.3.1	InstID Wildcard Test (2.6.4-10)	125
3.10	Node Addressing	127
3.10.1.1	Node Addressing Test (2.7-1)	127
3.10.1.2	Broadcast Error Test (2.8.4-9)	129
3.11	Other Functions	131
3.11.1	Notification Matrix	131
3.11.1.1	Notification Matrix Storage Test (NWM) (2.8.3-1a)	131
3.11.1.2	Notification Matrix Storage Test (Slave) (2.8.3-1b)	133
3.11.1.3	NotificationCheck Test (2.8.3-2)	135
3.11.1.4	Notification Matrix Double Entry Test (2.8.3-7)	137
3.11.1.5	Notification Error Test (2.8.3-10)	139
3.11.2	Message Segmentation	141
3.11.2.1	Segmented Message Rejection Test (2.8.4-1)	141
3.11.2.2	Segmented Message Sending Test (2.8.4-2)	143
3.11.2.3	Message Segmentation Error Test (2.8.4-3)	145
3.11.2.4	Message Segmentation Buffer Test (2.8.4-7)	147
3.11.2.5	Parallel Segmented Message Reception Test (2.8.4-8)	149
3.12	Source / Sink Identification	151
3.12.1.1	TEST_GSI_GSO_Identification (3.0-1)	151
3.13	Obligatory Tests For Sink And Source Devices	153
3.13.1	Sink Devices	153
3.13.1.1	TEST_GSI_SinkInfo (3.1-1)	153
3.13.1.2	TEST_GSI_Connect (3.1-3)	156
3.13.1.3	TEST_GSI_Connect_Repeat (3.1-4)	158
3.13.1.4	TEST_GSI_DisConnect (3.1-5)	160
3.13.1.5	TEST_GSI_DisConnect_Repeat (3.1-6)	162
3.13.2	Source Devices	164
3.13.2.1	TEST_GSO_SourceInfo (3.2-1)	164
3.13.2.2	TEST_GSO_Allocate (3.2-3)	167
3.13.2.3	TEST_GSO_Allocate_Repeat (3.2-4)	169
3.13.2.4	TEST_GSO_Allocate_NoChAvail (3.2-5)	171
3.13.2.5	TEST_GSO_Allocate_ReqChAvail (3.2-6)	173

3.13.2.6	TEST_GSO_Allocate_WrongSourceNr (3.2-7).....	175
3.13.2.7	TEST_GSO_DeAllocate (3.2-8).....	177
3.13.2.8	TEST_GSO_DeAllocate_Repeat (3.2-9)	179
3.13.2.9	TEST_GSO_SourceActivity (3.2-14)	181
4	APPENDIX 1: REQUIREMENTS AND MEASUREMENT UNCERTAINTY FOR TEST EQUIPMENT AND SETUP	183
5	APPENDIX 2: FBLOCK ET REFERENCE TO TEST CASES (INFORMATIVE).....	186
6	APPENDIX 3: DOCUMENT HISTORY	188

Document References

All documents which this MOST document have references to are listed here with the actual revision this document is referring to.

Number	Document	Revision
[1]	MOST Compliance Requirements Specification	2.3
[2]	MOST Dynamic Specification	3.0.2
[3]	MOST Specification	3.0E2
[4]	MOST FunctionBlock "NetBlock"	3.0.4
[5]	MOST FunctionBlock "NetworkMaster"	3.0.2
[6]	MOST FunctionBlock "Enhanced Testability"	3.0.4
[7]	MOST FunctionBlock "General FBlock"	3.0.7

Abbreviations

CR	Central Registry
DR	Decentral Registry
DUT	Device Under Test
I/F	Interface
MOST	Media Oriented Systems Transport
NCE	Network Change Event
NO	NormalOperation
NWM	Network Master
PM	Power Master
TM	Timing Master

Overview Of Core Compliance Tests (Sorted By Test Number)

Test name	Test number	Page
Generic FBlock Property test	2.1.0-1	25
Generic FBlock Method test	2.1.0-2	27
Wake-up / Shutdown test	2.1.1-4	35
Bypass test	2.1.1-5	29
System Lock Flag Test (a)	2.1.1-6a	31
System Lock Flag Test (b)	2.1.1-6b	33
Slave lock detection test	2.1.2-5	37
Slave wake-up	2.1.3-1	39
Waking slave timeout	2.1.3-4	41
Short unlock test	2.2.1-3	43
Accumulated short unlock test	2.2.1-4	45
Critical unlock test	2.2.1-5	47
Sudden Signal Off test	2.2.1-6	49
Shutdown Flag present test	2.2.1-7	51
Timeout execute / Timeout suspend	2.3.1-3	53
Timeout suspend	2.3.1-6	55
ShutDown.Start(Query) test	2.3.2-2	59
ShutDown.Start(Execute) test	2.3.2-3	61
Over-/ undervoltage test	2.3.3-1	63
Over-temperature signal off test	2.3.3-4	65
Temperature ShutDown Reaction Test	2.3.3-5	57
Restart stop test	2.4.1-1	67
Restart continue test	2.4.1-2	69
Reaction of NCE test	2.4.1-9	71
Re-enter delay test	2.4.1-11	73
Critical Voltage test	2.4.2-1	75
Low Voltage test	2.4.2-3	77
RBD procedure test MOST50 ePHY	2.5-2	79
TM RBD procedure test MOST50 ePHY	2.5-3	81
Wakeup after RBD test MOST50 ePHY	2.5-7	83
RBD missing TM test MOST50 ePHY	2.5-8	85
RBD closed ring test MOST150 oPHY	2.5-10	87
RBD interruption in front of DUT test MOST150 oPHY	2.5-11	89
RBD interruption behind DUT test MOST150 oPHY	2.5-12	91
NWM address initialization test	2.6.1-1	93
FBlock polling test	2.6.2-1	95
Device Ignore test (a)	2.6.2-3a	97
Device Ignore test (b)	2.6.2-3b	99
Device Integration test (a)	2.6.2-4a	101
Device Integration test (b)	2.6.2-4b	103
Device Integration test (c)	2.6.2-4c	105
Config(Ok) Delay test	2.6.2-5	107
Double FBlock test	2.6.2-6	109
Config(New) Order test	2.6.2-7	111
FBlock Status Change Detection test	2.6.3-1	113

Test name	Test number	Page
Address Initialization test	2.6.4-1	115
NWM Address Storage test	2.6.4-3	117
Address Re-Initialization test	2.6.4-4	119
FBlock Response Time test	2.6.4-8	121
Control message suppression test	2.6.4-9	123
InstID Wildcard Test	2.6.4-10	125
Node Addressing test	2.7-1	127
Notification Matrix Storage test (NWM)	2.8.3-1a	131
Notification Matrix Storage test (Slave)	2.8.3-1b	133
NotificationCheck test	2.8.3-2	135
Notification Matrix Double Entry test	2.8.3-7	137
Notification Error test	2.8.3-10	139
Segmented Message Rejection test	2.8.4-1	141
Segmented Message Sending test	2.8.4-2	143
Message Segmentation Error test	2.8.4-3	145
Message Segmentation Buffer test	2.8.4-7	147
Parallel Segmented Message Reception test	2.8.4-8	149
Broadcast error test	2.8.4-9	129
TEST_GSI_GSO_Identification	3.0-1	151
TEST_GSI_SinkInfo	3.1-1	153
TEST_GSI_Connect	3.1-3	156
TEST_GSI_Connect_Repeat	3.1-4	158
TEST_GSI_DisConnect	3.1-5	160
TEST_GSI_DisConnect_Repeat	3.1-6	162
TEST_GSO_SourceInfo	3.2-1	164
TEST_GSO_Allocate	3.2-3	167
TEST_GSO_Allocate_Repeat	3.2-4	169
TEST_GSO_Allocate_NoChAvail	3.2-5	171
TEST_GSO_Allocate_ReqChAvail	3.2-6	173
TEST_GSO_Allocate_WrongSourceNr	3.2-7	175
TEST_GSO_DeAllocate	3.2-8	177
TEST_GSO_DeAllocate_Repeat	3.2-9	179
TEST_GSO_SourceActivity	3.2-14	181

1 Introduction

The MOST (Media Oriented Systems Transport) system has been developed for communication systems with the need of high bandwidths like e.g. radio, phone, sound systems, CD, and voice control. Its architecture is based on a ring communication. To ensure a reliable operation of the whole network under all conditions, every single device that is connected to the ring has to fulfill the specification requirements. Devices of different manufacturers have to ensure correct operation in the overall system environment.

In order to verify the behavior of the devices, they have to pass different kinds of tests that check every single core function regarding the network management of the MOST system.

The objective of this document is the definition of all core functions and all tests required in order to prove compliance of a device with the MOST specification.

This document is an informative consolidated version of MOST Core Compliance Test Specification Rev. 3.0 and corresponding Errata Sheets. In case of any doubts these documents are relevant.

2 General Items

2.1 Definitions

2.1.1 DUT

The DUT is the entire device under test, including one external MOST interface. A DUT has at least one MOST node (but it could contain more than one). All tests and timings, specified by the MOST Core Compliance Tests, are always related to the external MOST interfaces.

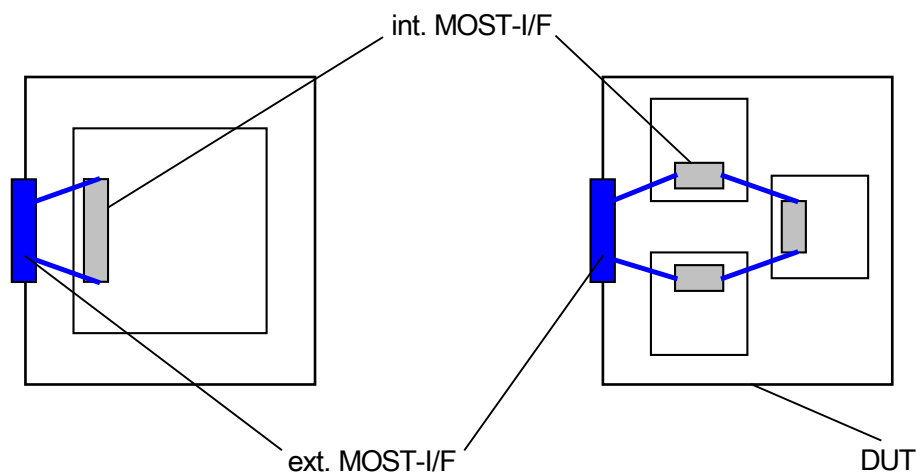


Figure 2-1 DUT with one internal MOST device and a DUT with three internal MOST devices.

If a device contains more than one node, some tests could end with different results, i.e. one for every node. For example, the “Segmented Message Rejection test” could end with two different results as one node supports segmented messages and the other one does not support segmented messages.

2.1.2 FBlock "EnhancedTestability"

This FBlock is used to trigger sequences which have to be tested in the MOST Compliance Test but which are normally triggered by a project specific, sometimes complicated, mechanism. Due to the nature of this FBlock neither notification nor processing messages will be implemented. The FBlock must be initialized every time the NetOn state is reached. The FBlock is only available during NetOn. All properties are reset to their default state when entering NetOn, if not mentioned otherwise. The functions in this FBlock describe a general interface for starting functionality partly implemented in the application, partly in the Network Service. If an application callback returns wrong or unexpected values the FBlock sends a “device malfunction” error message (code 0x0B).

In case the FBlock ET returns wrong values, e.g. maximum number of message buffers available in parallel for message reception, the corresponding test result will be indicated as "Not ok". Due to effort reasons, compliance testing relies on FBlock ET, the test houses has to inform the supplier correspondingly.

This version of MOST Core Compliance Test Specification requires implementation of FBlock ET Rev. 3.0.4.

2.1.3 Changes Of DUT

MOST Compliance Specification defines three classes of changes of a compliant device relative to the potential impact of change and amount of re-qualification required.

For Core Compliance Tests, the following rules are defined how to treat changes of DUT:

As the manufacturer is responsible for the compliance of a device with the standard, he is also responsible for re-testing in case of changes.

Beyond, the system integrator may request additional compliance verification.

Re-Test is mandatory in case of implementation of new MOST Specification version which leads to substantial changes in the Core Compliance test cases relevant for the DUT.

In case of changes the following recommendation for re-test is given:

- Architectural functional changes
- Extensions or reduction of functions that are in the scope of compliance verification.
- Changes of dynamic behaviour which affect timer boundaries and constraints of MOST Specification

2.1.4 Product Variants Of DUT / Family

To minimize test effort for product variants with many similarities, "family" is defined. A family member is a variation of parameters invariant to MOST Compliance Verification.

Such variants can be considered as equivalent "level 1" or "level 2" changes (refer to MOST Compliance Requirements Specification, para 3.9.2 "Guidelines for handling of product changes").

In case a change supersedes the already tested range it has to be considered as "level 3" change (refer to MOST Compliance Requirements Specification, para 3.9.2 "Guidelines for handling of product changes") and consequently cannot be considered as a family member.

Families for core are based on the same hardware. Sufficient coverage of variants must be tested so that each FBlock has been involved. The most complex variants should be preferred for testing.

Guideline „Improved Product ID Handling“

The MOST Compliance process requires a unique ProductID in order to enable a distinction of MOST products. The following guideline gives some advice how to handle it.

I) Use cases

A) A device will get a new SW

→ The Product ID may stay the same.

B) Different Coding of the device (parts of the SW will be set to "passive")

→ The Product ID may stay the same.

C) A device will be used in several countries (The same (MOST-)HW with different "Non MOST HW")

→ The Product ID should reflect the different variants.

D) The branding of the device has changed (e.g, new surface on the screen)

→ The Product ID should reflect the different variants.

IIa) Example for appropriate ProductIDs:

1) A unique ProductID should consist of at least 4 characters e.g.: 1234, ABC1, 0816, ABCD

IIb) Examples for inappropriate ProductIDs:

1) n.a.

2) STD_MIB

3) Gateway

2.1.5 Multi Node Devices

A multi node device is a device with an external MOST interface which is connected to more than one MOST node.

Remark: When dealing with devices with several external MOST interfaces each MOST interface shall be treated as a separate DUT. Realize, one of those interfaces may be a Multi Node Device.

If a Multi Node Device consists of several independent nodes, each node will be treated individually:

1 TM and (at least) 1 slave node:	M: All tests for TM S: All test for slave but not "All devices except TM"
1 Master (NWM) node and (at least) 1 slave node:	M: All tests for NWM S: All test for slave but not "All devices except NWM"
1 Master (PM) node and (at least) 1 slave node:	M: All tests for PM S: All test for slave but not "All devices except PM"
1 Master (TM/NWM) node and (at least) 1 slave node:	M: All tests for NWM/TM S: All test for slave but not "All devices except TM" but not "All devices except NWM"
1 Master (TM/PM/NWM) node and (at least) 1 slave node:	M: All tests for PM /NWM/TM S: All test for slave but not "All devices except PM" but not "All devices except TM" but not "All devices except NWM"
Several slave nodes:	S: All test for slave

The DUT passes the test successfully if all testable nodes of the DUT has passed the test successfully.

3 Procedures Of MOST Core Compliance Tests

3.1 General Notes

3.1.1 Timer Definitions

Names of Timer, used by this Core Compliance Test Specification are based on the MOST Specification. Compared to the MOST Specification, an underscore is used to separate the timer name.

Example:

Name	Min Value	Typ Value	Max Value	Unit	Type	Definition
t _{Config}	t_Config_min	t_Config	t_Config_max	ms	T	Time that may pass after initialization of the MOST Network Interface Controller in a device until transition to NetOn state.

3.1.2 Deadlock Prevention

To prevent a deadlock during testing, three timeouts have been defined:

- t_DeadLockShort = 1s
- t_DeadLockMid = 20s
- t_DeadLockLong = 5 min

These timeouts are not defined by MOST Specification. They are only relevant for compliance testing. If a DUT fails a test due to timeout of a deadlock timer, the test could be performed again (after consultation with manufacturer) with extended timeout for deadlock timer. Extended timeouts of deadlock timer have to be indicated in the test results.

3.1.3 Uninitialized Node Address

This test specification uses the variable "uninitialized_node_address" to identify the address of an uninitialized node. Use always the value that is specified in the MOST Specification, the DUT is based on.

3.1.4 Address Of Tester

Default address of the tester is default logical address corresponding to the ring position.
If DUT uses an address normally used by a tester (e.g. in case of supplier, using static addresses in dynamic address range), the tester has to use another address from free address range).

3.1.5 DUT Manufacturer Information List

This list contains all information that have to be provided by the device manufacturer for compliance testing. It also includes remarks and references to corresponding Core Compliance Test Cases.

MCTHs can use information from this list to support the manufacturers providing test relevant information.

Information, normally stored in FBlock ET, are not included in this table. Description of this information can be derived directly from FBlock ET.

Item / Property	Note / Remark	Reference to Core Compliance Test Case
MOST Network Configuration		
MOST Specification	MOST Specification Rev 3.0 E2 required	All test cases
MOST Physical Layer	"MOST50 ePHY", "MOST150 oPHY" or "MOST150 cPHY"	All test cases
DUT contains TimingMaster		All test cases
DUT contains NetworkMaster		All test cases
DUT contains PowerMaster		All test cases
DUT contains ConnectionMaster		3.1-3 (TEST_GSI_Connect) 3.1-4 (TEST_GSI_Connect_Repeat) 3.1-5 (TEST_GSI_DisConnect) 3.1-6 (TEST_GSI_DisConnect_Repeat) 3.2-3 (TEST_GSO_Allocate) 3.2-4 (TEST_GSO_Allocate_Repeat) 3.2-5 (TEST_GSO_Allocate_NoChAvail) 3.2-6 (TEST_GSO_Allocate_ReqChAvail) 3.2-7 (TEST_GSO_Allocate_WrongSourceNr) 3.2-8 (TEST_GSO_DeAllocate) 3.2-9 (TEST_GSO_DeAllocate_Repeat) 3.2-14 (TEST_GSO_SourceActivity)
Multi Node Device	If DUT contains more than one node, the following information should be provided: - How many nodes does the DUT contain - Topology of DUT (position of PM, TM, NWM and/or multi-node, if applicable) - Position of Node to be tested	All test cases
DUT sample frequency	If DUT is not TM, the tester has to provide MOST signal with the correct sample frequency (44.1kHz or 48.0 kHz).	All test cases

Item / Property	Note / Remark	Reference to Core Compliance Test Case
Required value of Boundary Descriptor (in case tester = TM)	Value of Boundary Descriptor, specified by system integrator (System Boundary Descriptor). If not otherwise stated, all tests will be performed with this value of Boundary Descriptor.	All test cases
Power Management (Sleep Mode, Wake-up)		
Device supports SleepMode	Manufacturer has to provide whether DUT supports SleepMode: Yes: The device has to reduce its power consumption below threshold before timeout expires No: The reduction of power consumption is not detectable	para 3.1 2.3.2-3 (ShutDown.Start(Execute) test)
SleepMode: I_SleepMode_Threshold	Power consumption in SleepMode	para 3.1
SleepMode: t_PwrSwitchOffDelay_max	Timeout for SleepMode; which period of time has to be exceeded from "MOST signal off at Rx" until DUT enters SleepMode resp. reduces power consumption.	para 3.1
Wakeup method: DUT is wakeable via MOST signal	DUT designed to be woken by MOST signal (defined as MOST signal on).	2.1.1-5 (Bypass test) 2.2.1-5 (Critical unlock test) 2.5-7 (Wakeup after RBD test MOST50 ePHY)
Wakeup method: DUT is electrical wakeable	DUT supports electrical wakeup. How to generate electrical wakeup (e.g. signal form and duration).	2.2.1-5 (Critical unlock test) 2.5-7 (Wakeup after RBD test MOST50 ePHY)
Wakeup preconditions	Preconditions that have to be applied to the DUT for wake-up (e.g. active CAN interface) Additionally information whether DUT needs special conditions during operation (e.g. KL15) to stay in normal operation.	para 3.1
DUT is capable to wake via MOST signal		2.1.1-6a (System Lock Flag Test (a)) 2.1.1-4 (Wake-up / Shutdown test) 2.1.3-1 (Slave wake-up) 2.1.3-4 (Waking slave timeout) 2.4.1-1 (Restart stop test) 2.4.1-2 (Restart continue test)
Delay between connection to power (of DUT) and ability of DUT to detect wakeup	Potentially, the tester has to wait for a short period of time between connecting the DUT to power and generating MOST signal to wake-up the DUT. Otherwise, the DUT fails to detect wake-up event.	
NormalOperation: Delay until all FBlocks of the DUT are available after Configuration.Status(Ok) (= t_WaitForApplication)	This delay covers: NetworkMaster: Period of time, the DUT needs to register own FBlocks into its CR after Configuration.Status(Ok) NetworkSlave: Delay between "NetOn" and availability of application	Para 3.1.6 (DUT: Normal Operation) 2.6.2-3b (Device Ignore test (b))
t_WaitBeforeRescan_max		2.6.2-5 (Config(Ok) Delay test)
t_WaitAfterOvertempShutdown_min		2.3.3-5 (Temperature ShutDown Reaction Test)

Item / Property	Note / Remark	Reference to Core Compliance Test Case
PM supports suspend	If the PM does not support suspend the following test case will be omitted: - 2.3.1-3 (Timeout execute / Timeout suspend) - 2.3.1-6 (Timeout suspend)	2.3.1-3 (Timeout execute / Timeout suspend) 2.3.1-6 (Timeout suspend)
RBD		
RBD Phase 3 support	DUT supports RBD phase 3. Input only relevant for MOST150 oPHY devices.	2.5-10 (RBD closed ring test MOST150 oPHY) 2.5-11 (RBD interruption in front of DUT test MOST150 oPHY) 2.5-12 (RBD interruption behind DUT test MOST150 oPHY)
t_DiagResultAvailable	Manufacturer dependent time interval when ET.DiagResult is available. Information is only necessary in case RBD phase 3 is not supported. The time interval shall be given with respect to the end of RBD phase 2. Input only relevant for MOST150 oPHY devices. In case DUT supports RBD phase 3: t_DiagResultAvailable = 0ms	2.5-10 (RBD closed ring test MOST150 oPHY) 2.5-11 (RBD interruption in front of DUT test MOST150 oPHY)
MOST150 oPHY device: Performing optional RBD tests	For MOST150 oPHY, RBD tests are optional. This DUT manufacturer information will state if compliance testing includes RBD tests. Input only relevant for MOST150 oPHY devices.	2.5-10 (RBD closed ring test MOST150 oPHY) 2.5-11 (RBD interruption in front of DUT test MOST150 oPHY) 2.5-12 (RBD interruption behind DUT test MOST150 oPHY)
Addressing		
DUT uses static node address in dynamic address range	If the DUT uses a static node address, that is in the specified dynamic address range, the node address has to be provided	para 3.1 2.6.4-4 (Address Re-Initialization test)
Free address	Logical node address that can be used by the tester during testing.	2.6.4-3 (NWM Address Storage test)
Free FBlock range	FBlocks that are not used by the DUT and which can be used by the tester (the range is determined by the size of the central registry divided by 254)	2.6.2-4b (Device Integration test (b)) 2.6.2-4c (Device Integration test (c))
Group address of DUT	The group address of the DUT, specified by manufacturer	2.7-1 (Node Addressing test)
Physical Parameter (Voltage levels)		
U_DUT_Operating_max	The highest voltage level the DUT operates normally.	2.3.3-1 (Over-/ undervoltage test) 2.4.2-1 (Critical Voltage test)
U_DUT_Operating	At this voltage level, the DUT operates normally. If not other stated, all compliance tests will be performed at this voltage level.	All test cases
U_Critical	When reducing supply voltage (starting from U_DUT_Operating), the limit at which the application will no longer work safely but where communication is still possible.	2.4.2-1 (Critical Voltage test)
U_Low	When reducing supply voltage (starting from U_DUT_Operating), the device specific limit, where even the NetInterface no longer works reliably, so even communication cannot be maintained.	2.3.3-1 (Over-/ undervoltage test) 2.4.2-3 (Low Voltage test)
Messaging		

Item / Property	Note / Remark	Reference to Core Compliance Test Case
LowLevel and MidLevel retries	Number and Timing of LowLevel and MidLevel retries (not total transmission attempts) has to be provided by manufacturer for function NetBlock.FBlockIDs in NetInterface Normal Operation.	2.6.2-3b (Device Ignore test (b))
DUT supports segmented messages	DUT is able to send and receive segmented messages	2.8.4-2 (Segmented Message Sending test) 2.8.4-3 (Message Segmentation Error test) 2.8.4-7 (Message Segmentation Buffer test) 2.8.4-8 (Parallel Segmented Message Reception test)
Sink / Source		
List of FBlocks, containing sink and/or source functionality	The list has to consider all FBlocks, reported by FBlockID.Status	3.0-1 (TEST_GSI_GSO_Identification)
Sink devices: List of all supported sink numbers with ContentType, ContentDescription (data type of the parameter) and TransmissionClass.		3.1-1 (TEST_GSI_SinkInfo) 3.1-3 (TEST_GSI_Connect) 3.1-4 (TEST_GSI_Connect_Repeat) 3.1-5 (TEST_GSI_DisConnect) 3.1-6 (TEST_GSI_DisConnect_Repeat)
Source devices: List of all supported source numbers with ContentType, ContentDescription (data type of the parameter) and TransmissionClass		3.2-1 (TEST_GSO_SourceInfo) 3.2-3 (TEST_GSO_Allocate) 3.2-4 (TEST_GSO_Allocate_Repeat) 3.2-5 (TEST_GSO_Allocate_NoChAvail) 3.2-6 (TEST_GSO_Allocate_ReqChAvail) 3.2-7 (TEST_GSO_Allocate_WrongSourceNr) 3.2-14 (TEST_GSO_SourceActivity)
Source devices: BlockWidth and ConnectionLabel		3.2-3 (TEST_GSO_Allocate)
DUT supports SourceActivity		3.2-14 (TEST_GSO_SourceActivity)

3.1.6 Further Definitions - States Of DUT

DUT: NormalOperation	
Def.: NetBlock must be available "DUT: NormalOperation" equal to "DeviceNormalOperation" of MOST Specification	
Effectuate state of DUT	Detect state of DUT
<p>a) DUT is slave device: Tester has to behave according to MOST Specification and start the system normally:</p> <ul style="list-style-type: none"> - Wait for DUT opens its bypass (MPR = MPR_nominal) - Sending Configuration.Status(NotOk) - perform system scan (including retries in case of invalid address of DUT) - Sending Configuration.Status(Ok) - Wait for t_WaitForApplication <p>b) DUT is NWM device: Tester has to behave like a normal slave device. It has to process and respond all requests from DUT to enable DUT entering Configuration.Status(Ok). The tester has to respond network scan by DUT with empty FBlock list. (alternatively, the tester could register some FBlocks). Additionally, the tester has to wait for relevant node opens its bypass (MPR = MPR_nominal) in case the DUT is a Multi Node Device. Finally, tester has to wait for t_WaitForApplication.</p>	<p>a) DUT is slave device: DUT has to respond to FBlockIDs.Get</p> <p>b) DUT is NWM device: DUT has to respond to FBlockIDs.Get</p>

DUT: SleepMode	
Effectuate state of DUT	Detect state of DUT
<p>a) DUT is power master: - Switch off MOST signal</p> <p>b) DUT is power slave: - Switch off MOST signal</p>	<p>The manufacturer has to provide:</p> <ul style="list-style-type: none"> - whether DUT supports SleepMode - t_PwrSwitchOffDelay_max: Timeout for SleepMode; which period of time has to be exceed from "MOST signal off at Rx" until DUT enters SleepMode) - I_SleepMode_Threshold : Threshold of current for SleepMode detection) <p>a) DUT is power master: If DUT supports SleepMode: Monitoring power consumption. The current has to reach or drop below I_SleepMode_Threshold. If DUT generates MOST signal at Tx although timeout t_PwrSwitchOffDelay_max expires, the test will be stopped ("DUT not ok: No SleepMode possible"). The timer for timeout will be started as soon as DUT switches off MOST signal at Tx for the first time. If current does not reach or drop below I_SleepMode_Threshold within timeout t_PwrSwitchOffDelay_max, the test will be stopped ("DUT not ok: No SleepMode possible").</p> <p>If DUT does not support SleepMode: As soon as the timeout t_PwrSwitchOffDelay_max expires and DUT does not generate MOST signal at Tx, it will be assumed that DUT has entered SleepMode, independent from power consumption. If DUT generates MOST signal at Tx although t_PwrSwitchOffDelay_max expires, the test will be stopped ("DUT not ok: No SleepMode possible").</p> <p>For MOST50 ePHY, no monitoring of activity performed.</p> <p>b) DUT is power slave: refer to "DUT is power master"</p>

DUT: NetInterface Off	
Effectuate state of DUT	Detect state of DUT
Switch off tester MOST signal	DUT does not generate MOST signal

DUT: RBD finished	
Effectuate state of DUT	Detect state of DUT
<p>Only start of RBD can be triggered. - Trigger RBD via FB ET (AutoWakeUp)</p> <p>Note: Some devices does not finish RBD automatically. They need a trigger event to finish RBD. This trigger has to be generated by FB ET (by means of timer). The test equipment does not provide interface for that trigger.</p>	<p>DUT switches off MOST signal (after RBD).</p>

3.1.7 Further Definitions - Procedures

Procedure	Procedure of tester
Perform Wake-up	<p>a) Tester is TM:</p> <ol style="list-style-type: none"> 1. Generate MOST signal 2. Wait for MOST signal from DUT (timeout t_DeadLockShort) 3. Wait for stable lock (timeout t_DeadLockMid) <p>b) Tester is not TM:</p> <ol style="list-style-type: none"> 1. Generate wake-up event (e.g. slave wakeup including t_Restart) 2. Wait for MOST signal at input 3. Generate MOST signal 4. Wait for stable lock 5. Wait for System Lock Flag set <p>Notes:</p> <p>If DUT already generates MOST signal, it will be sufficient that tester switches on MOST signal immediately.</p> <p>In some cases, the DUT needs some pre-conditions (e.g. active CAN interface) for wake-up. The pre-conditions depend on the manufacturer. These pre-conditions have to be provided to the DUT before start of testing.</p> <p>If the AutoWakeup is triggered via FBlock ET, the DUT does not enter the normal SleepMode before performing the wake-up. The DUT enters "NetInterface Off". This mode cannot be detected by monitoring the power consumption of the DUT. With entering state "NetInterface Off", the DUT switches off the MOST signal at its output.</p>

Procedure	Procedure of tester
Perform ShutDown	<p>a) DUT is power master:</p> <ul style="list-style-type: none"> - Trigger ShutDown by means of FB ET - If MOST signal is off, DUT has performed ShutDown <p>If t_DeadLockMid expires after triggering ShutDown and DUT continues generating MOST signal, the tester switches off the MOST signal by itself.</p> <p>If DUT does not support FB ET: Tester switches off the MOST signal by itself as ShutDown cannot be triggered via FB ET.</p> <p>b) DUT is power slave:</p> <ul style="list-style-type: none"> - Switch off MOST signal <p>NOTE: If DUT is PM, preconditions have to be switched off that could prevent DUT from performing ShutDown.</p>
Generate Unlock	<p>To generate an unlock event of predictable duration two requirements must be met:</p> <p>First the preamble at the beginning of at least every third MOST frame has to be made invalid or delayed during the period of unlock.</p> <p>Second a PLL unlock must be avoided.</p>
Trigger RBD	<p>During NormalOperation, the function "AutoWakeup" (0x201) of FB ET has to be called with parameter "Diagnosis = True".</p>
MOST signal on	<p>Tester switches MOST signal on (MOST signal = modulated signal according to MOST Specification)</p>
MOST signal off	<p>Tester switches MOST signal off (MOST signal = modulated signal according to MOST Specification)</p>
NCE with unlock	<p>The NCE has to be generated between TM and DUT.</p>
NCE without unlock	<p>The NCE has to be generated between DUT and TM</p>

3.1.8 Preconditions

If it is proven that the DUT is not possible to reach preconditions (such as modulated signal, lock, scan performed, Config ok), the test case will be marked as "DUT not Ok".

3.2 Overview Of The Experimental Set-up

All MOST Core Compliance Tests require the same experimental set-up with two tester. Every tester is able to operate in master mode, slave mode and spy mode. Every compliance test contains a description of the experimental set-up that describes the operating mode of tester 1 and tester 2.

To avoid overtemperature situations of DUT during testing, the manufacturer recommendation regarding to cooling have to be met.

3.2.1 Experimental Set-up

The set-up consists of the DUT and two tester. A spy node in front of each tester is connected to the MOST ring to log the whole communication. The power supply of the DUT is adjustable and the power consumption of the DUT can be monitored.

Each tester can be set into master mode, slave mode or spy mode. Master mode means, the tester behaves like a TimingMaster. Slave mode means, the tester behaves like a TimingSlave. Spy mode means, the tester behaves like a listen only node.

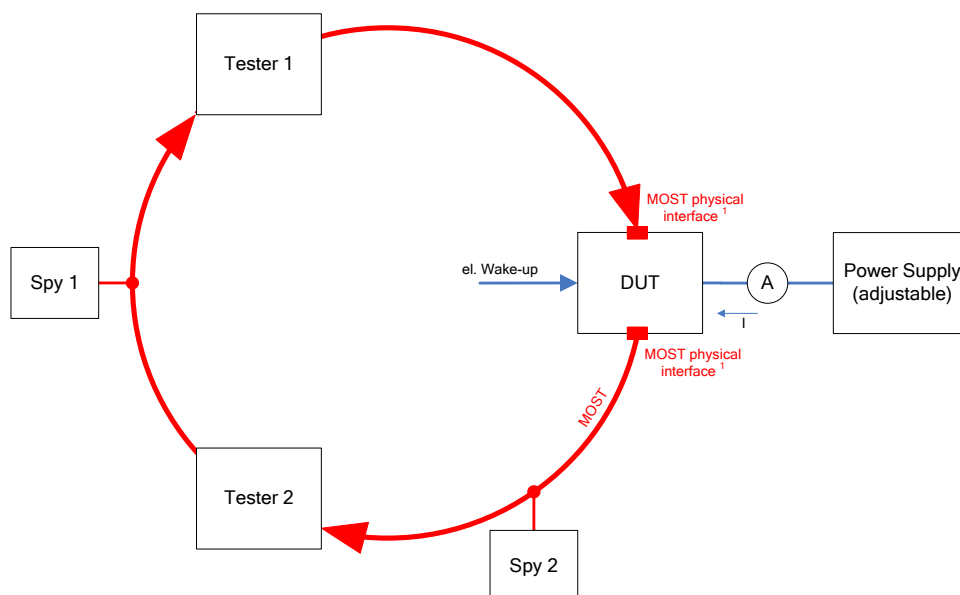


Figure 3-1 Experimental setup

¹ When testing a MOST150 cPhy device (simplex or duplex), an appropriate C-O converter (Coax-Optical converter) can be used to connect the DUT to the existing set-up.

3.3 Static FBlock Behaviour

3.3.1.1 Generic FBlock Property Test (2.1.0-1)

Name of test	Generic FBlock Property test 2.1.0-1
Reference to MOST Specification	2.2.3.5 OType
Value of Interest	
Start Conditions	DUT in NormalOperation
Test description	
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices
Note	<p>Test has to be performed with the following FBlocks of the DUT:</p> <ul style="list-style-type: none"> - NWM - NetBlock - FBlocks containing sink functions (identified by test TEST_GSI_GSO_Identification) - FBlocks containing source functions (identified by test TEST_GSI_GSO_Identification) <p>FBlockID defined by DUT (e.g. NetBlock) InstID: All InstIDs as reported by DUT (FBlockID.Status) FktID: All FktIDs reported by DUT (FktID.Status) and also listed in relevant FBlock Library (NWM, GeneralFBlock) of MOSTCo as well. Additionally, All FCat Fkts. will be checked of NetBlock.</p> <p>Test covers all OP-Types (0 to F). Responses will be only evaluated from actually tested FBlockID, InstID and FktID. Each other message of the DUT will be ignored during the test. The FBlock Library gives evidence whether it is a property respectively a method.</p> <p>Note: Today a device may not implement all functions and OTypes of the FBlock Library. If implemented, the functions / OTypes must behave according to the FBlock Library.</p> <p>Note: There will be no range checks.</p> <p>Note: In case the OType test allows "None" as "permitted answer", the observation time period will be 2x t_Property for properties respectively 2 x t_ProcessingDefault1_max for methods.</p> <p>Note: For NetBlock, Error 0x03 (Fkt. not available) is also permitted as answer.</p> <p>Note: All commands from DUT (OType 0x0 .. 0x8) will be ignored for all FBlocks.</p> <p>Note: For Netblock and NMW FBlock, OTypes 0x09 ... 0xF not tested.</p>
Results	<p>DUT ok: The DUT has passed the test.</p> <p>DUT not ok: The DUT fails to pass the test.</p>

OPType	Send to DUT	Permitted answer from DUT
0 (Set)	Set (without parameter)	- Status - None (within 2x t_property) - All Errors except 0x01,0x02,0x03,0x0A,0x0C
1 (Get)	Get(without parameter)	- Status - All Errors except 0x01,0x02,0x03,0x0A,0x0C
2 (SetGet)	SetGet(without parameter)	- Status - All Errors except 0x01,0x02,0x03,0x0A,0x0C
3 (Increment)	Increment(1)	- Status - All Errors except 0x01,0x02,0x03,0x0A,0x0C
4 (Decrement)	Decrement(1)	- Status - All Errors except 0x01,0x02,0x03,0x0A,0x0C
5 (GetInterface)	GetInterface	- Interface - All Errors except 0x01,0x02,0x03,0x0A,0x0C
6 (M: StartResultAck)	StartResultAck (SenderHandle=0x12,0x34)	- ErrorAck(SenderHandle=0x12,0x34, ErrorCode 0x04) - Error(ErrorCode 0x04)
7 (M: AbortAck)	AbortAck (SenderHandle=0x12,0x34)	- ErrorAck(SenderHandle=0x12,0x34, ErrorCode 0x04) - Error(ErrorCode 0x04)
8 (M: StartAck)	StartAck (SenderHandle=0x12,0x34)	- ErrorAck(SenderHandle=0x12,0x34, ErrorCode 0x04) - Error(ErrorCode 0x04)
9 (M: ErrorAck)	ErrorAck	None
A (M: ProcessingAck)	ProcessingAck	None
B (M: Processing)	Processing	None
C (Status)	Status	None
D (M: ResultAck)	ResultAck	None
E (Interface)	Interface	None
F (Error)	Error	None

3.3.1.2 Generic FBlock Method Test (2.1.0-2)

Name of test	Generic FBlock Method test 2.1.0-2
Reference to MOST Specification	2.2.3.5 OPType
Value of Interest	
Start Conditions	DUT in NormalOperation
Test description	
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices
Note	<p>Test has to be performed with the following FBlocks of the DUT:</p> <ul style="list-style-type: none"> - NWM - NetBlock - FBlocks containing sink functions (identified by test TEST_GSI_GSO_Identification) - FBlocks containing source functions (identified by test TEST_GSI_GSO_Identification) <p>FBlockID defined by DUT (e.g. NetBlock) InstID: All InstIDs as reported by DUT (FBlockID.Status) FktID: All FktIDs reported by DUT (FktID.Status) and also listed in relevant FBlock Library (NWM, GeneralFBlock) of MOSTCo as well. Additionally, All FCat Fkts. will be checked of NetBlock.</p> <p>Test covers all OP-Types (0 to F). Responses will be only evaluated from actually tested FBlockID, InstID and FktID. Each other message of the DUT will be ignored during the test. The FBlock Library gives evidence whether it is a property respectively a method.</p> <p>Note: Today a device may not implement all functions and OPTypes of the FBlock Library. If implemented, the functions / OPTypes must behave according to the FBlock Library.</p> <p>Note: There will be no range checks.</p> <p>Note: Each OPType test presumes that the previous test has completed the communication process, e.g. if after StartResult the Result is not received, then the process has to be finished by sending Abort. In case Abort fails, the DUT can be reset to finish the test step and continue with the next step. In case the OPType test allows "None" as "permitted answer", the observation time period will be 2x t_Property for properties respectively 2 x t_ProcessingDefault1_max for methods.</p> <p>Note:In case the OPType test allows "Processing" as "permitted answer": As response „Processing“ might be received before Result or an Error, max. 50 Processing will be accepted by the tester, then "Abort" will be sent.</p> <p>Note: For NetBlock, Error 0x03 (Fkt. not available) is also permitted as answer</p> <p>Note: All commands from DUT (OPType 0x0 .. 0x8) will be ignored for all FBlocks.</p> <p>Note: For Netblock and NMW FBlock, OPTypes 0x09 ... 0xF not tested.</p> <p>Note: During this test case it must be considered that sending of an error is optional in function DTCP_Control.</p>
Results	<p>DUT ok: The DUT has passed the test.</p> <p>DUT not ok: The DUT fails to pass the test.</p>

OPType	Send to DUT	Permitted answer from DUT
0 (Start)	Start	<ul style="list-style-type: none"> - None - Result - Processing - All Errors except 0x01,0x02,0x03,0x0A,0x0C
1 (Abort)	Abort	<ul style="list-style-type: none"> - None - All Errors except 0x01,0x02,0x03,0x0A,0x0C
2 (StartResult)	StartResult	<ul style="list-style-type: none"> - Processing - Result - All Errors except 0x01,0x02,0x03,0x0A,0x0C
3 (P: Increment)	Increment	<ul style="list-style-type: none"> - Error 0x04
4 (P: Decrement)	Decrement	<ul style="list-style-type: none"> - Error 0x04
5 (GetInterface)	GetInterface	<ul style="list-style-type: none"> - Interface - All Errors except 0x01,0x02,0x03,0x0A,0x0C
6 (StartResultAck)	StartResultAck (SenderHandle=0x12,0x34)	<ul style="list-style-type: none"> - ProcessingAck(SenderHandle=0x12,0x34) - ResultAck(SenderHandle=0x12,0x34) - All ErrorAcks (SenderHandle=0x12,0x34) except 0x01,0x02,0x03,0x0A,0x0C - All Errors except 0x01,0x02,0x03,0x0A,0x0C
7 (AbortAck)	AbortAck (SenderHandle=0x12,0x34)	<ul style="list-style-type: none"> - None - All ErrorAcks (SenderHandle=0x12,0x34) except 0x01,0x02,0x03,0x0A,0x0C - All Errors except 0x01,0x02,0x03,0x0A,0x0C
8 (StartAck)	StartAck (SenderHandle=0x12,0x34)	<ul style="list-style-type: none"> - None - ResultAck(SenderHandle=0x12,0x34) - ProcessingAck(SenderHandle=0x12,0x34) - All ErrorAcks (SenderHandle=0x12,0x34) except 0x01,0x02,0x03,0x0A,0x0C - All Errors except 0x01,0x02,0x03,0x0A,0x0C
9 (ErrorAck)	ErrorAck	None
A (ProcessingAck)	ProcessingAck	None
B (Processing)	Processing	None
C (Result)	Result	None
D (ResultAck)	ResultAck	None
E (Interface)	Interface	None
F (Error)	Error	None

3.4 Wake-up

3.4.1 Wake-up – General

3.4.1.1 Bypass Test (2.1.1-5)

Name of test	Bypass test 2.1.1-5
Reference to MOST Specification	3.1.2.2.2 NetInterfaceInit
Value of Interest	t_Lock + t_WaitNodes Bypass
Start Conditions	DUT in SleepMode
Test description	MOST signal is applied to the input of the DUT. The time between MOST signal on and MPR = MPR_Nominal must not exceed t_Lock_max + t_WaitNodes_max.
Experimental set-up	- Tester 1 in master mode or slave mode; depends on DUT - Tester 2 in spy mode
Device type	All devices that are wakeable via MOST signal [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices that are wakeable via MOST signal except TM and PM
Note	The test can be applied for master and for slave devices. Test only applicable for devices, wakeable by MOST signal.
Results	DUT ok: The DUT has passed the test. DUT not ok: The DUT fails to open the bypass right on time.

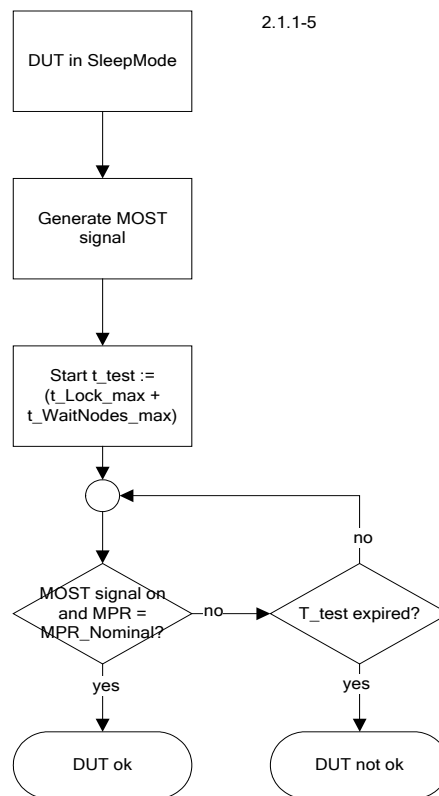


Figure 3-2 Flow of Bypass test.

3.4.2 Wake-up – Timing Master

3.4.2.1 System Lock Flag Test (a) (2.1.1-6a)

Name of test	System Lock Flag Test (a) 2.1.1-6a
Reference to MOST Specification	3.1.2.2.2 NetInterfaceInit
Value of Interest	System Lock Flag
Start Conditions	DUT in NormalOperation
Test description	A wake-up event is triggered and the ring is interrupted in front of the DUT. As long as the DUT generates MOST signal at output, the System Lock Flag must not be set.
Experimental set-up	- Tester 1 in slave mode - Tester 2 in spy mode
Device type	TM, which is capable to wake via MOST signal. [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices containing TM, which are capable to wake via MOST signal
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok: System Lock Flag erroneously set without stable lock.

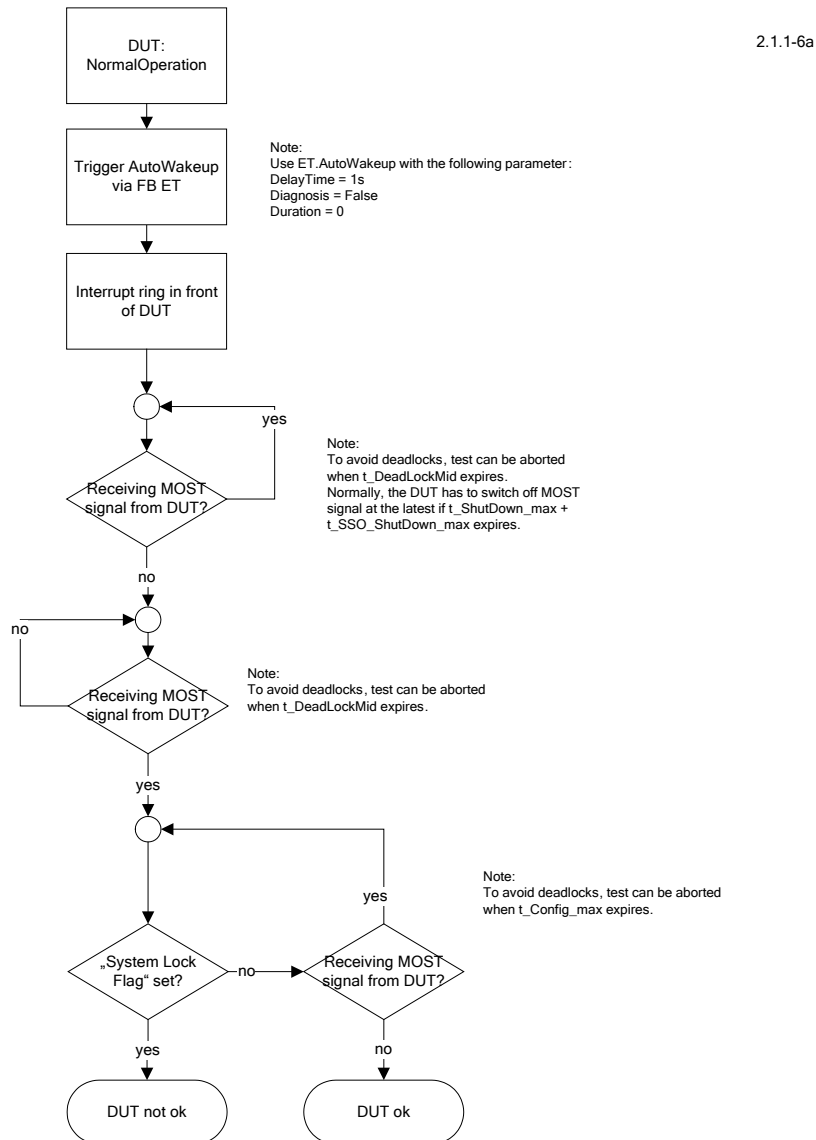


Figure 3-3 Flow of System Lock Flag test (a).

3.4.2.2 System Lock Flag Test (b) (2.1.1-6b)

Name of test	System Lock Flag Test (b) 2.1.1-6b
Reference to MOST Specification	3.1.2.2.2 NetInterfaceInit
Value of Interest	System Lock Flag T_Config_max
Start Conditions	DUT: NetInterface Off Ring closed
Test description	A wake-up event is triggered. The output of the tester is disabled (ring interrupted in front of DUT). System Lock Flag must not be set by DUT. If the tester detects a stable lock, the ring will be closed (by activating of the tester's output). The System Lock Flag must be set within t_Config_max.
Experimental set-up	- Tester 1 in slave mode - Tester 2 in spy mode
Device type	TM [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices containing TM
Note	
Results	DUT ok: The DUT has passed the test DUT not ok (1): Timeout t_Config without stable lock DUT not ok (2): System Lock Flag erroneously set DUT not ok (3): System Lock Flag fails to be set

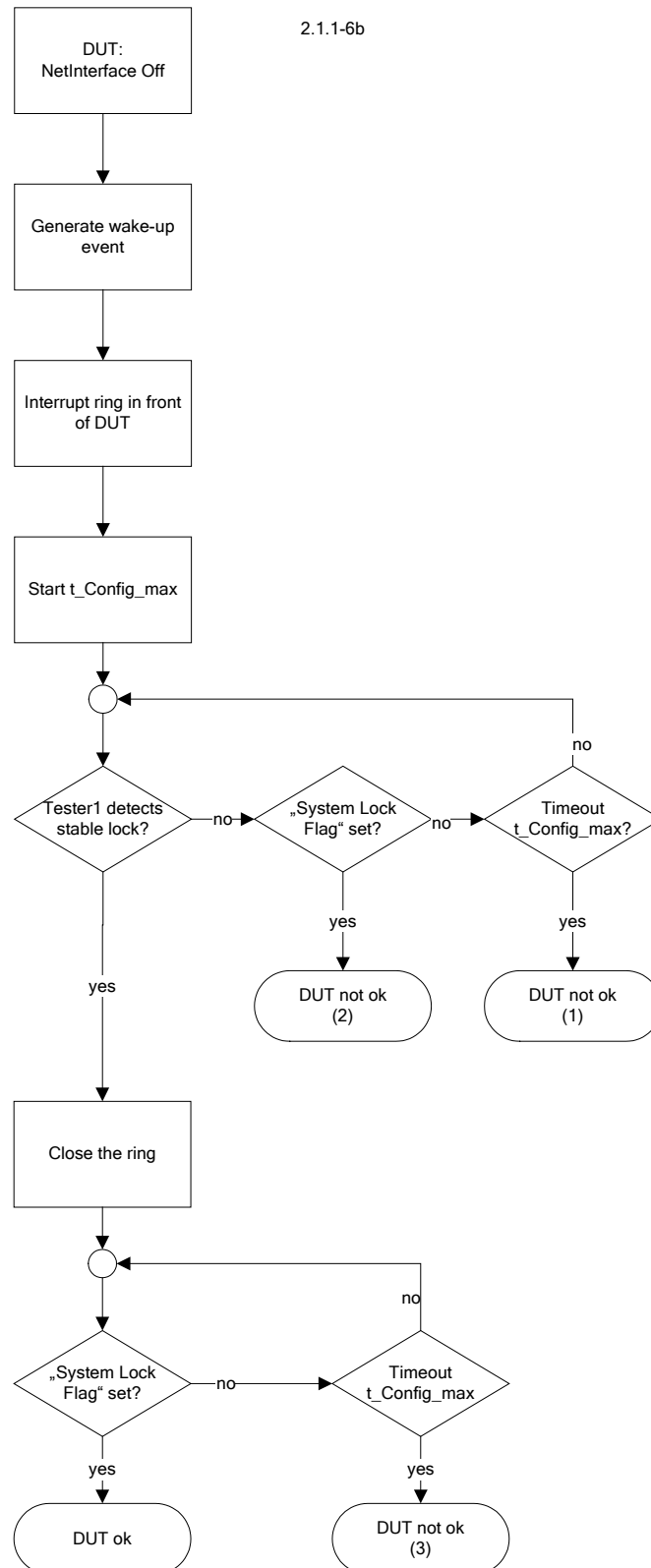


Figure 3-4 Flow of System Lock Flag test (b).

3.4.2.3 Wake-up / Shutdown Test (2.1.1-4)

Name of test	Wake-up / Shutdown test 2.1.1-4
Reference to MOST Specification	3.1.2.2.2 NetInterfaceInit
Value of Interest	t_Config
Start Conditions	DUT in NormalOperation Ring Closed
Test description	The AutoWakeup will be triggered by the tester via FB ET. If the tester detects a stable lock (MOST-signal), the timer t_Config will be started. The DUT has to switch off the MOST signal within t_Config.
Experimental set-up	- Tester 1 in slave mode - Tester 2 in spy mode
Device type	TM, which is capable to wake via MOST signal. [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices containing TM, which are capable to wake via MOST signal
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok (1): No signal from DUT received after wake-up. DUT not ok (2): DUT switches off the signal too early. DUT not ok (3): DUT switches off the signal too late.

2.1.1-4

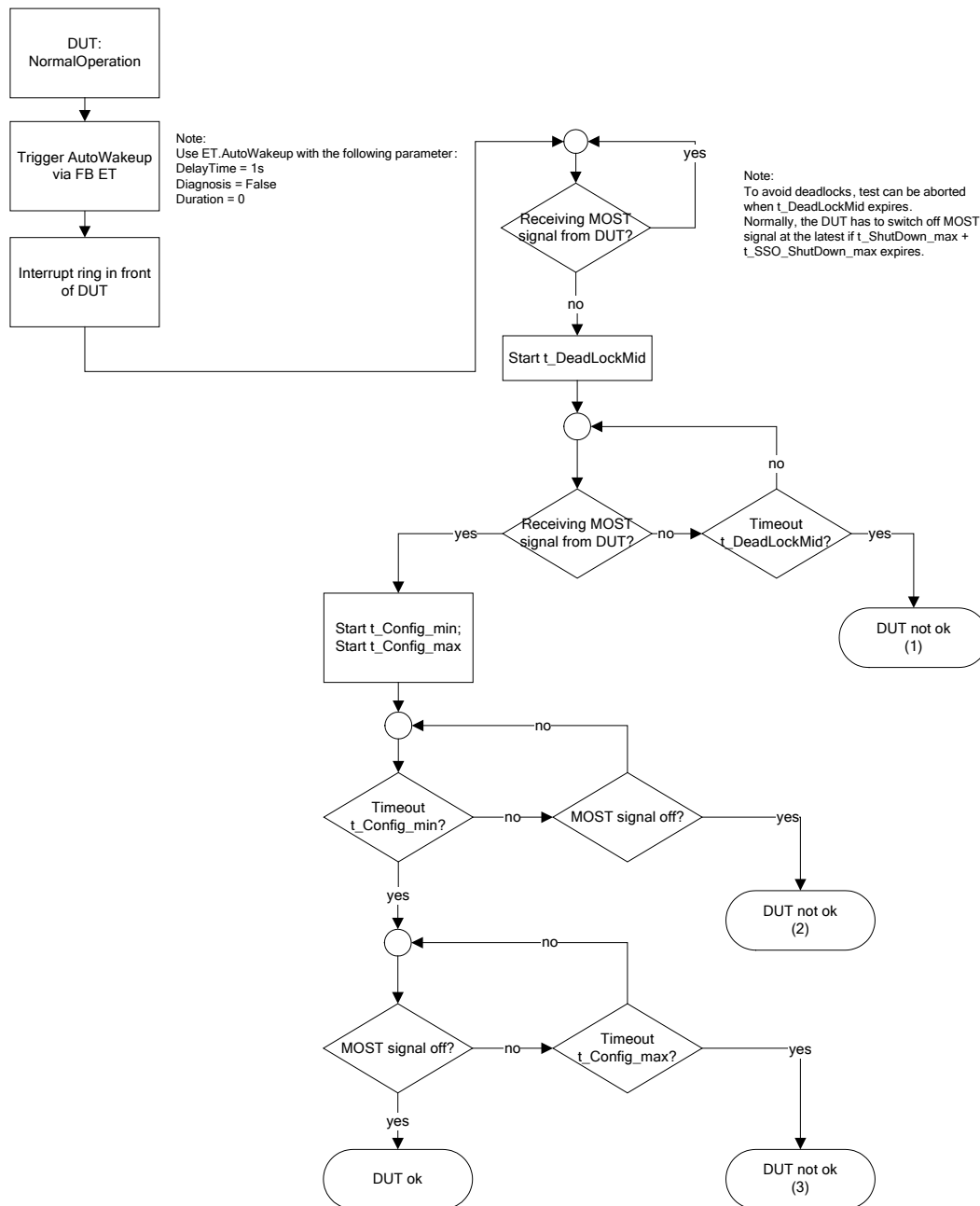


Figure 3-5 Flow of Wake-up / Shutdown test.

3.4.3 Wake-up – Timing Slave

3.4.3.1 Slave Lock Detection Test (2.1.2-5)

Name of test	Slave lock detection test 2.1.2-5
Reference to MOST Specification	3.1.2.2.2 NetInterfaceInit
Value of Interest	t_Config + t_WaitNodes
Start Conditions	DUT: NetInterface Off
Test description	The tester is configured as timing master. It generates the MOST signal without setting the System Lock Flag. The System Lock Flag remains cleared even in case the tester detects a stable lock. The DUT has to switch off the MOST signal as soon as (t_Config_max + t_WaitNodes_max) expires after switching on the MOST signal.
Experimental set-up	- Tester 1 in master mode - Tester 2 in spy mode
Device type	All devices except TM [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices without TM
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok: The DUT fails to switch off the MOST signal right on time.

2.1.2-5

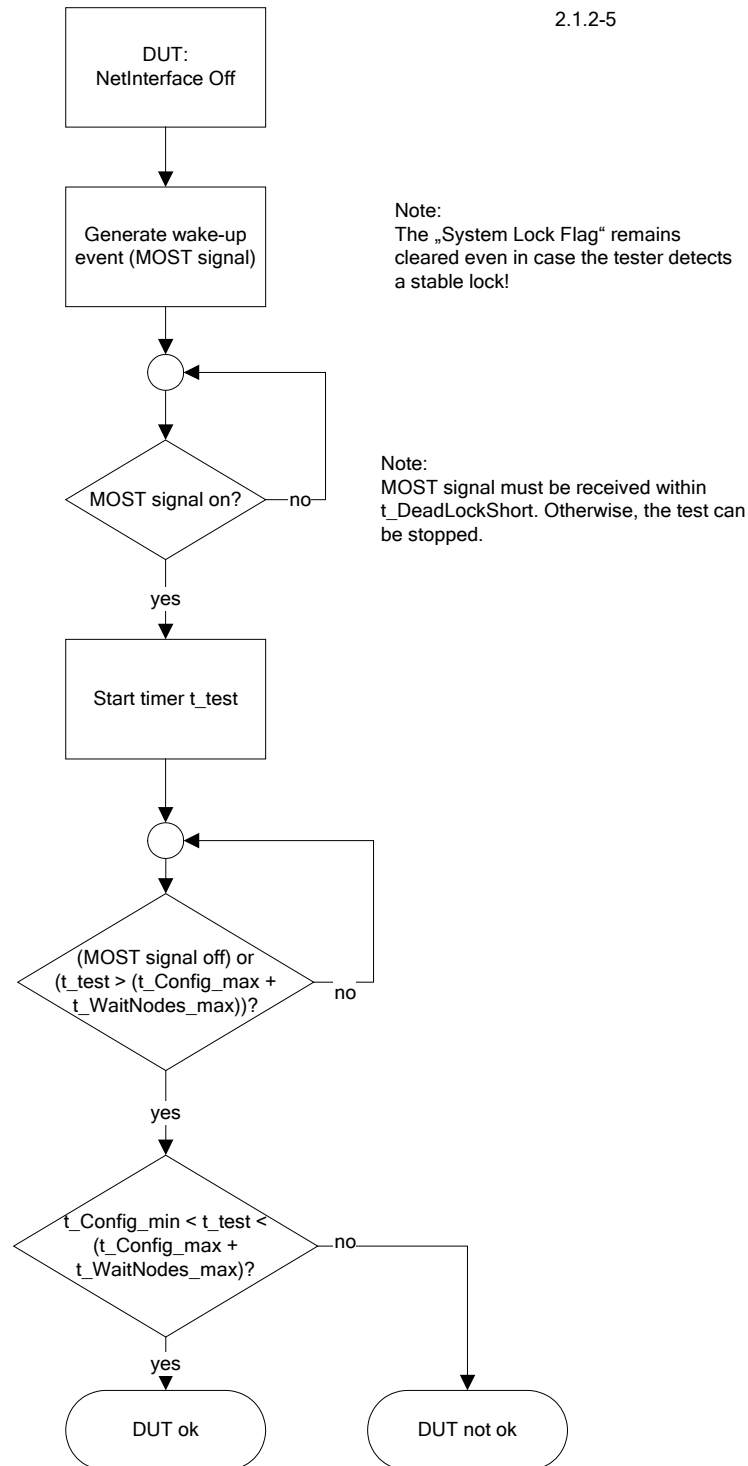


Figure 3-6 Flow of Slave lock detection test.

3.4.3.2 Slave Wake-up (2.1.3-1)

Name of test	Slave wake-up 2.1.3-1
Reference to MOST Specification	3.1.2.2.2 NetInterfaceInit / 3.1.2.3.1 Waking of the Network
Value of Interest	wake-up event t_Restart
Start Conditions	DUT in NormalOperation
Test description	The AutoWakeup will be triggered by the tester via FB ET. The DUT has to generate the MOST signal right on time (depends on manufacturer statement). If the tester receives the MOST signal, it has to generate the MOST signal by itself. The DUT has to stop generating the MOST signal within t_DeadLockShort. Then it has to wait for t_Restart_min before switching on the MOST signal again.
Experimental set-up	- Tester 1 in master mode - Tester 2 in spy mode
Device type	All devices which are capable to wake via MOST signal except TM [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices without TM which are capable to wake via MOST signal
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT fails to generate any MOST signal after wake-up. (Or: external wake-up not supported) DUT not ok (2): The DUT fails to stop generating the MOST signal within t_DeadLockShort. DUT not ok (3): The DUT fails to wait t_Restart before generating the MOST signal again. DUT not ok (4): The DUT fails to generate the MOST signal if t_Restart expires.

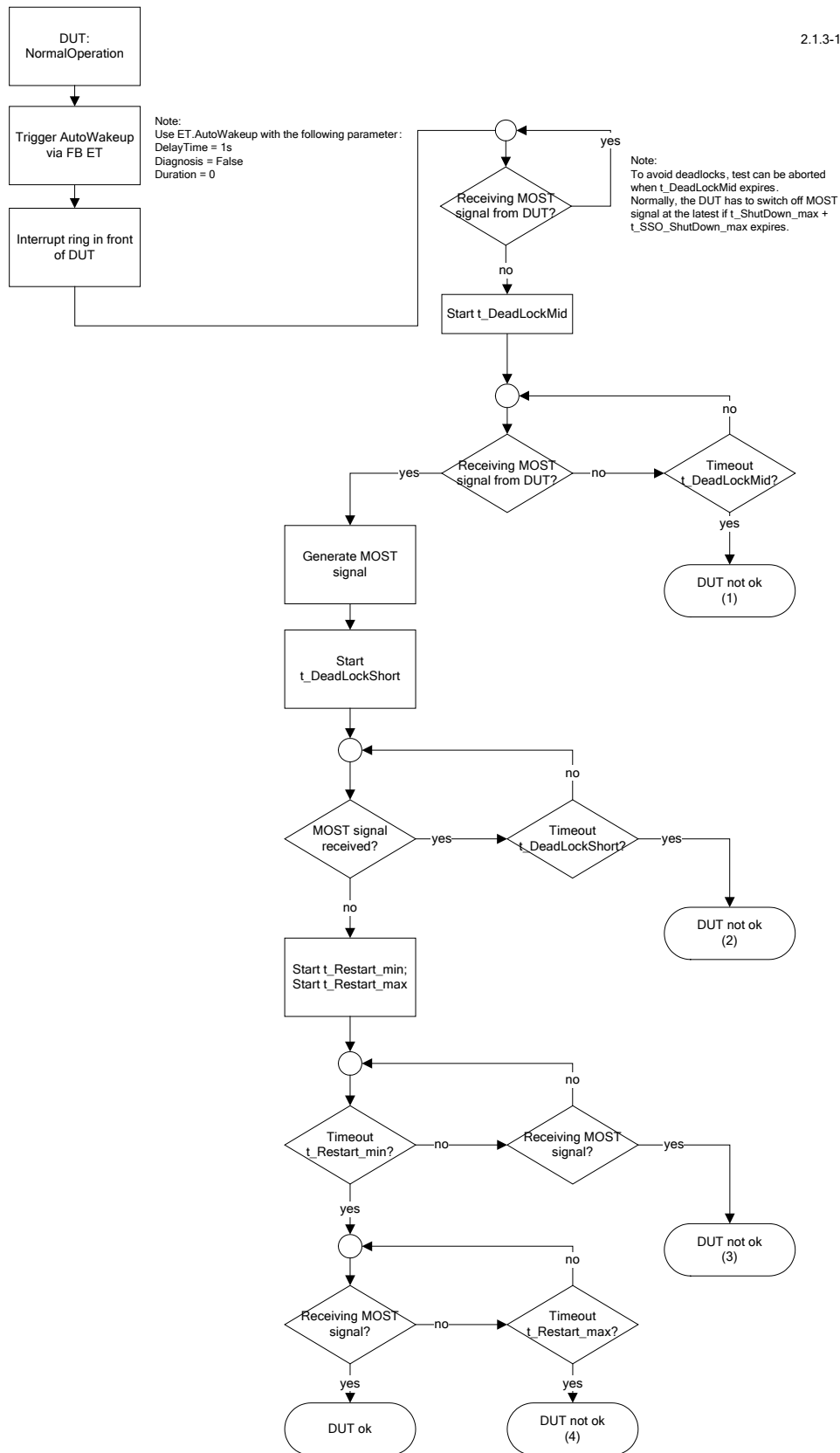


Figure 3-7 Flow of Slave wake-up.

3.4.3.3 Waking Slave Timeout (2.1.3-4)

Name of test	Waking slave timeout 2.1.3-4
Reference to MOST Specification	3.1.2.2.2 NetInterfaceInit
Value of Interest	t_Config
Start Conditions	DUT in NormalOperation Ring closed
Test description	The AutoWakeup will be triggered by the tester via FB ET. Then the ring will be interrupted in front of the DUT. The DUT has to generate the MOST signal right in time (depends on manufacturer statement). If t_Config_max expires, the DUT has to switch off the MOST signal.
Experimental set-up	- Tester 1 in master mode - Tester 2 in spy mode
Device type	All devices which are capable to wake via MOST signal except TM [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices without TM which are capable to wake via MOST signal
Note	This test forces the DUT to perform one wake-up retry.
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT fails to generate any MOST signal after external wake-up. (Or: external wake-up not supported) DUT not ok (2): The DUT fails to generate any MOST signal after external wake-up (during retry). DUT not ok (3): The DUT fails to switch off the MOST signal right in time. DUT not ok (4): The DUT fails to switch off the MOST signal right in time (during retry).

2.1.3-4

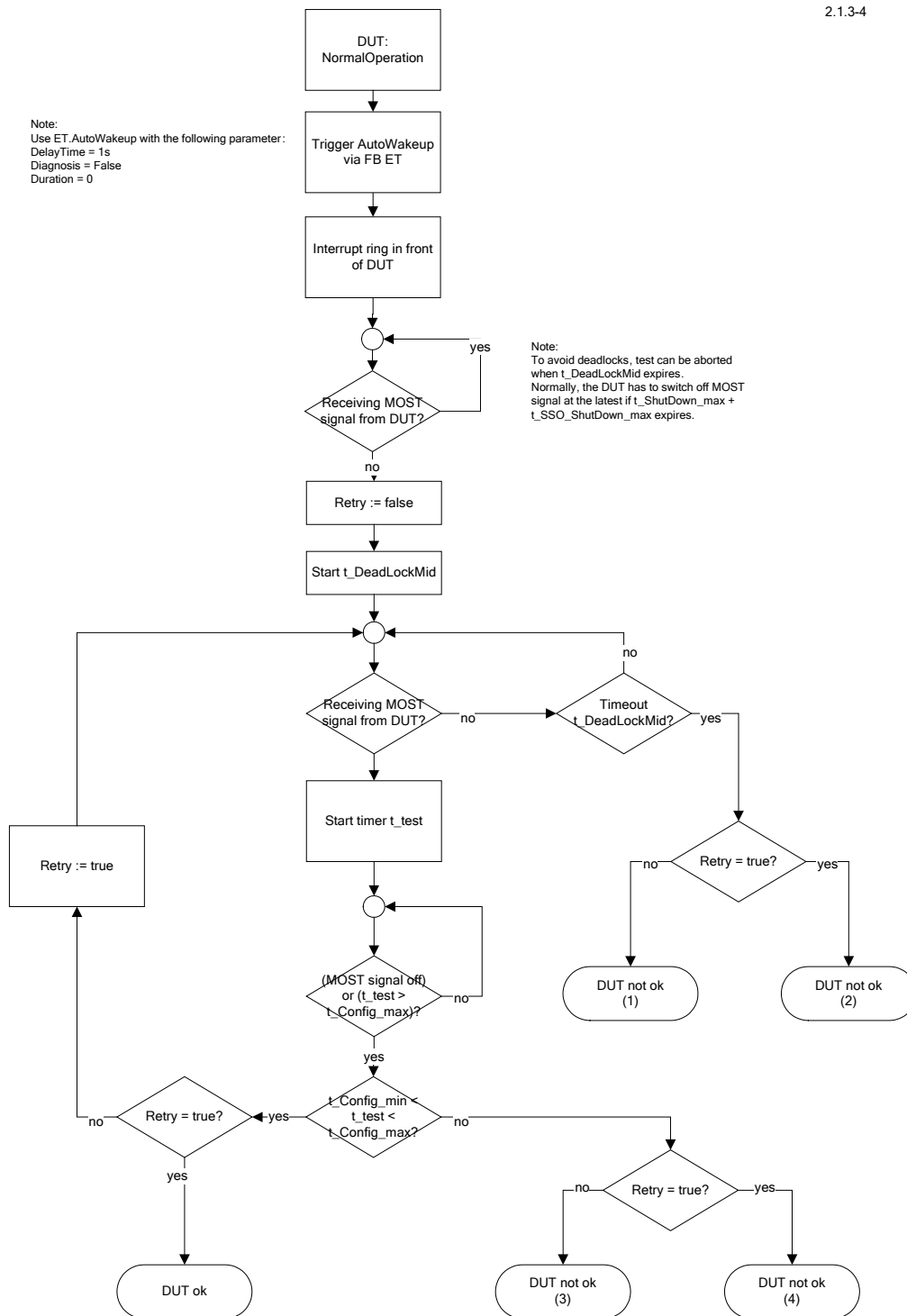


Figure 3-8 Flow of Waking slave timeout.

3.5 Normal Operation

3.5.1 Normal Operation – Unlock

3.5.1.1 Short Unlock Test (2.2.1-3)

Name of test	Short unlock test 2.2.1-3
Reference to MOST Specification	3.1.2.2.3 NetInterfaceNormalOperation
Value of Interest	t_Unlock t_Lock
Start Conditions	DUT in NormalOperation
Test description	The tester generates 10 short unlocks (each 0.2 x t_Unlock_typical) with a pause of t_Lock_max between each unlock. The DUT must not stop generating the MOST signal. At the end of the test, the DUT has to continue with NormalOperation.
Experimental set-up	Depends on DUT: DUT is TM: - Tester 1 in slave mode - Tester 2 in spy mode DUT is not TM: - Tester 1 in slave mode - Tester 2 in master mode
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices
Note	The reaction of the application (mute) is not tested by this test. Procedure for unlock generation described in para 3.1 (General Notes)
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT has switched off the MOST signal. DUT not ok (2): The DUT does not perform NO at the end of the test.

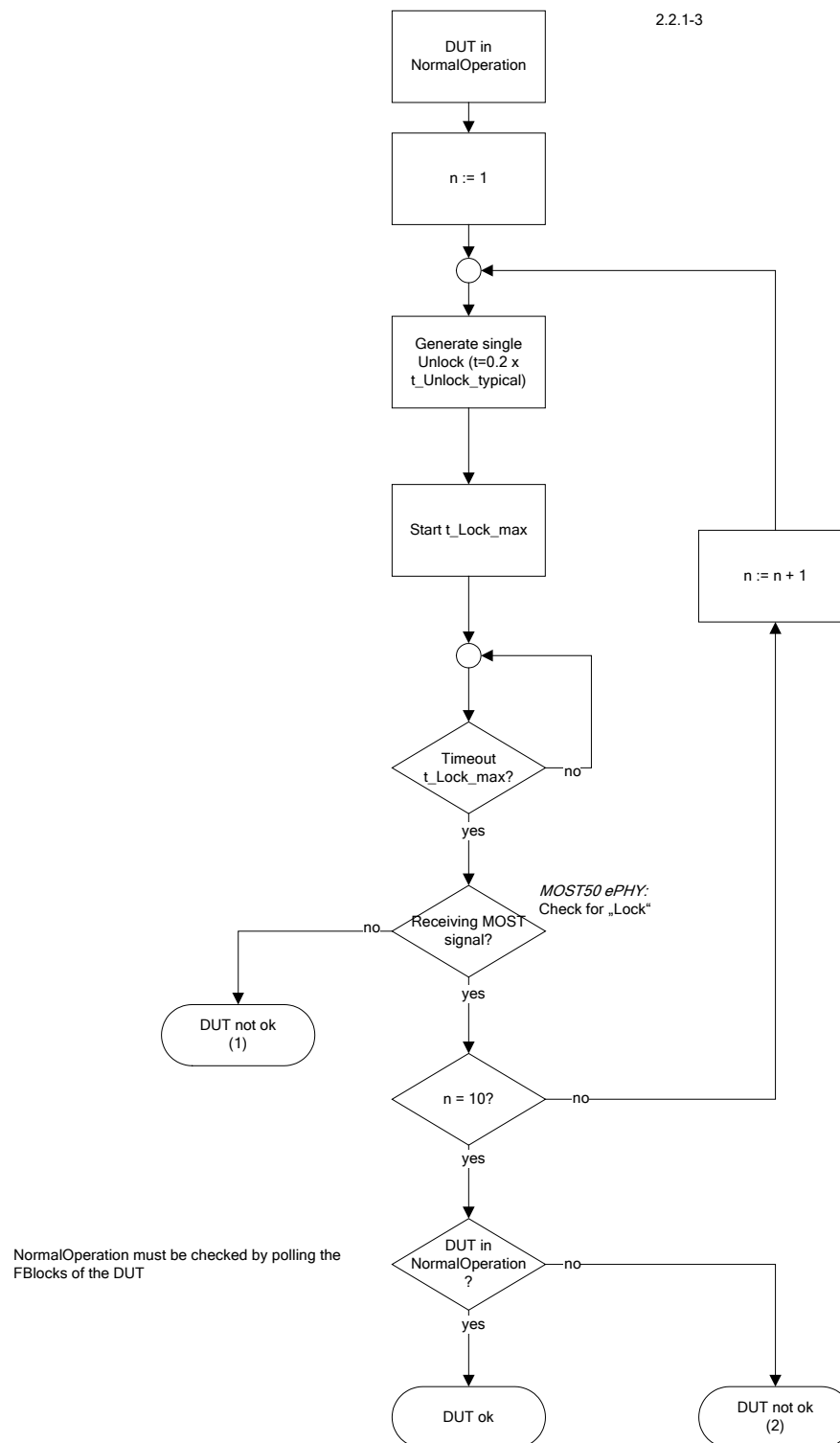


Figure 3-9 Flow of Short unlock test.

3.5.1.2 Accumulated Short Unlock Test (2.2.1-4)

Name of test	Accumulated short unlock test 2.2.1-4
Reference to MOST Specification	3.1.2.2.3 NetInterfaceNormalOperation
Value of Interest	t_Unlock t_Lock Shutdown Flag t_SSO_ShutDown t_Restart
Start Conditions	DUT in NormalOperation
Test description	The tester checks the Shutdown Flag at the beginning of the test. The tester generates an unlock sequence (consists of a series of unlocks ($0.5 \times t_Unlock_min$) and locks ($0.8 \times t_Lock_min$)). The DUT must not detect a critical unlock too early ($< 122ms$) or too late ($> 336ms$). The tester monitors the shutdown flag. It must be set by DUT within $122ms + t_ShutDown_typ$ and $336ms + t_ShutDown_max$. The DUT must not switch off the MOST signal before $t_SSO_ShutDown$ expires. If the MOST signal is switched on again by tester (lock possible; Shutdown Flag not set), the DUT has to wait for $t_Restart_min$ before generating the MOST signal again.
Experimental set-up	- Tester 1 in slave mode - Tester 2 in master mode or slave mode (depends on DUT)
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices
Note	Procedure for unlock generation described in para 3.1 (General Notes).
Result	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT has set the Shutdown Flag erroneously. DUT not ok (2): The DUT switches off MOST signal too early in case of accumulated short unlocks. DUT not ok (3): The DUT fails to set the ShutDown Flag in case of accumulated short unlocks DUT not ok (4): The DUT sets the ShutDown Flag too early in case of accumulated short unlocks. DUT not ok (5): The DUT fails to switch off MOST signal in case of accumulated short unlocks. DUT not ok (6): The DUT switches off MOST signal too early in case of accumulated short unlocks. DUT not ok (7): The DUT fails to wait for $t_Restart$ before switching on the MOST signal again (OR: <i>electrical Wake-up/MOST signal wake-up not supported</i>).

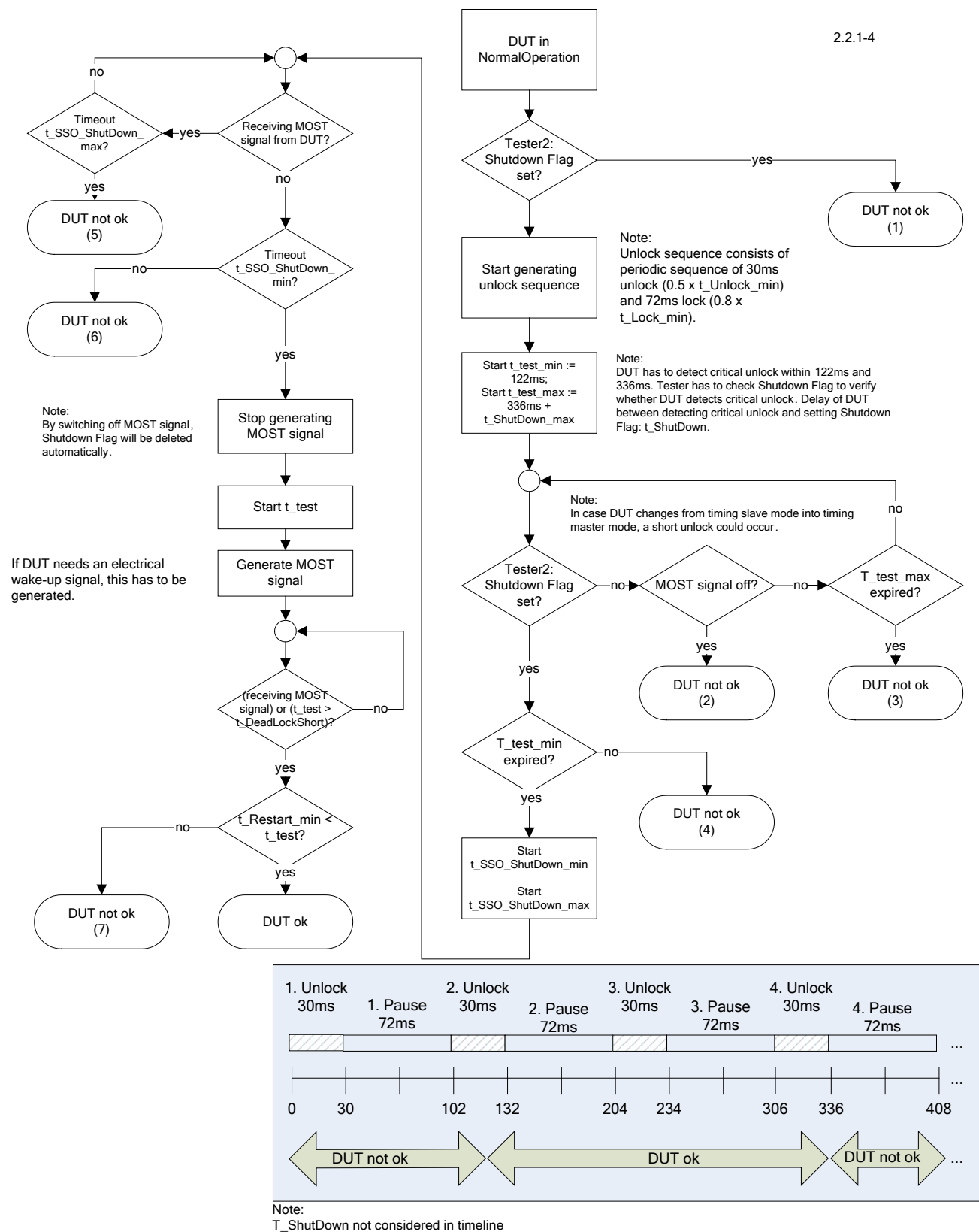


Figure 3-10 Flow of Accumulated short unlock test.

3.5.1.3 Critical Unlock Test (2.2.1-5)

Name of test	Critical unlock test 2.2.1-5
Reference to MOST Specification	3.1.2.2.3 NetInterfaceNormalOperation
Value of Interest	t_Unlock t_Restart Shutdown Flag t_SSO_ShutDown
Start Conditions	DUT in NormalOperation
Test description	The tester checks the Shutdown Flag at the beginning of the test and starts to generate an unlock. The DUT has to detect critical unlock and has to set the ShutDown Flag. The DUT must not switch off the MOST signal before t_SSO_ShutDown expires. If the MOST signal is switched on again by tester (lock possible; Shutdown Flag not set), the DUT has to wait for t_Restart_min before generating the MOST signal again.
Experimental set-up	- Tester 1 in slave mode - Tester 2 in master mode or slave mode (depends on DUT)
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices
Note	Procedure for unlock generation described in para 3.1 (General Notes). If DUT does not support wake-up neither via MOST signal nor via electrical wake-up, the second part of the test (t_Restart) is not applicable
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT has set the Shutdown Flag erroneously. DUT not ok (2): The DUT fails to set ShutDown Flag right in time. DUT not ok (3): The DUT fails to switch off MOST signal in case of critical unlock. DUT not ok (4): The DUT switches off MOST signal too early. DUT not ok (5): The DUT fails to wait for t_Restart before switching on the MOST signal again (OR: <i>electrical Wake-up/MOST signal wake-up not supported</i>)

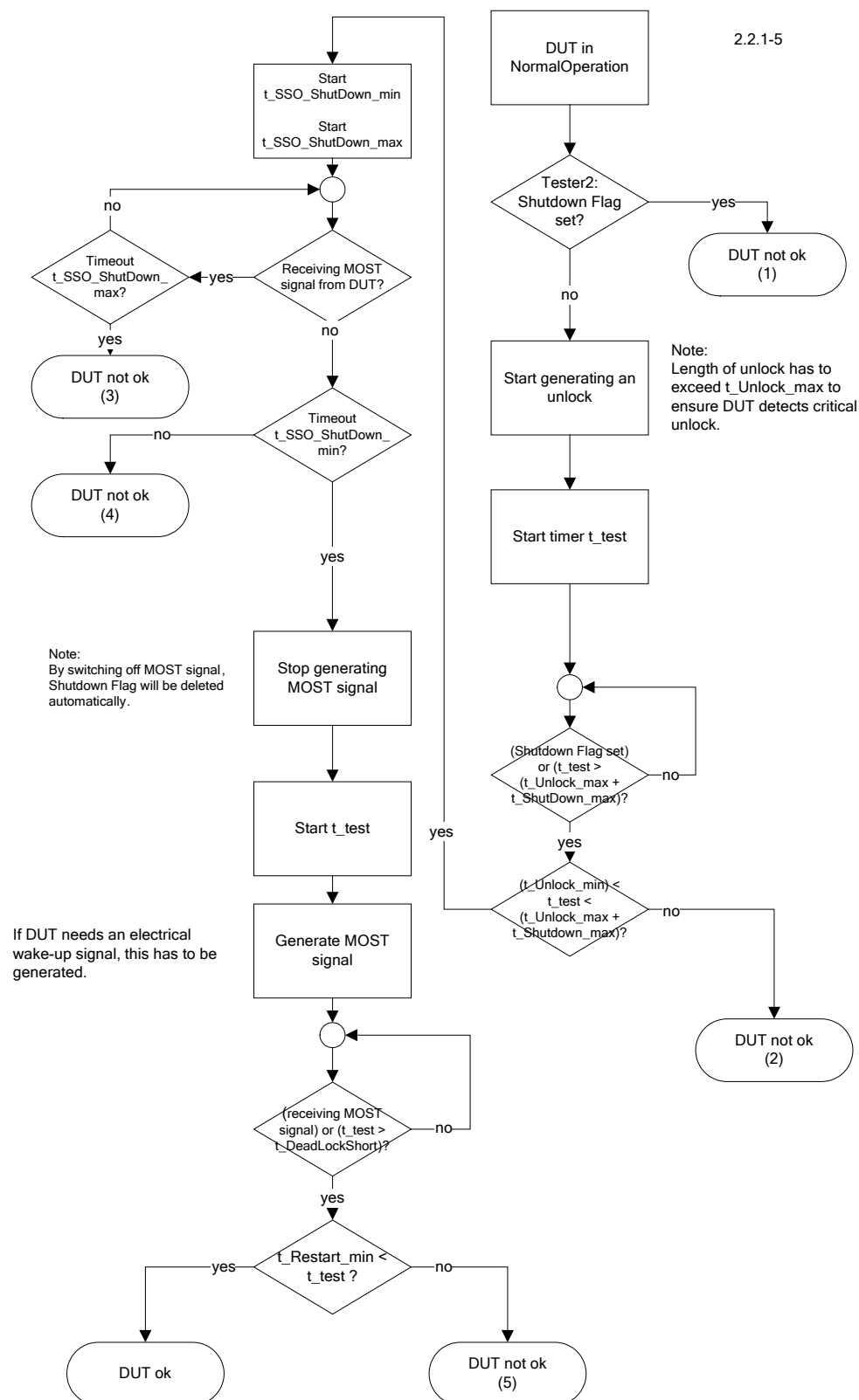


Figure 3-11 Flow of Critical unlock test.

3.5.1.4 Sudden Signal Off Test (2.2.1-6)

Name of test	Sudden Signal Off test 2.2.1-6
Reference to MOST Specification	3.1.4.2 Detection of Sudden Signal Off and Critical Unlock
Value of Interest	t_SSO_ShutDown Shutdown Flag NetBlock.ShutDownReason
Start Conditions	DUT in NormalOperation
Test description	The ring will be interrupted in front of the DUT. The DUT has to set the Shutdown Flag and continue generating MOST signal in master mode until t_SSO_ShutDown expires. After restart, the DUT has to indicate correct ShutDownReason.
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in slave mode
Device type	All devices, but in Multi Node Devices only the first node. [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices
Note	Test has to be performed twice: 1. Generate Sudden Signal Off (interrupt ring) 2. Critical unlock
Results	DUT ok: The DUT has passed the test DUT not ok (1): Shutdown Flag already set in NormalOperation DUT not ok (2): DUT fails to switch off MOST signal DUT not ok (3): DUT fails to set Shutdown Flag DUT not ok (4): DUT switches off MOST signal too early DUT not ok (5): DUT indicates wrong ShutDownReason after restart DUT not ok (6): DUT does not set ShutDown Flag correctly

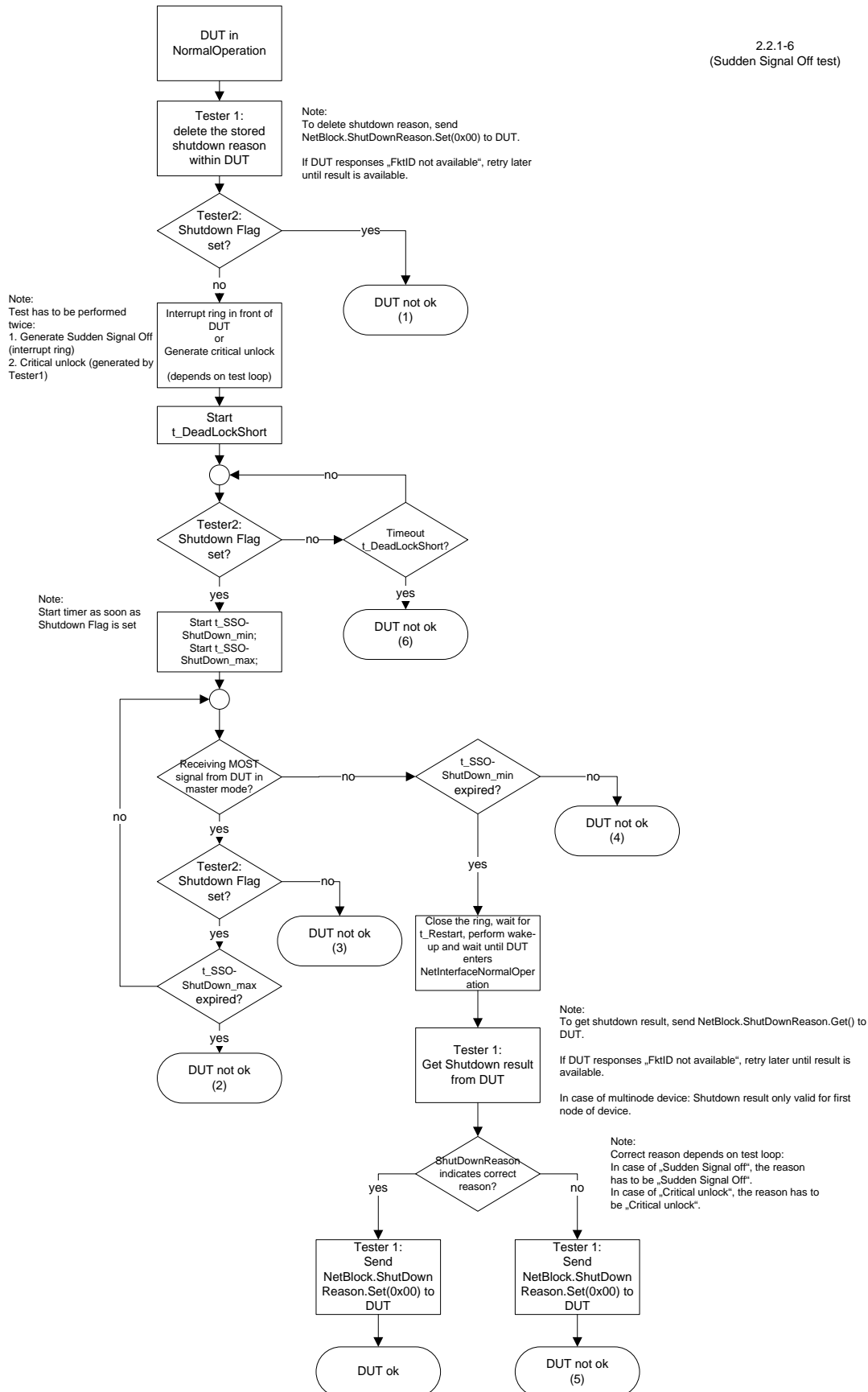


Figure 3-12 Flow of Sudden Signal off test.

3.5.1.5 Shutdown Flag Present Test (2.2.1-7)

Name of test	Shutdown Flag present test 2.2.1-7
Reference to MOST Specification	3.1.4.2 Detection of Sudden Signal Off and Critical Unlock
Value of Interest	NetBlock.ShutDownReason
Start Conditions	DUT in NormalOperation
Test description	By interrupting the ring between Tester 2 and Tester 1, Tester 1 will set the Shutdown Flag automatically. The DUT has to switch off MOST signal within t_DeadLockShort. After restart, the DUT has to indicate correct ShutDownReason.
Experimental set-up	- Tester 1 in master mode - Tester 2 in spy mode
Device type	All devices except TM [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices without TM
Note	
Results	DUT ok: The DUT has passed the test DUT not ok (1): DUT fails to switch off MOST signal DUT not ok (2): DUT indicates wrong ShutDownReason after restart

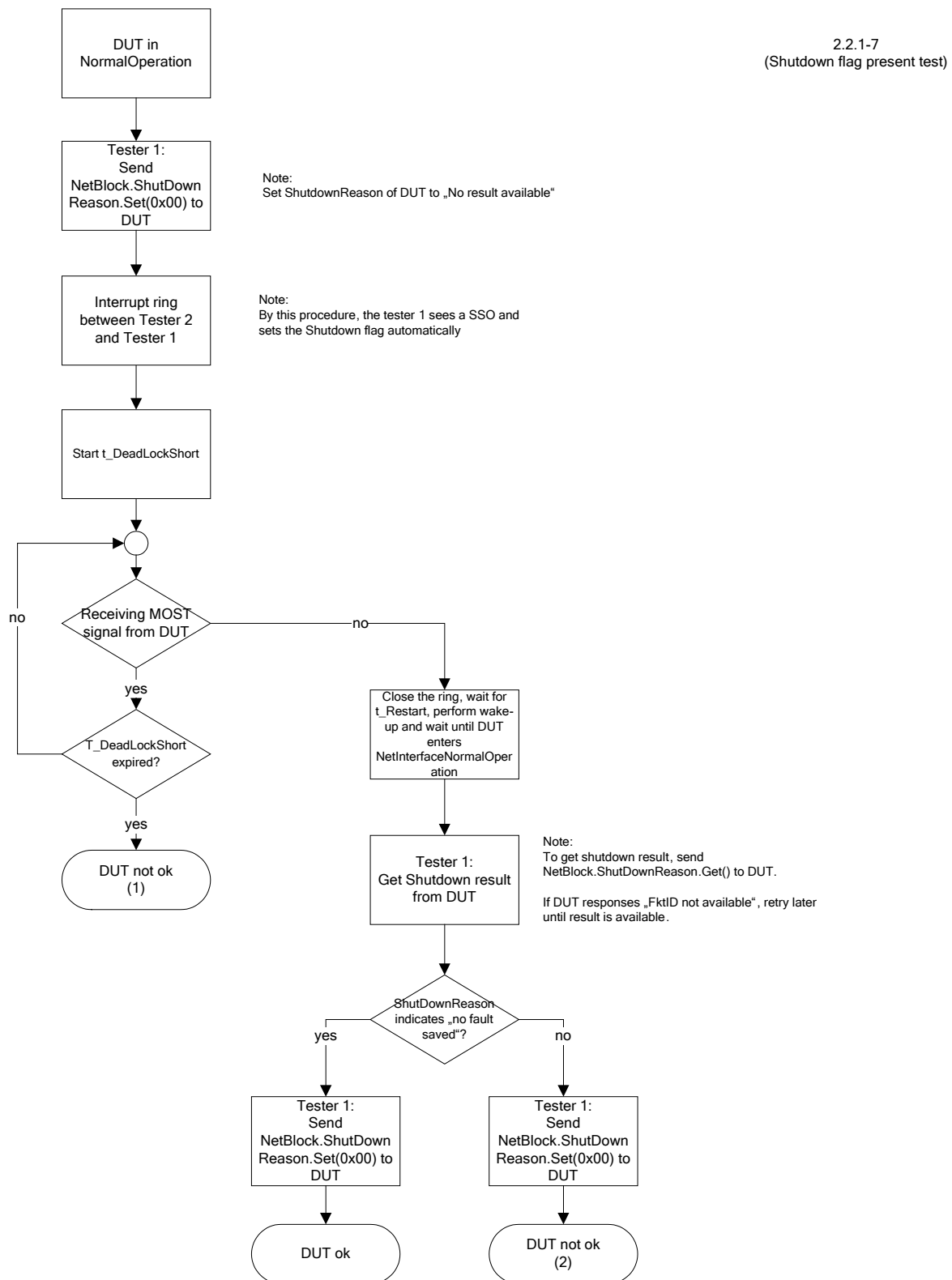


Figure 3-13 Flow of Shutdown flag present test.

3.6 Power Management

3.6.1 Power Management – Power Master

3.6.1.1 Timeout Execute / Timeout Suspend (2.3.1-3)

Name of test	Timeout execute / Timeout suspend 2.3.1-3
Reference to MOST Specification	3.1.2.3.2 Network Shutdown
Value of Interest	ShutDown.Start(Query) t_WaitSuspend ShutDown.Start(Execute) t_ShutDownWait + t_SSO_Shutdown
Start Conditions	DUT in NormalOperation
Test description	The DUT is triggered to perform a ShutDown (via FBlock ET). It has to perform a shutdown by sending ShutDown.Start(Query). Then it has to wait for t_WaitSuspend_min (test 2.3.1-6 could be performed) before sending a ShutDown.Start(Execute). The DUT has to wait for t_ShutDownWait_min + t_SSO_Shutdown_min (ignoring ShutDown.Result(Suspend)) before switching off the MOST signal.
Experimental set-up	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
Device type	PM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing PM. The power slaves must be configured to not send ShutDown.Result(Suspend).
Note	Pre-conditions have to be switched off that could prevent DUT from performing ShutDown
Result	DUT ok (1): The DUT has passed the test. DUT not ok (1): The DUT fails to start the shutdown procedure. DUT not ok (2): DUT sends ShutDown.Start(Execute) too early. DUT not ok (3): DUT switches off the MOST signal too early. DUT not ok (4): DUT switches off the MOST signal too late or fails to set Shutdown Flag. DUT not ok (5): DUT fails to perform NormalOperation during t_WaitSuspend

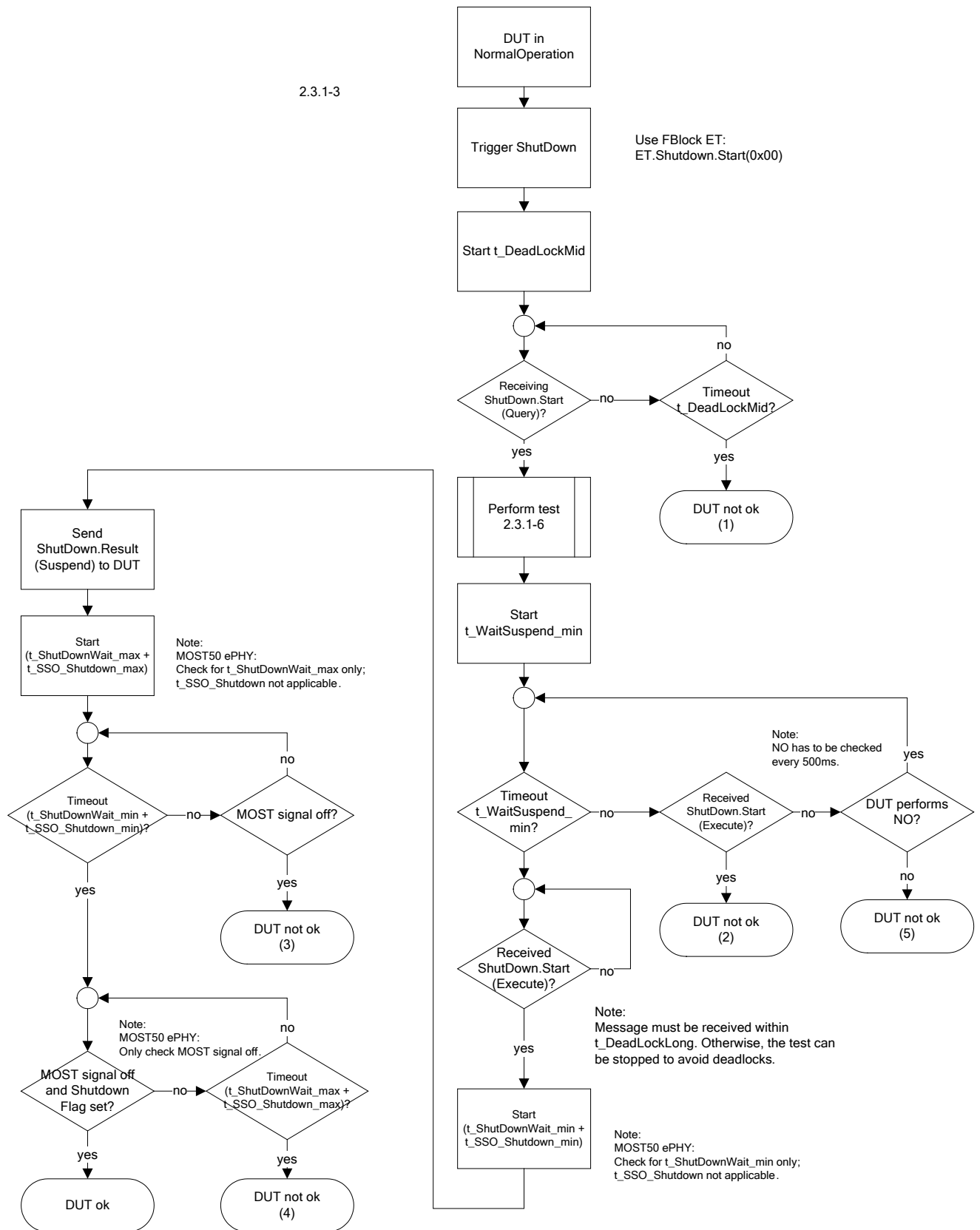


Figure 3-14 Flow of Timeout execute / Timeout suspend.

3.6.1.2 Timeout Suspend (2.3.1-6)

Name of test	Timeout suspend 2.3.1-6
Reference to MOST Specification	3.1.2.3.2 Network Shutdown
Value of Interest	ShutDown.Start(Suspend) t_RetryShutDown
Start Conditions	DUT in NormalOperation
Test description	The test is continued from 2.3.1-3. The tester sends a ShutDown.Result(Suspend) to the DUT before t_WaitSuspend expires. The DUT has to stay in NormalOperation until t_RetryShutDown_min expires. Then it has to start a new ShutDown attempt before t_RetryShutDown_max expires.
Experimental set-up	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
Device type	PM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing PM. The power slaves must be configured to not send ShutDown.Result(Suspend).
Note	
Results	DUT ok (1): The DUT has passed the test. DUT ok (2): The DUT has passed the test (it overrides suspend requests from slave) DUT not ok (1): The DUT fails to perform NO during t_RetryShutDown. DUT not ok (2): DUT fails to send ShutDown.Start(Query). DUT not ok (3): DUT sends ShutDown.Start(Query) too early.

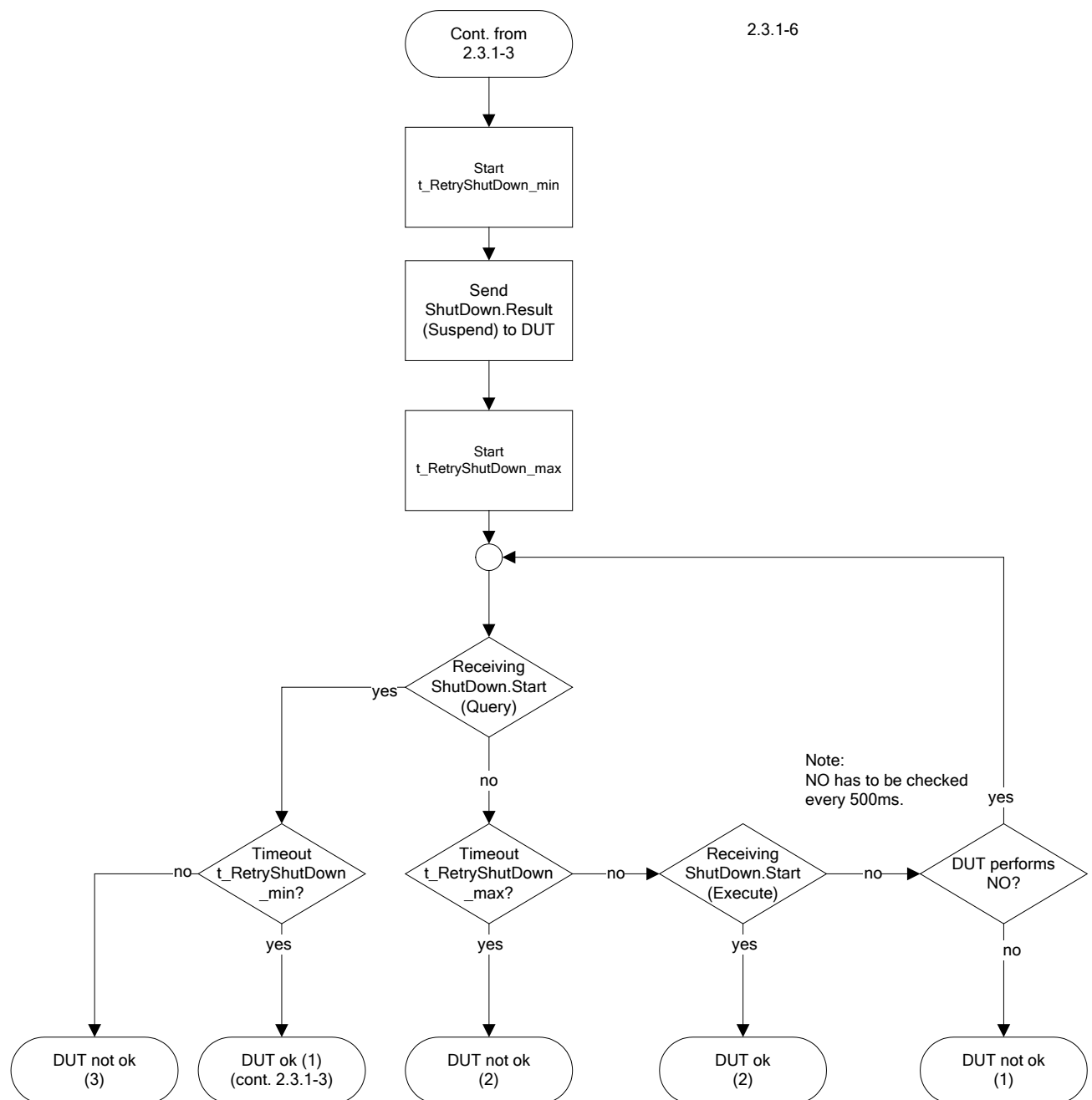


Figure 3-15 Flow of Timeout suspend.

3.6.1.3 Temperature ShutDown Reaction Test (2.3.3-5)

Name of test	Temperature ShutDown Reaction Test 2.3.3-5
Reference to MOST Specification	3.1.5.6.1 Levels of Temperature Alert MOST Dynamic Specification: 5.7 Network restart after over-temperature shutdown
Value of Interest	t_WaitAfterOvertempShutdown
Start Conditions	DUT in NormalOperation
Test description	Tester 1 sets the DUT into "overtemperature mode" by sending ShutDown.Result(0x03) and switches off the MOST signal. The DUT has to wait for t_WaitAfterOvertempShutdown resp. t_Restart before performing restart.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	PM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing PM
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok: DUT fail to enter overtemperature mode (does not keep t_WaitAfterOvertempShutdown resp. t_Restart) before restart

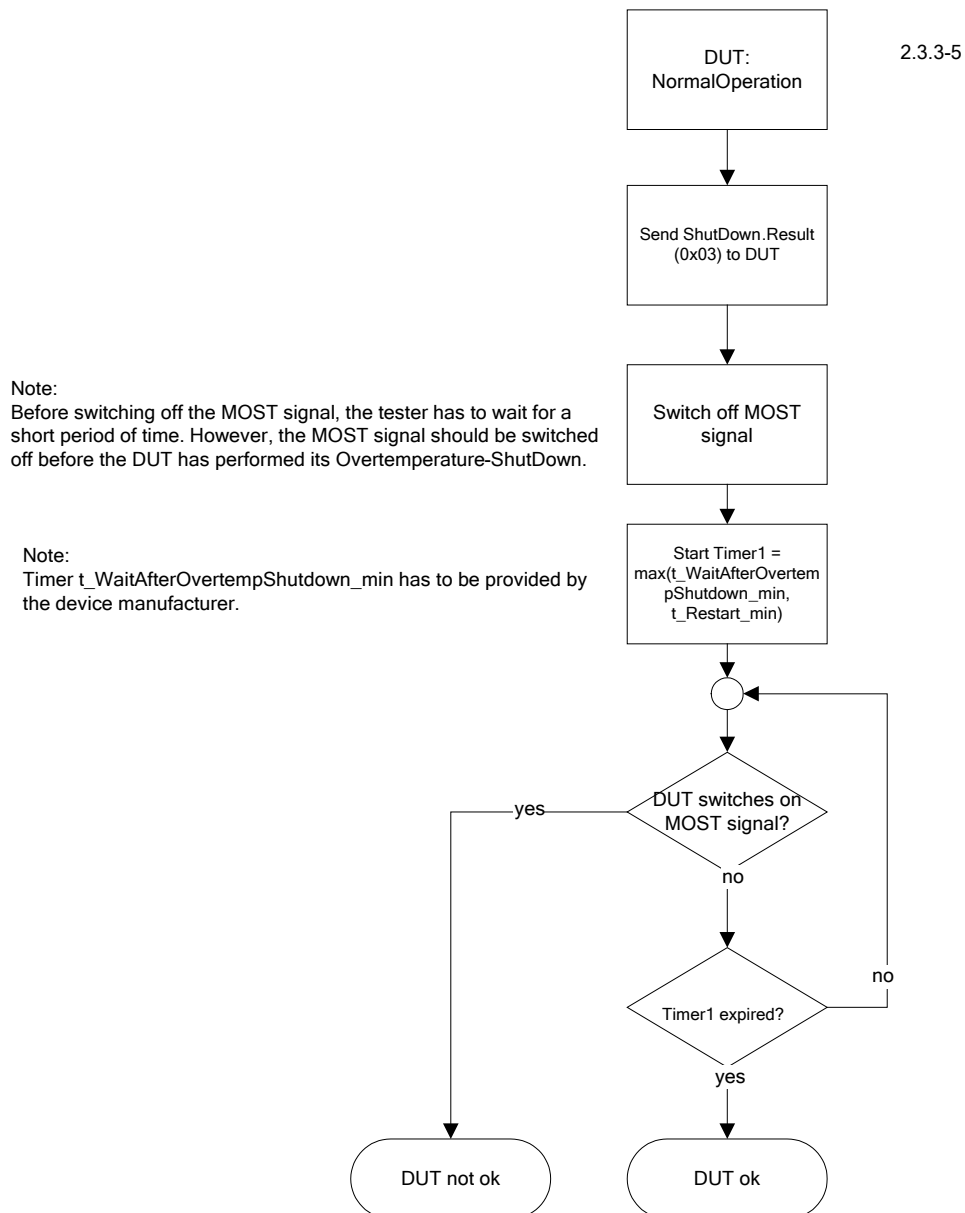


Figure 3-16 Flow of Temperature ShutDown Reaction test.

3.6.2 Power Management – Power Slave

3.6.2.1 ShutDown.Start(Query) Test (2.3.2-2)

Name of test	ShutDown.Start(Query) test 2.3.2-2
Reference to MOST Specification	3.1.2.3.2 Network Shutdown
Value of Interest	Message ShutDown.Result(Suspend) t_WaitSuspend
Start Conditions	DUT in NormalOperation.
Test description	The tester sends ShutDown.Start(Query). The DUT has to continue performing NormalOperation. It is allowed to sent ShutDown.Result(Suspend) once. In that case, ShutDown.Result(Suspend) has to be sent within t_WaitSuspend_min after ShutDown.Start(Query).
Experimental set-up	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices except PM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices without PM
Note	If test ends with "DUT ok" , continue with test case 2.3.2-3 (ShutDown.Start(Execute) test).
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT fails to continue NO after ShutDown.Start(Query). DUT not ok (2): The DUT sends ShutDown.Result(Suspend) too often. DUT not ok (3): The DUT sends ShutDown.Result(Suspend) too late. DUT not ok (4): The DUT switches off MOST signal by itself.

2.3.2-2

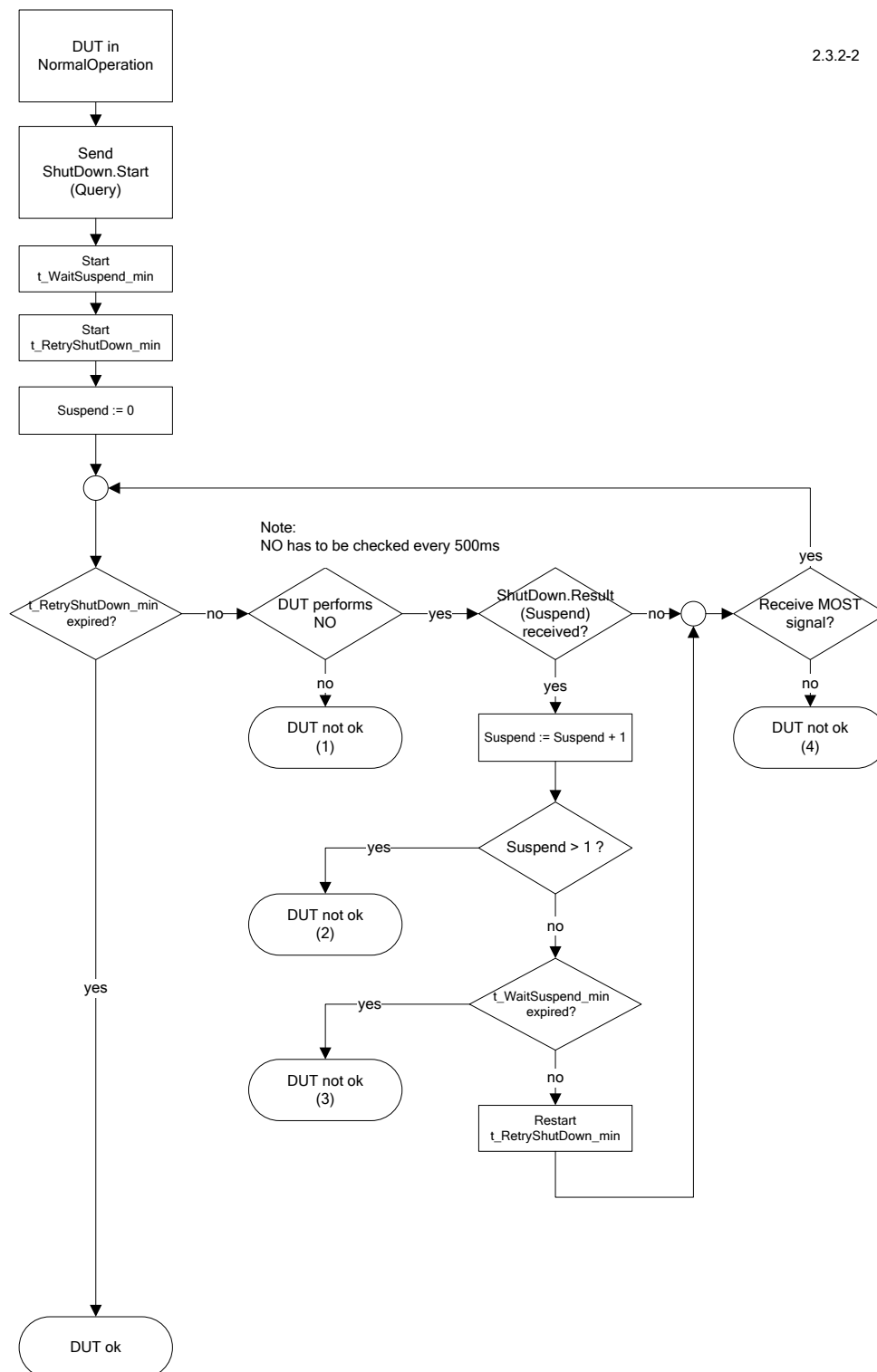


Figure 3-17 Flow of ShutDown.Start(Query) test.

3.6.2.2 ShutDown.Start(Execute) Test (2.3.2-3)

Name of test	ShutDown.Start(Execute) test 2.3.2-3
Reference to MOST Specification	3.1.2.3.2 Network Shutdown
Value of Interest	t_PwrSwitchOffDelay
Start Conditions	The test continues directly from "ShutDown.Start(Query) test" (2.3.2-2).
Test description	After sending ShutDown.Start(Query) and waiting for t_WaitSuspend_min, the tester sends a ShutDown.Start(Execute). The DUT has to prepare for shutdown. It has to continue performing NormalOperation. All kinds of communication have to be possible until the tester switches off the MOST signal. The DUT must not switch off the signal by itself. If the tester switches off the MOST signal, the DUT must not enter SleepMode before t_PwrSwitchOffDelay_min is expired.
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode - Power consumption of DUT is monitored
Device type	All devices except PM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices without PM
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT fails to perform NormalOperation after reception of ShutDown.Start(Execute). DUT not ok (2): DUT switches off MOST signal by itself too early. DUT not ok (3): DUT fails to switch off MOST signal within t_DeadLockShort. DUT not ok (4): DUT enters SleepMode too early.

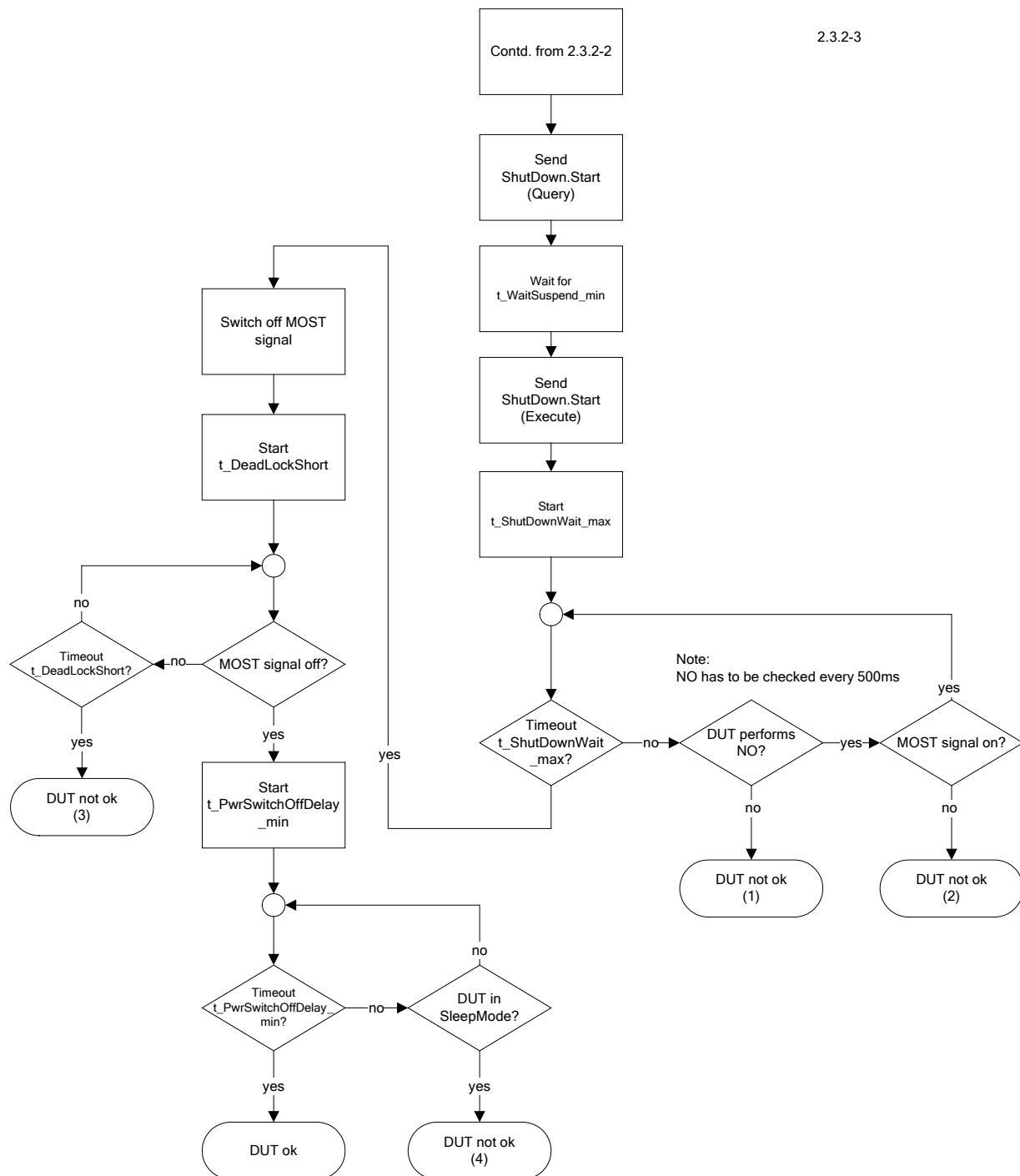
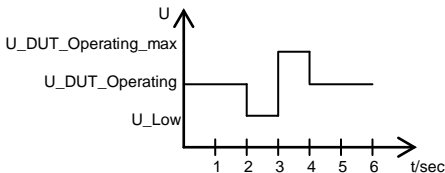


Figure 3-18 Flow of ShutDown.Start(Execute) test.

3.6.3 Power Management – General

3.6.3.1 Over-/ Undervoltage Test (2.3.3-1)

Name of test	Over-/ undervoltage test 2.3.3-1
Reference to MOST Specification	3.1.5.5 Undervoltage Management
Value of Interest	Undervoltage Management
Start Conditions	DUT: NetInterface Off
Test description	The voltage level of the power supply is adjusted to normal operation level (U_DUT_Operating). The voltage level is reduced to U_Low, then it is increased to U_DUT_Operating_max. Every voltage level is hold for 1 second. The DUT must not start generating any kind of MOST signal.
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT); MOST signal off - Tester 2 in spy mode
Device type	All devices except PM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices without PM
Note	The Trigger AutoWakeUp must not be applied Test has to be performed twice: a) Ramp function of the power supply for testing: Increasing and decreasing of voltage with ramps of 0,1V/0,5s. b) Square egde function of the power supply for testing:  The voltage levels U_Low, U_DUT_Operating and U_DUT_Operating_max are provided by the manufacturer (see DUT manufacturer information list).
Results	DUT ok: The DUT has passed the test. DUT not ok (1): Change to U_Low results in MOST signal generation. DUT not ok (2): Change to U_DUT_Operating_max results in MOST signal generation. DUT not ok (3): Change to U_DUT_Operating results in MOST signal generation.

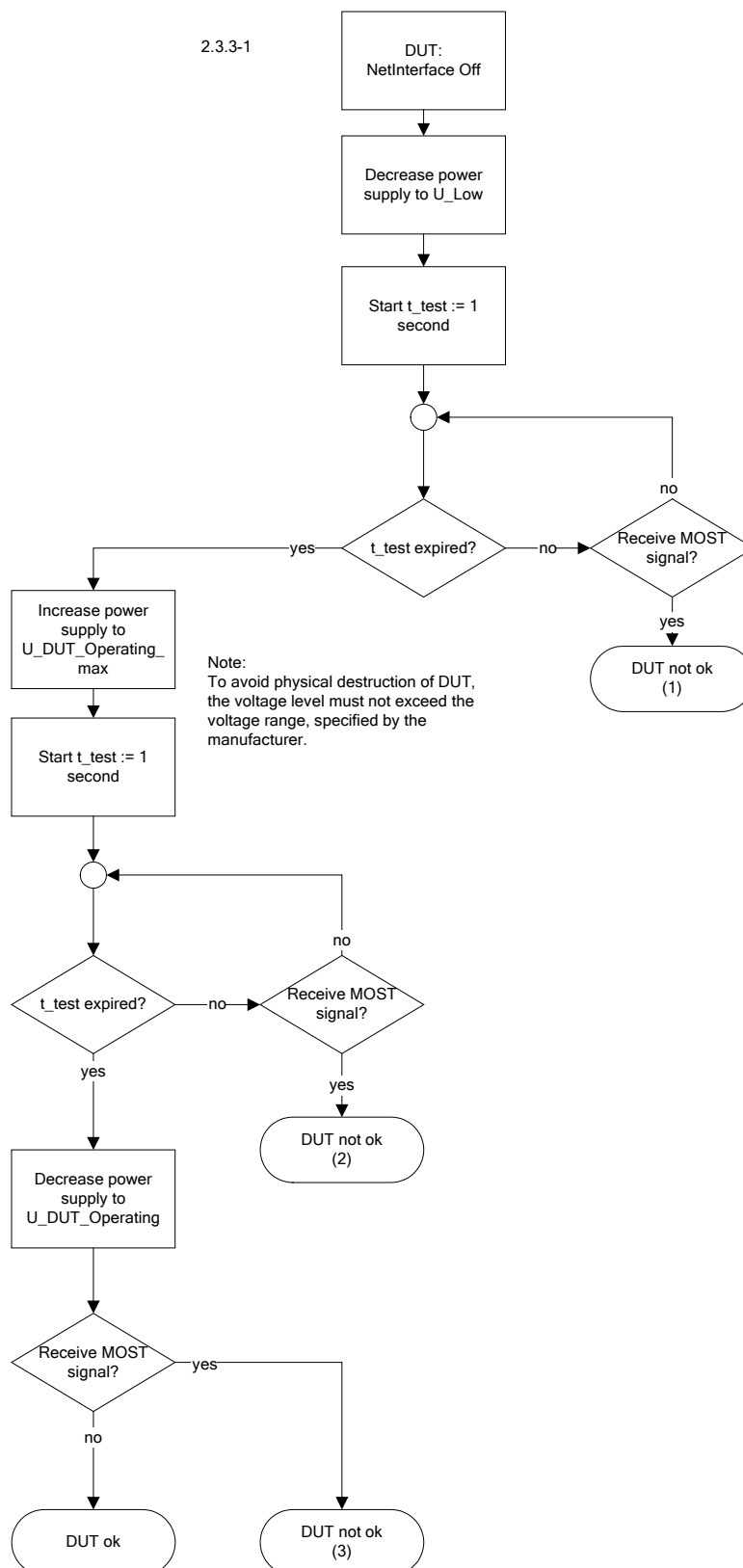


Figure 3-19 Flow of Over-/ undervoltage test.

3.6.3.2 Over-temperature Signal Off Test (2.3.3-4)

Name of test	Over-temperature signal off test 2.3.3-4
Reference to MOST Specification	3.1.5.6 Over-Temperature Management
Value of Interest	NetBlock.ShutdownResult(0x03)
Start Conditions	DUT in NormalOperation
Test description	The DUT performs NormalOperation. The temperature is increased to $\vartheta_{shutdown}$ (simulated by FBlock ET only). The DUT has to send a temperature warning message; communication must not be disturbed. The temperature is increased again to $\vartheta_{critical}$ (simulated by FBlock ET only). The DUT has to switch off MOST signal.
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in slave mode
Device type	All devices except PM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices without PM
Note	This test covers overtemperature behavior only. All other critical events are covered by other tests, already (Undervoltage / Overvoltage: 2.4.2-1; Unlock: 2.2.1-5)
Results	DUT ok (1): The DUT has passed the test. DUT ok (2): DUT does not support temperature monitoring / management DUT not ok (1): DUT fails to send temperature warning message. DUT not ok (2): DUT disturbs communication. DUT not ok (3): DUT fails to switch off in case of over-temperature.

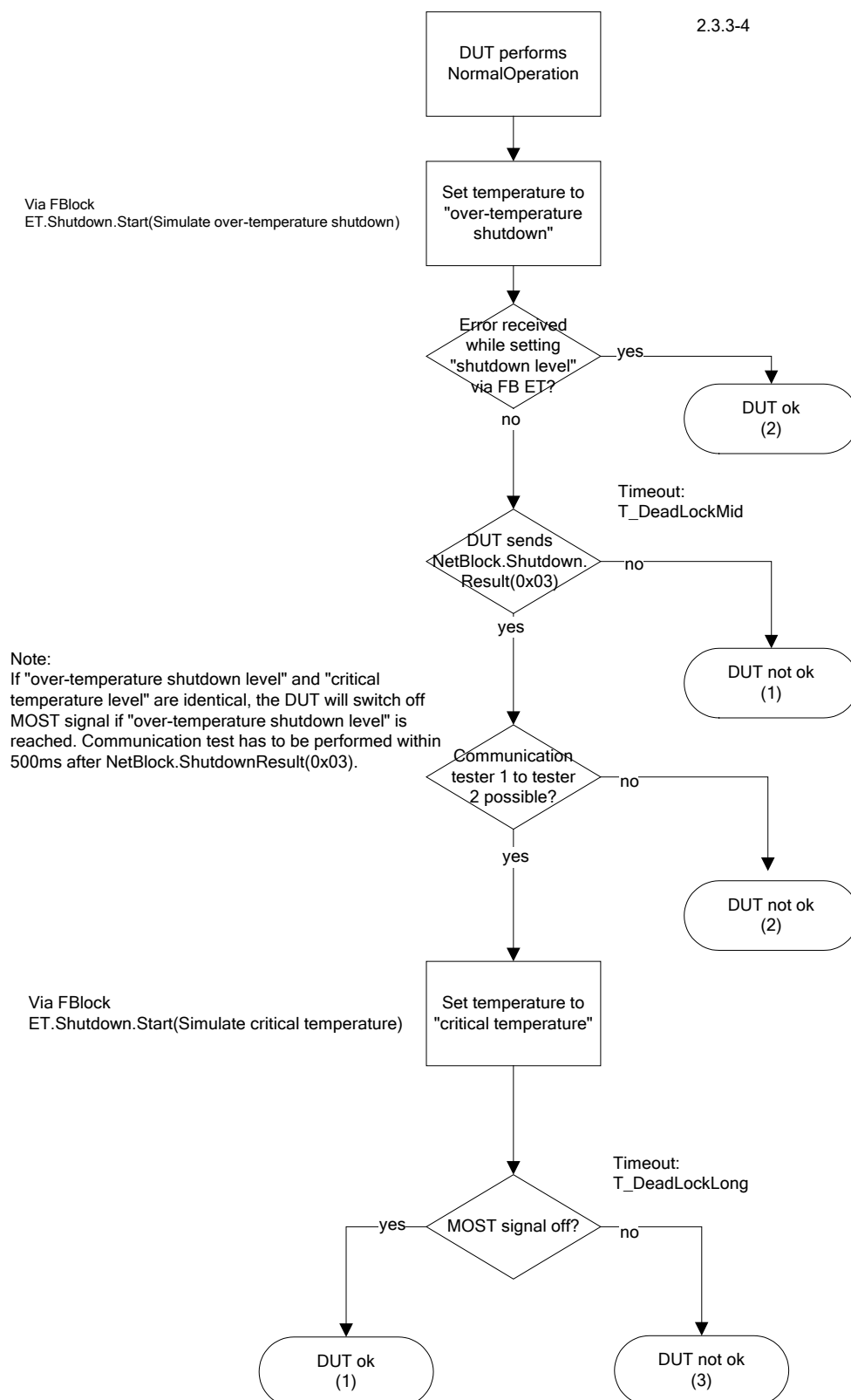


Figure 3-20 Flow of Over-temperature signal off test.

3.7 Error Management

3.7.1 Error Management – MOST Signal Path

3.7.1.1 Restart Stop Test (2.4.1-1)

Name of test	Restart stop test 2.4.1-1
Reference to MOST Specification	3.1.5.1.2 Waking
Value of Interest	Wake-up retries
Start Conditions	DUT in NormalOperation
Test description	The AutoWakeup will be triggered by the tester via FB ET (wakeup condition will be simulated permanently during the whole test by setting parameter "Duration = 0"). Then the ring will be interrupted in front of the DUT. The DUT has to perform up to three wake-up retries. Then is has to stop the wake-up attempts.
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices which are capable to wake via MOST signal, except PM. [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices without PM which are capable to wake via MOST signal.
Note	Test not applicable to a device that includes the PM.
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT performs too much / too little wake-up retries. DUT not ok (2): The DUT performs more than three wake-up retries.

2.4.1-1

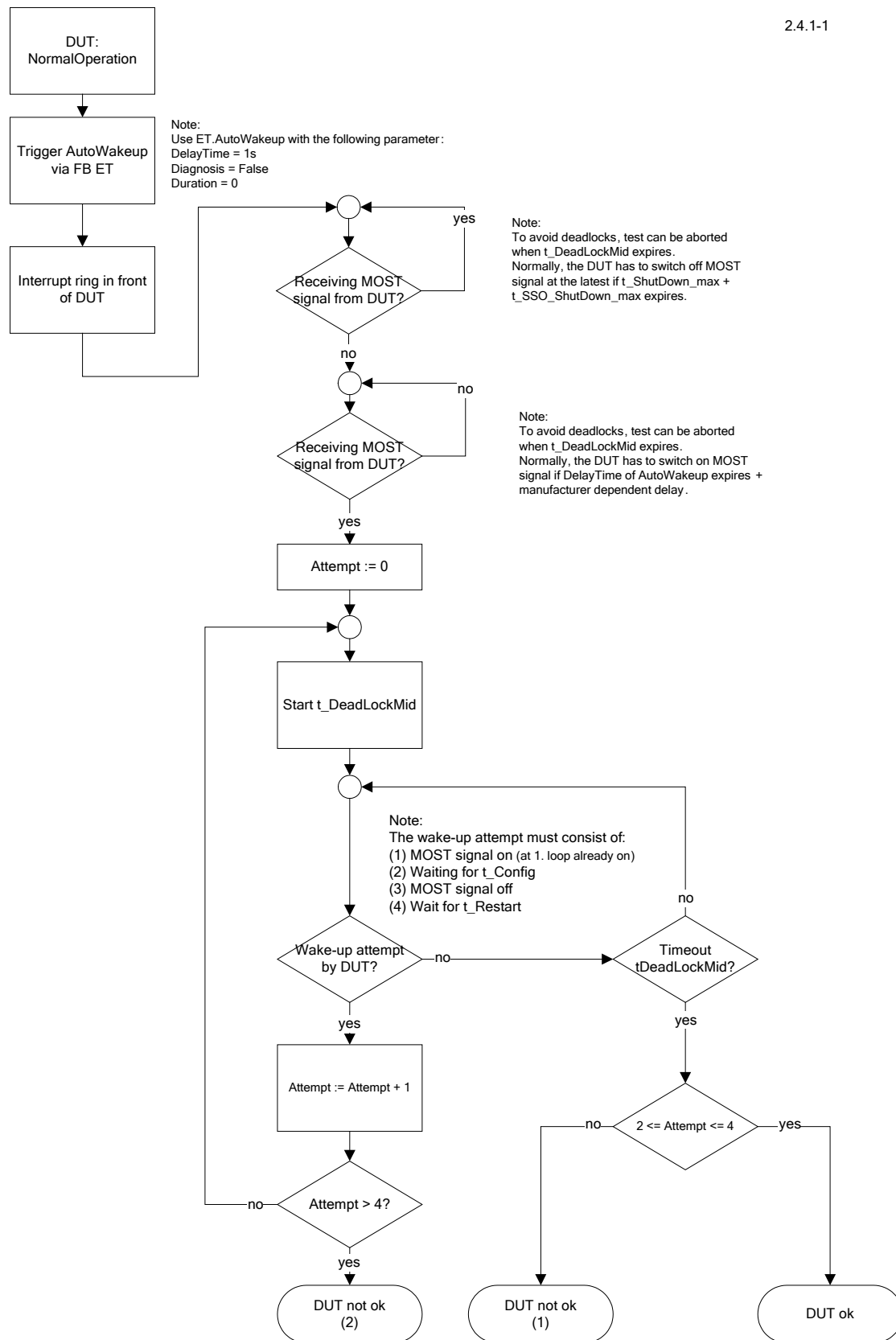


Figure 3-21 Flow of Restart stop test.

3.7.1.2 Restart Continue Test (2.4.1-2)

Name of test	Restart continue test 2.4.1-2
Reference to MOST Specification	3.1.5.1.2 Waking
Value of Interest	Wake-up retries
Start Conditions	DUT in NormalOperation
Test description	The AutoWakeup will be triggered by the tester via FB ET (using parameter Duration = 15 to simulate extended local wakeup event). Then the ring will be interrupted in front of the DUT. The DUT has to perform at least three wake-up retries. If the DUT generates more than three wake-up retries, it has to stop as soon as the wake-up event is stopped.
Experimental set-up	- Tester 1 in slave mode or in master mode (depends on DUT) - Tester 2 in spy mode
Device type	PM which is capable to wake via MOST signal [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices containing PM which are capable to wake via MOST signal
Note	
Results	DUT ok (1): The DUT has passed the test (stops after three retries). DUT ok (2): The DUT has passed the test (perform wake-ups as long as the wake-up event is active). DUT not ok (1): The DUT fails to perform at least one wake-up retry. DUT not ok (2): The DUT continues wake-up attempts without wake-up event.

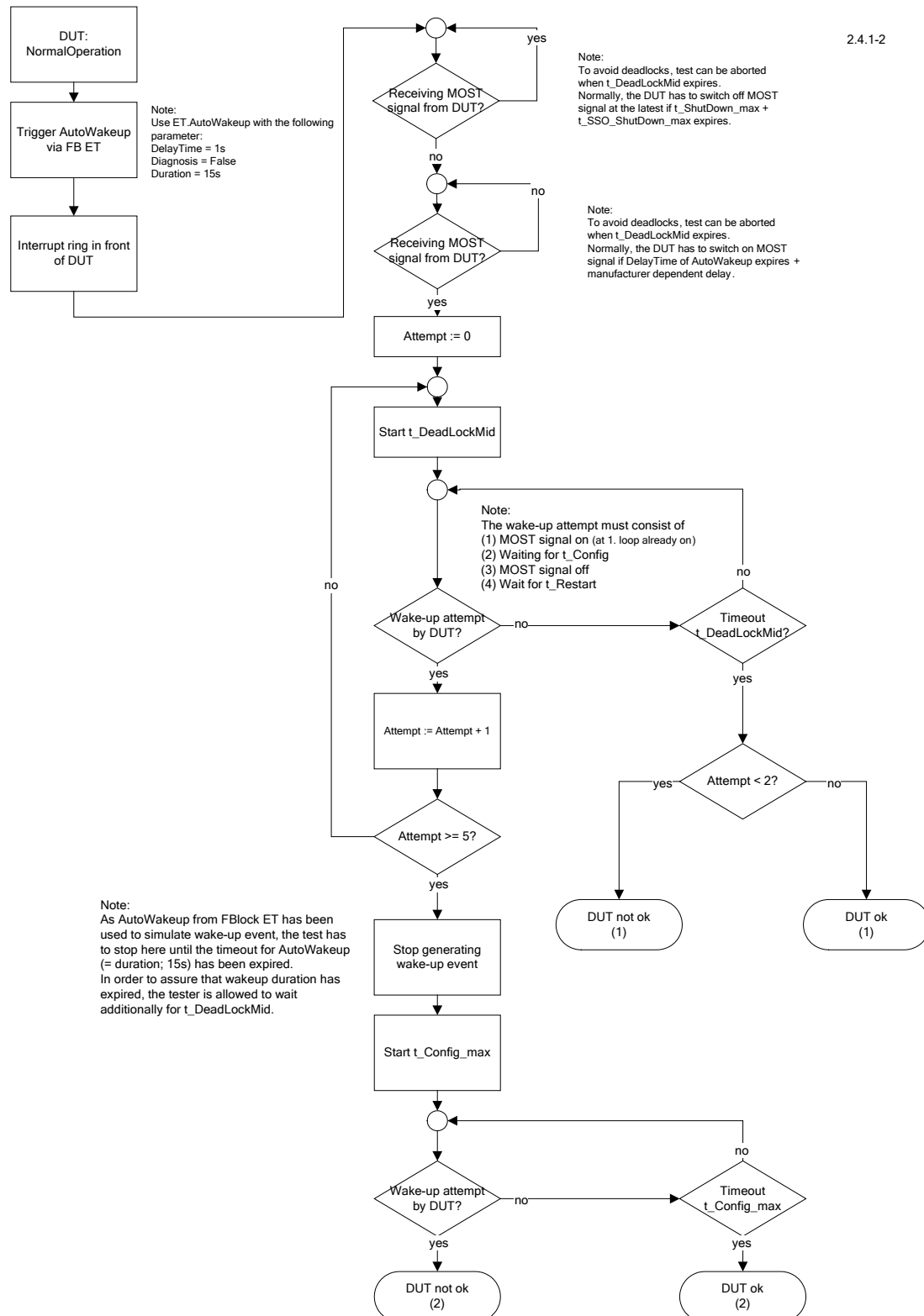


Figure 3-22 Flow of Restart continue test.

3.7.1.3 Reaction Of NCE Test (2.4.1-9)

Name of test	Reaction of NCE test 2.4.1-9
Reference to MOST Specification	3.1.5.3 Network Change Event
Value of Interest	NCE t_WaitAfterNCE ConfigStatus(Invalid)
Start Conditions	DUT in NormalOperation
Test description	Tester 2 opens its bypass. The DUT has to check the system configuration between t_WaitAfterNCE_min and t_DeadLockMid. It has to send ConfigStatus(NewExt). Tester 2 closes its bypass. The DUT has to check the system configuration between t_WaitAfterNCE_min and t_DeadLockMid. It has to send ConfigStatus(Invalid). Tester 2 generates NCE without FBlock changes (tester 2 opens all-bypass; waits for t_WaitAfterNCE_min/2 and closes the bypass again). The DUT has to check the system configuration between t_WaitAfterNCE_min and t_DeadLockMid. It has to send ConfigStatus(NewExt) with empty FBlock list.
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in slave mode; bypass closed
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM.
Note	The test has to be performed once: a) The NCE is generated with unlock. Tester1 has to respond to every system configuration check of the DUT normally but with other FBlocks than tester2.
Results	DUT ok (a/b): The DUT has passed the test. DUT not ok (1a/1b): The DUT fails to check the system configuration. DUT not ok (2a/2b): The DUT fails to send ConfigStatus(NewExt). DUT not ok (3a/3b): The DUT fails to check the system configuration. DUT not ok (4a/4b): The DUT fails to send ConfigStatus(Invalid). DUT not ok (5a/5b): The DUT checks the system configuration too early. DUT not ok (6a/6b): The DUT checks the system configuration too early. DUT not ok (7a/7b): The DUT fails to check the system configuration. DUT not ok (8a/8b): The DUT checks the system configuration too early. (fails to restart t_WaitAfterNCE if new NCE occurs after timer has been started) DUT not ok (9a/9b): The DUT fails to send ConfigStatus(NewExt) with empty FBlock list.

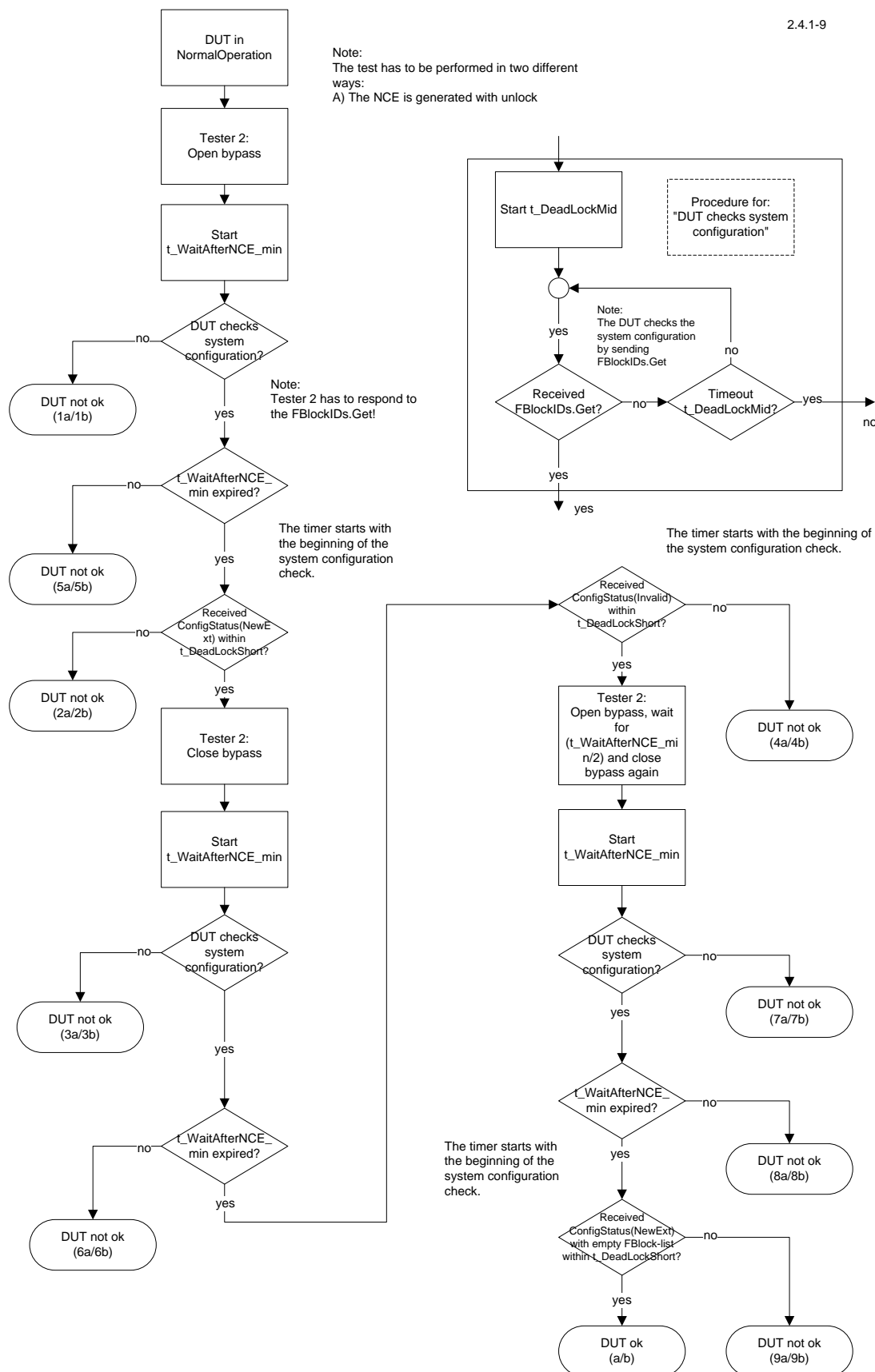


Figure 3-23 Flow of Reaction of NCE test.

3.7.1.4 Re-enter Delay Test (2.4.1-11)

Name of test	Re-enter delay test 2.4.1-11
Reference to MOST Specification	3.2.9 Timing Definitions
Value of Interest	t_Bypass
Start Conditions	DUT in NormalOperation
Test description	The DUT is triggered to leave the ring for a short time (via FBlock ET; Reset). It must not enter the ring again, before t_Bypass_min expires.
Experimental set-up	- Tester 1 in master mode - Tester 2 in spy mode
Device type	All devices except TM [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices without TM
Note	By means of monitoring the MPR register, the tester could detect whether DUT opens / closes bypass
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT enters the ring too early.

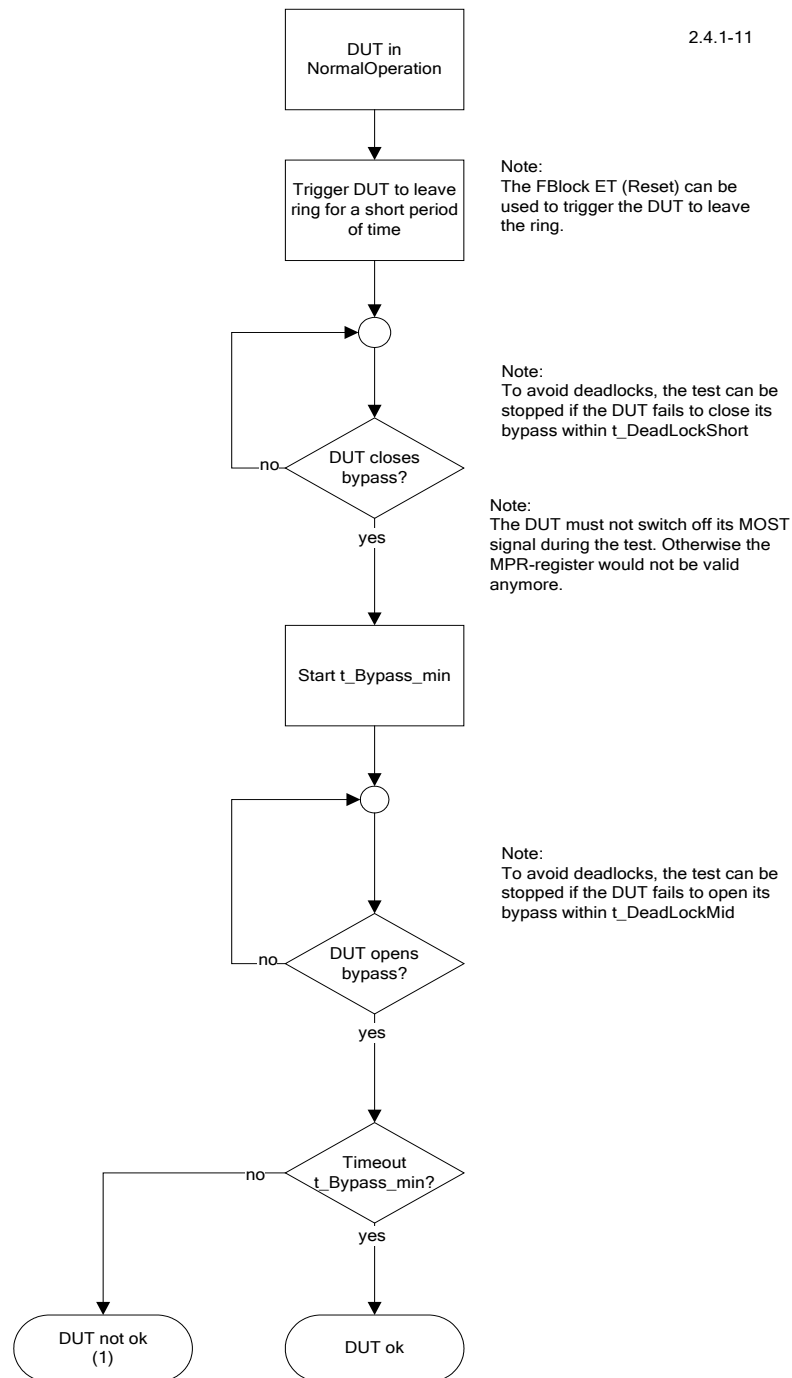


Figure 3-24 Flow of Re-enter delay test.

3.7.2 Error Management – Voltage Level

3.7.2.1 Critical Voltage Test (2.4.2-1)

Name of test	Critical Voltage test 2.4.2-1
Reference to MOST Specification	3.1.5.5 Undervoltage Management
Value of Interest	Voltage Level of Power Supply
Start Conditions	DUT in NormalOperation Voltage level of power supply = U_DUT_Operating
Test description	The voltage level is reduced to U_Critical, then the voltage level is raised to U_DUT_Operating. During the test, the DUT must not switch off the MOST signal or disturb communication. Then the voltage level will be increased to U_DUT_Operating_max. Until U_DUT_Operating_max is reached, the DUT must not switch off or disturb communication.
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in slave mode
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices
Note	1. The voltage levels (U_Critical, U_DUT_Operating, U_DUT_Operating_max) are provided by the DUT manufacturer (see DUT Manufacturer Information List) 2. Increasing and decreasing voltage must be done in steps of 0.1V/0.5s
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT switches off MOST signal too early (before U_Critical). DUT not ok (2): DUT disturbs communication in case of U_DUT_Operating → U_Critical. DUT not ok (3): DUT disturbs communication in case of U_Critical → U_DUT_Operating. DUT not ok (4): DUT switches off too early (before U_DUT_Operating_max) DUT not ok (5): DUT disturbs communication in case of U_DUT_Operating → U_DUT_Operating_max.

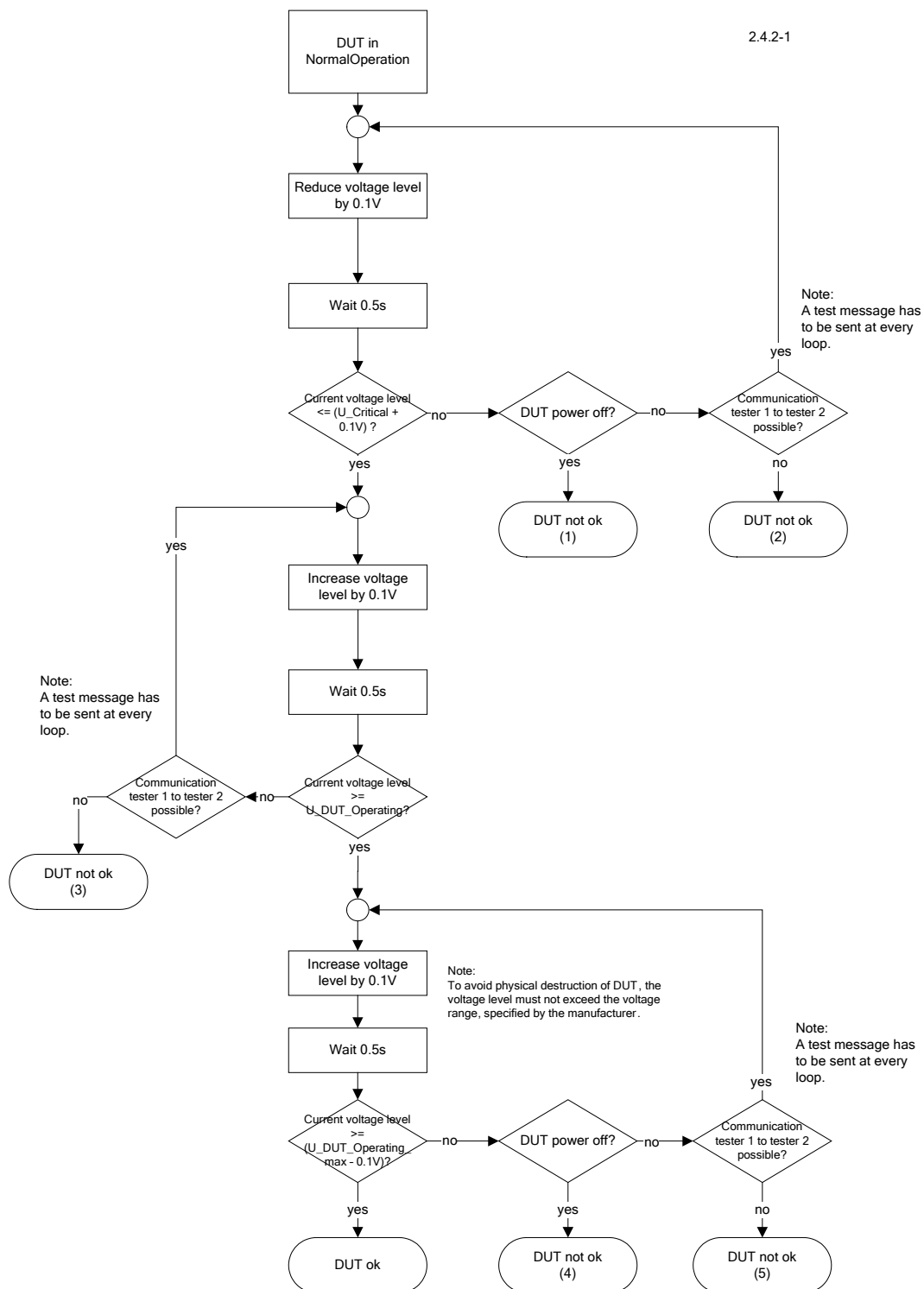


Figure 3-25 Flow of Critical Voltage test.

3.7.2.2 Low Voltage Test (2.4.2-3)

Name of test	Low Voltage test 2.4.2-3
Reference to MOST Specification	3.1.5.5 Undervoltage Management
Value of Interest	Voltage Level of Power Supply
Start Conditions	DUT in NormalOperation Voltage level of power supply = U_DUT_Operating
Test description	During NormalOperation, the voltage level is reduced slowly (0.1V/0.5s). As long as the DUT continues sending MOST signal, the communication between tester 1 and tester 2 has to be possible. If the voltage level drops below U_Low, the DUT is allowed to switch off the MOST signal. The voltage level is set to U_DUT_Operating slowly (0.1V/0.5s) and a wake-up event is generated. The DUT has to start generating the MOST signal.
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in slave mode
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices
Note	1. The voltage levels (U_DUT_Operating, U_Low) are provided by the DUT manufacturer. 2. Increasing and decreasing voltage must be done in steps of 0.1V/0.5s
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT disturbs communication before MOST signal is switched off. DUT not ok (2): DUT fails to meet requirement of voltage level U_Low (switches off too late / too early). DUT not ok (3): DUT fails to process a wake-up after entering U_DUT_Operating state from U_Low state.

2.4.2-3

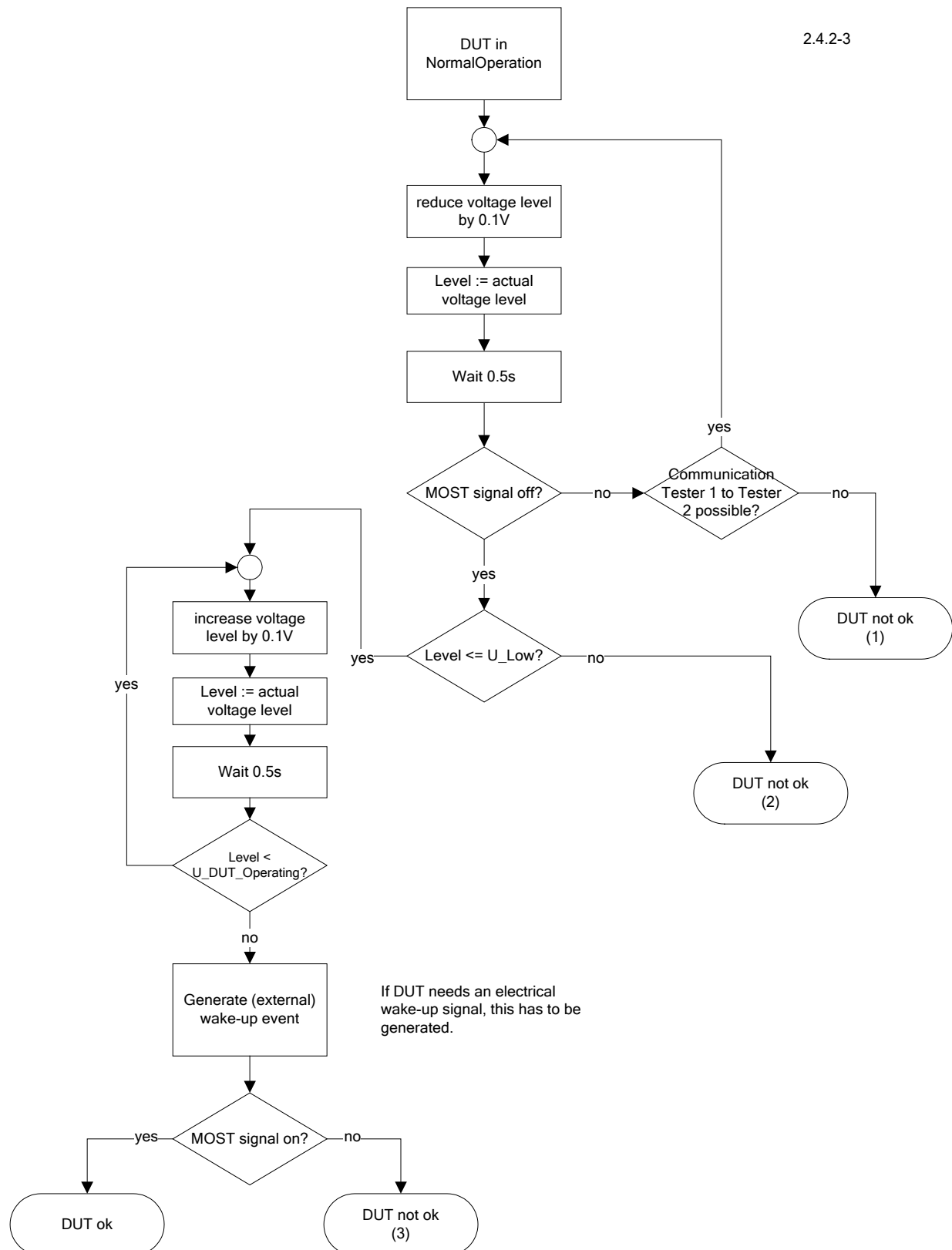


Figure 3-26 Flow of Low Voltage test.

3.8 Ring Break Diagnosis

3.8.1.1 RBD Procedure Test (2.5-2) MOST50 ePHY

Name of test	RBD procedure test MOST50 ePHY 2.5-2
Reference to MOST Specification	8.1.9 Ring Break Diagnosis
Value of Interest	t_Diag_Slave t_Diag_Master t_PwrSwitchOffDelay RBD result
Start Conditions	DUT in NormalOperation Ring closed
Test description	<p>The DUT is triggered to start the RBD and the ring is interrupted before the DUT. It has to perform and finish the RBD within (t_Diag_Slave_max or t_Diag_Master_max) + t_PwrSwitchOffDelay_max. At the end of the RBD, it has to switch off MOST signal. The ring is closed and the tester turns on the MOST signal to read out the result of the RBD from the DUT. The result has to show an interruption of the ring right before the DUT.</p> <p>The tester triggers the RBD again and powers down the ring. The DUT has to perform and finish the RBD within t_Diag_Slave_max / t_Diag_Master_max. At the end of the RBD, the DUT has to perform NormalOperation (detectable by polling the FBlocks). It must not stop generating MOST signal.</p> <p>The RBD result has to show a closed ring.</p>
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices supporting RBD <input type="checkbox"/> MOST150 oPHY <input type="checkbox"/> MOST150 cPHY <input checked="" type="checkbox"/> MOST50 ePHY
Multi Node Device	All devices supporting RBD
Note	
Results	<p>DUT ok: The DUT has passed the test.</p> <p>DUT not ok (1): DUT fails to finish RBD within t_Diag_Master / t_Diag_Slave (ring interrupted).</p> <p>DUT not ok (2): Wrong RBD result (in case of interrupted ring).</p> <p>DUT not ok (3): DUT fails to finish RBD within t_Diag_Master / t_Diag_Slave (ring closed).</p> <p>DUT not ok (4): DUT stops generating MOST signal.</p> <p>DUT not ok (5): Wrong RBD result (in case of closed ring).</p>

2.5-2

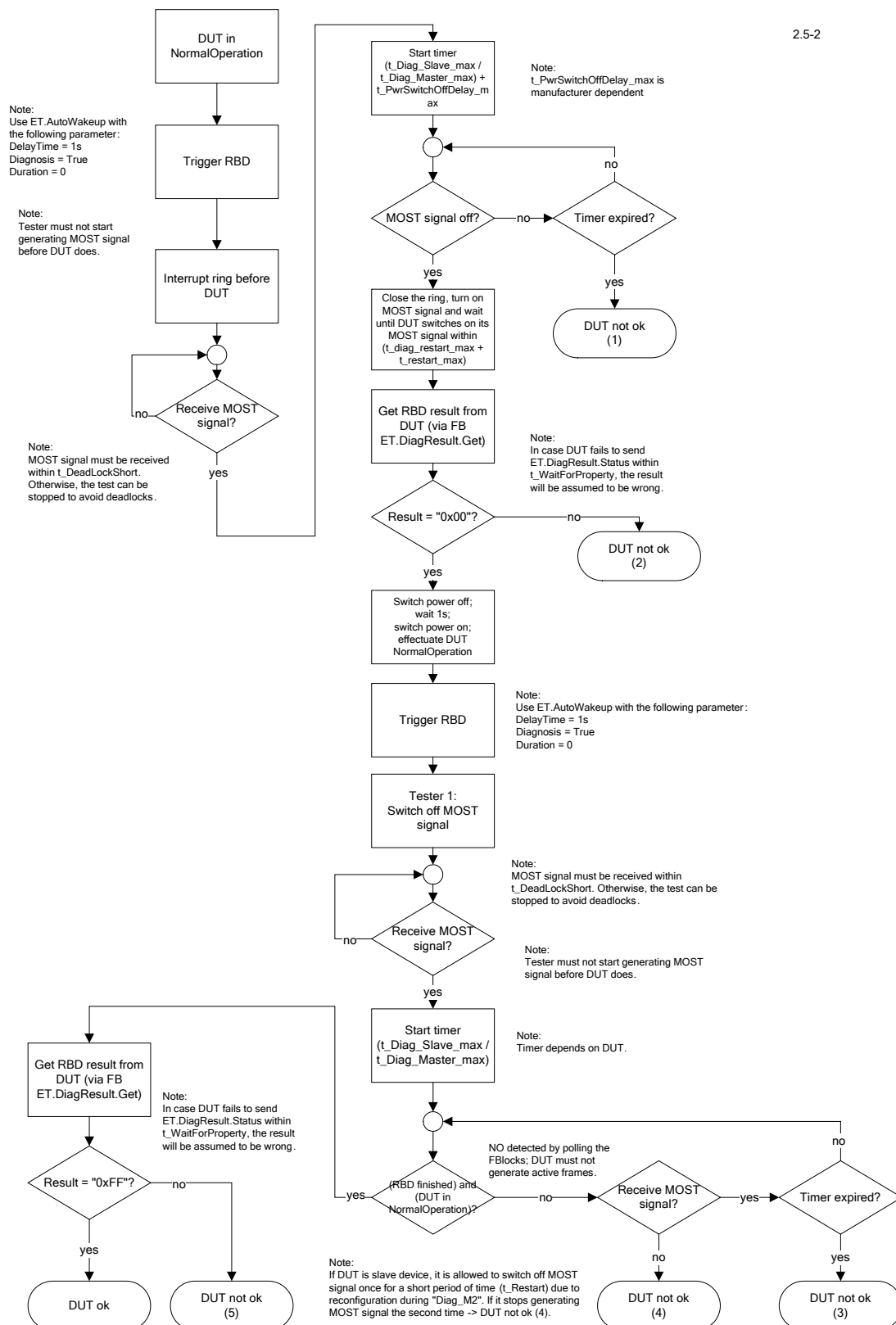


Figure 3-27 Flow of RBD procedure test.

3.8.1.2 TM RBD Procedure Test (2.5-3) MOST50 ePHY

Name of test	TM RBD procedure test MOST50 ePHY 2.5-3
Reference to MOST Specification	8.1.9 Ring Break Diagnosis
Value of Interest	t_Diag_Slave t_Diag_Master t_PwrSwitchOffDelay RBD result
Start Conditions	DUT in NormalOperation Ring closed
Test description	<p>The DUT is triggered to start the RBD and the ring will be interrupted between Tester 2 and Tester 1. The DUT has to perform and finish the RBD within (t_Diag_Slave_max or t_Diag_Master_max) + t_PwrSwitchOffDelay_max. At the end of the RBD, it has to switch off MOST signal.</p> <p>The ring is closed and the tester 1 turns on the MOST signal to read out the result of the RBD from the DUT. The result has to show an interruption of the ring right before Tester 1.</p> <p>The tester triggers the RBD again and powers down the ring. The DUT has to perform and finish the RBD within t_Diag_Slave_max / t_Diag_Master_max. At the end of the RBD, the DUT has to perform NormalOperation (detectable by polling the FBlocks). It must not stop generating MOST signals.</p> <p>The RBD result has to show a closed ring.</p>
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in slave mode
Device type	<p>All devices supporting RBD</p> <p>[] MOST150 oPHY [] MOST150 cPHY [x] MOST50 ePHY</p>
Multi Node Device	All devices supporting RBD
Note	Tester 1 has to behave like normal MOST device regarding to RBD procedure (based on MOST Specification Rev. 3.0E2 Addendum A; MOST50 Adaption).
Results	<p>DUT ok: The DUT has passed the test.</p> <p>DUT not ok (1): DUT fails to finish RBD within t_Diag_Master / t_Diag_Slave (ring interrupted).</p> <p>DUT not ok (2): Wrong RBD result (in case of interrupted ring).</p> <p>DUT not ok (3): DUT fails to finish RBD within t_Diag_Master / t_Diag_Slave (ring closed).</p> <p>DUT not ok (4): DUT stops generating MOST signal.</p> <p>DUT not ok (5): Wrong RBD result (in case of closed ring).</p>

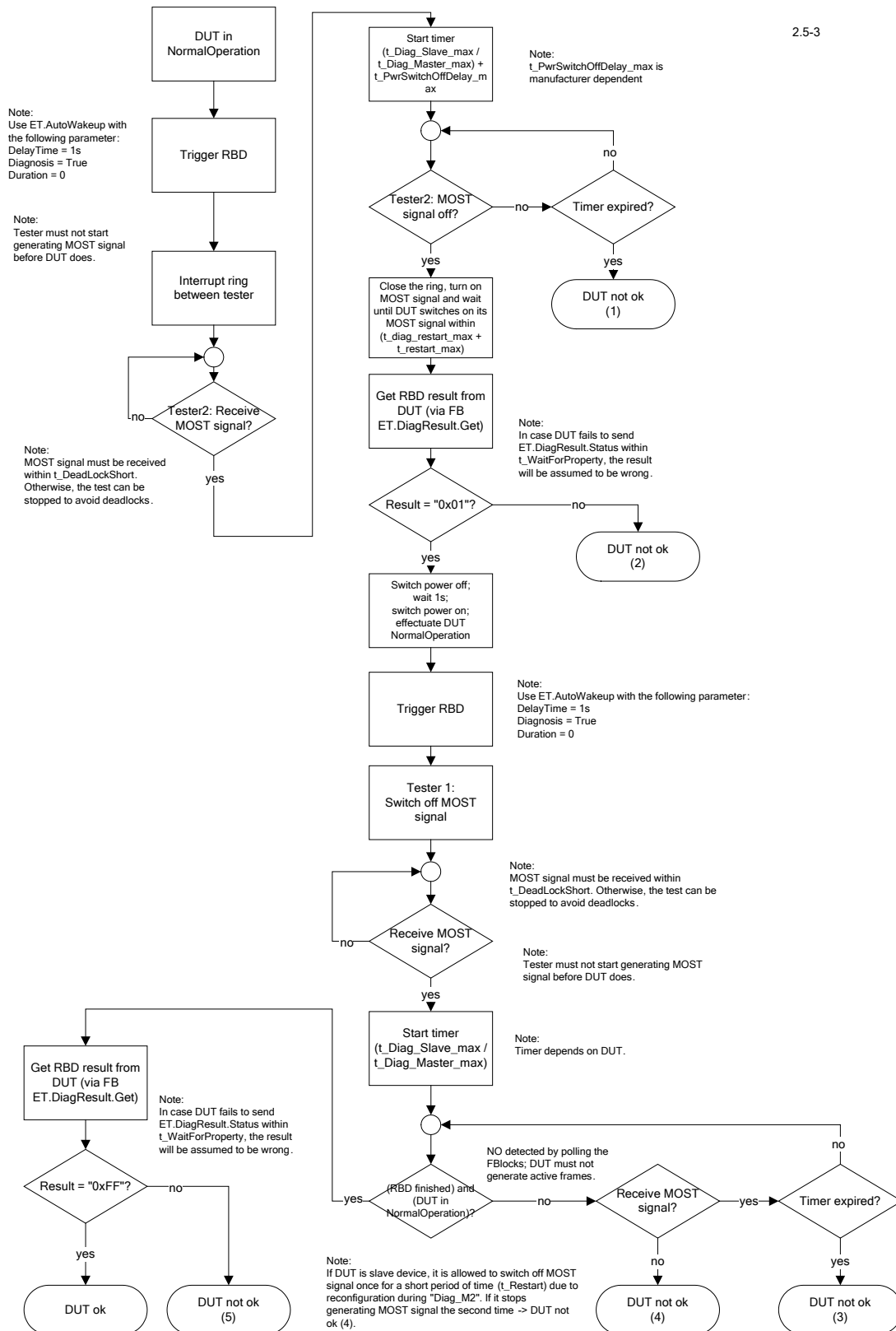


Figure 3-28 Flow of TM RBD procedure test.

3.8.1.3 Wakeup After RBD Test (2.5-7) MOST50 ePHY

Name of test	Wakeup after RBD test MOST50 ePHY 2.5-7
Reference to MOST Specification	8.1.9 Ring Break Diagnosis
Value of Interest	(t_Diag_Restart + t_Restart)
Start Conditions	DUT in NormalOperation Ring closed
Test description	The DUT is triggered to start the RBD. If the RBD is finished, the DUT has to switch off MOST signal. The tester starts generating MOST signal. The DUT has to be able to detect this wake-up event. It has to wait for (t_Diag_Restart_min + t_Restart) before generating MOST signal at its output.
Experimental set-up	- Tester 1 in slave mode or in master mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices supporting RBD [] MOST150 oPHY [] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices supporting RBD
Note	Test not applicable if DUT does not support wake-up via MOST interface or electrical wake-up.
Results	DUT ok: The DUT has passed the test. DUT not ok: DUT fails to meet (t_Diag_Restart + t_Restart) after RBD

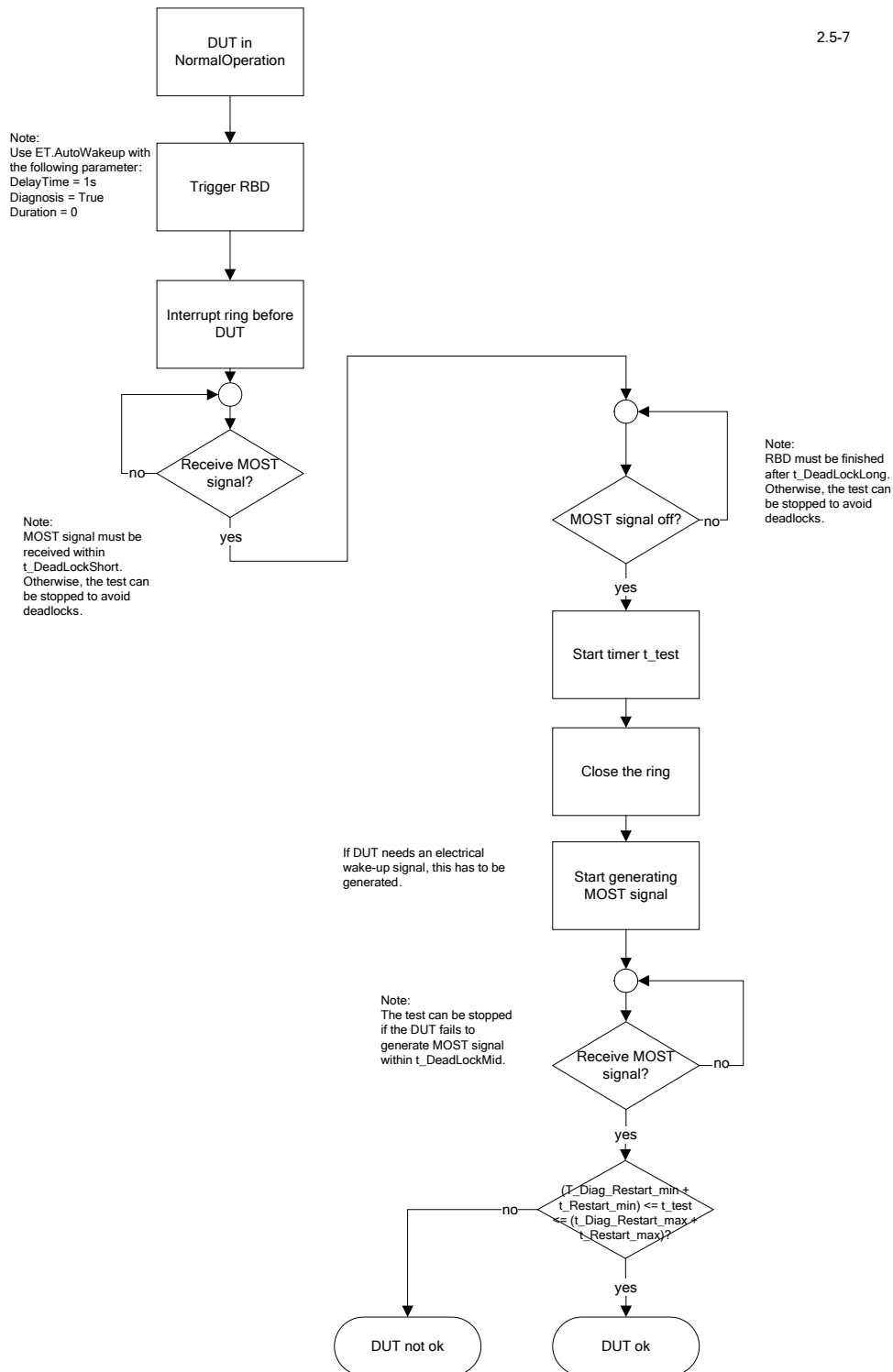


Figure 3-29 Flow of Wakeup after RBD test.

3.8.1.4 RBD Missing TM Test (2.5-8) MOST50 ePHY

Name of test	RBD missing TM test MOST50 ePHY 2.5-8
Reference to MOST Specification	8.1.9 Ring Break Diagnosis
Value of Interest	RBD result t_Diag_Slave t_Diag_Master
Start Conditions	DUT in NormalOperation Ring closed
Test description	The DUT is triggered to start the RBD. If the RBD is finished, the DUT has to switch off MOST signal. The DUT is woken-up again and the result of the RBD is checked. It must not indicate "fully operational network (0xFF)".
Experimental set-up	- Tester 1 in master mode - Tester 2 in spy mode
Device type	All devices supporting RBD except TM. [] MOST150 oPHY [] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices supporting RBD without TM.
Note	Tester 1 has to behave like normal MOST device regarding to RBD procedure for TimingSlave (based on MOST Specification Rev. 3.0E2 Addendum A; MOST50 Adaption).
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT fails to finish RBD right in time. DUT not ok (2): Wrong RBD-result.

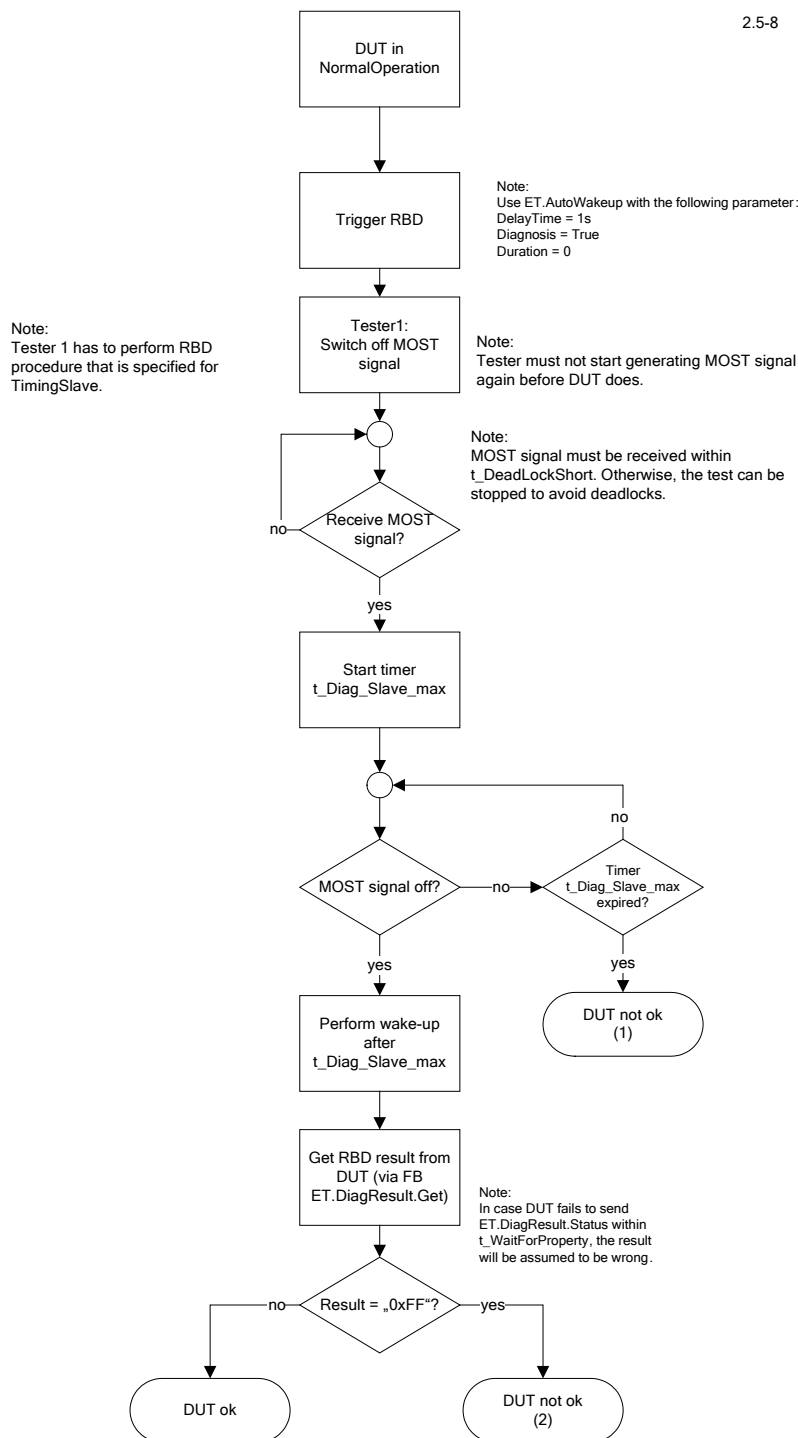


Figure 3-30 Flow of RBD missing TM test.

3.8.1.5 RBD Closed Ring Test (2.5-10) MOST150 oPHY

Name of test	RBD closed ring test MOST150 oPHY 2.5-10
Reference to MOST Specification	3.1.4.1 Ring Break Diagnosis
Value of Interest	RBD result
Start Conditions	DUT in NormalOperation Ring closed
Test description	<p>The DUT is triggered to start the RBD. The tester has to perform normal RBD procedure (DUT = TM → tester has to perform RBD for TimingSlave; DUT <> TM → Tester has to perform RBD for TimingMaster).</p> <p>At the end of RBD, the DUT has to enter NetInterfaceNormalOperation and must not send NetBlock.RBDResult.Status by itself. The result of RBD will be requested via ET.DiagResult.Get and checked for correctness (= no error).</p>
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in spy mode
Device type	<p>All devices performing RBD</p> <p>[x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY</p>
Multi Node Device	All devices performing RBD
Note	<p>For MOST150 oPHY, RBD tests are optional. DUT manufacturer information will state if RBD tests are performed.</p>
Results	<p>DUT ok: The DUT has passed the test. DUT not ok (1): DUT reports RBD result via NetBlock.RBDResult.Status DUT not ok (2): Wrong RBD result reported by DUT via ET.DiagResult.Status</p>

2.5-10

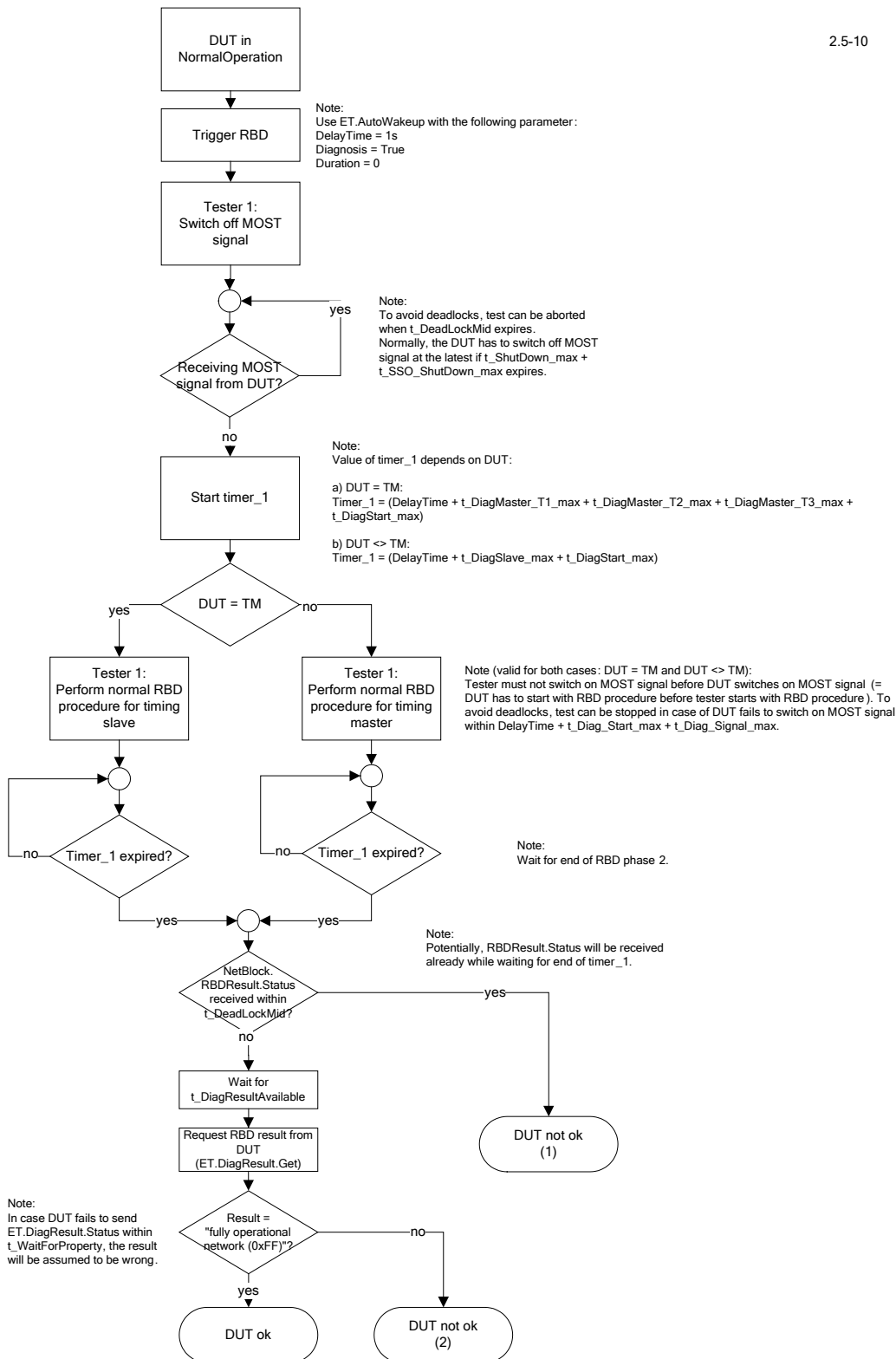


Figure 3-31 Flow of RBD closed ring test.

3.8.1.6 RBD Interruption In Front Of DUT Test (2.5-11) MOST150 oPHY

Name of test	RBD interruption in front of DUT test MOST150 oPHY 2.5-11
Reference to MOST Specification	3.1.4.1 Ring Break Diagnosis
Value of Interest	RBD result
Start Conditions	DUT in NormalOperation Ring closed
Test description	<p>The DUT is triggered to start the RBD and the ring will be interrupted in front of the DUT. The tester has to perform normal RBD procedure (DUT = TM → tester has to perform RBD for TimingSlave; DUT <> TM → Tester has to perform RBD for TimingMaster).</p> <p>If DUT sends NetBlock.RBDResult.Status by itself within t_DeadLockMid, the result will be evaluated and has to be correct.</p> <p>The ring is closed and the tester performs a wake-up and sets DUT to NormalOperation. The result of RBD will be requested via ET.DiagResult.Get and checked for correctness (relative node position with respect to the ring break)</p>
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in spy mode
Device type	<p>All devices performing RBD, but in Multi Node Device only the first node.</p> <p>[x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY</p>
Multi Node Device	All devices performing RBD
Note	<p>For MOST150 oPHY, RBD tests are optional.</p> <p>DUT manufacturer information will state if RBD tests are performed.</p>
Results	<p>DUT ok: The DUT has passed the test.</p> <p>DUT not ok (1): Wrong RBD result reported by DUT via NetBlock.RBDResult.Status</p> <p>DUT not ok (2): Wrong RBD result reported by DUT via ET.DiagResult.Status</p>

2.5-11

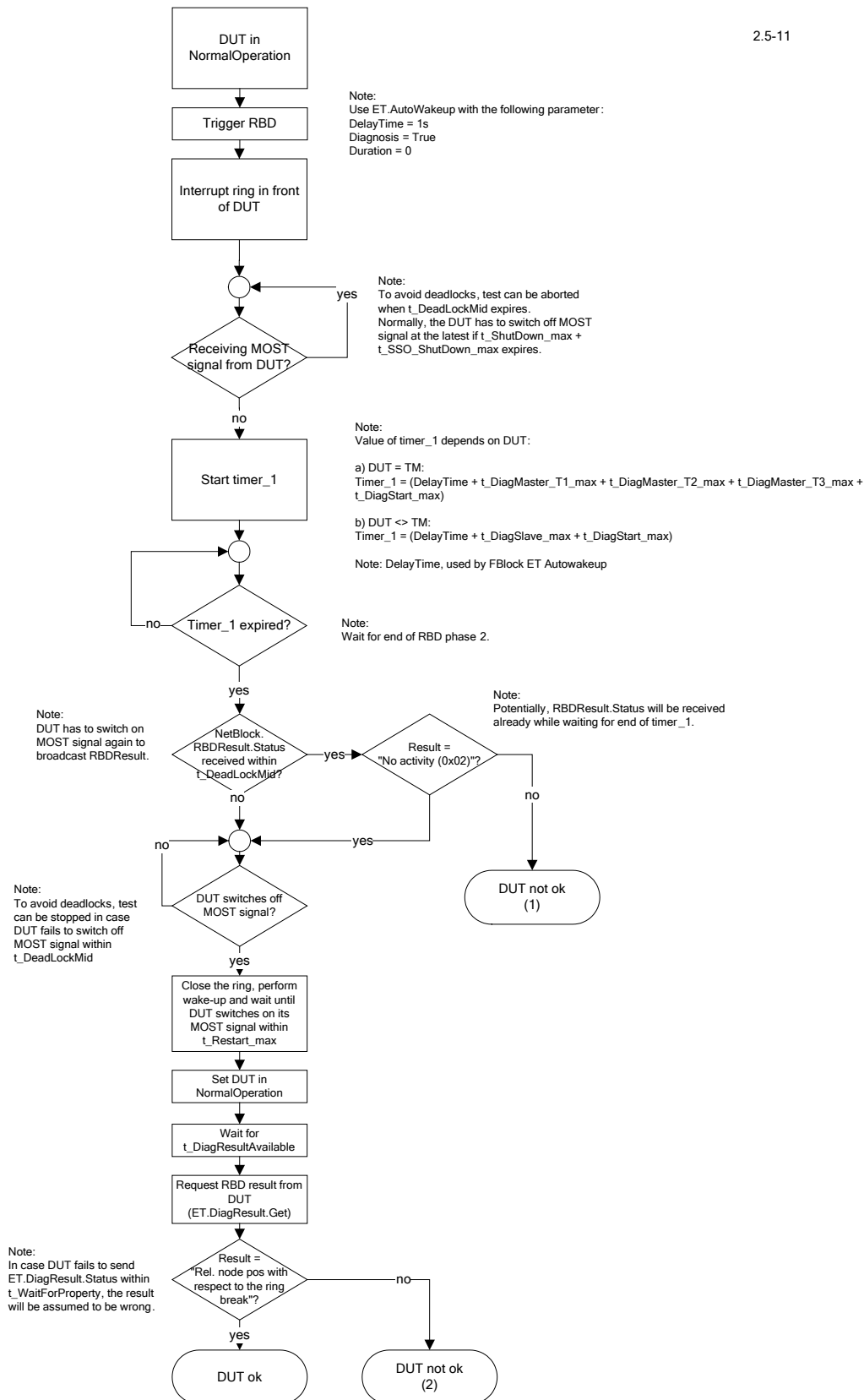


Figure 3-32 Flow of RBD interruption in front of DUT test.

3.8.1.7 RBD Interruption Behind DUT Test (2.5-12) MOST150 oPHY

Name of test	RBD interruption behind DUT test MOST150 oPHY 2.5-12
Reference to MOST Specification	3.1.4.1 Ring Break Diagnosis
Value of Interest	RBD result
Start Conditions	DUT in NormalOperation Ring closed
Test description	The DUT is triggered to start the RBD. The tester1 has to perform normal RBD procedure (DUT = TM → tester has to perform RBD for TimingSlave; DUT <> TM → Tester has to perform RBD for TimingMaster). At the end of RBD, the DUT must not send NetBlock.RBDResult.Status by itself.
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in slave mode
Device type	All devices performing RBD [x] MOST150 oPHY [x] MOST150 cPHY [] MOST50 ePHY
Multi Node Device	All devices performing RBD
Note	For MOST150 oPHY, RBD tests are optional. DUT manufacturer information will state if RBD tests are performed.
Results	DUT ok: The DUT has passed the test. DUT not ok: DUT reports RBD result via NetBlock.RBDResult.Status

2.5-12

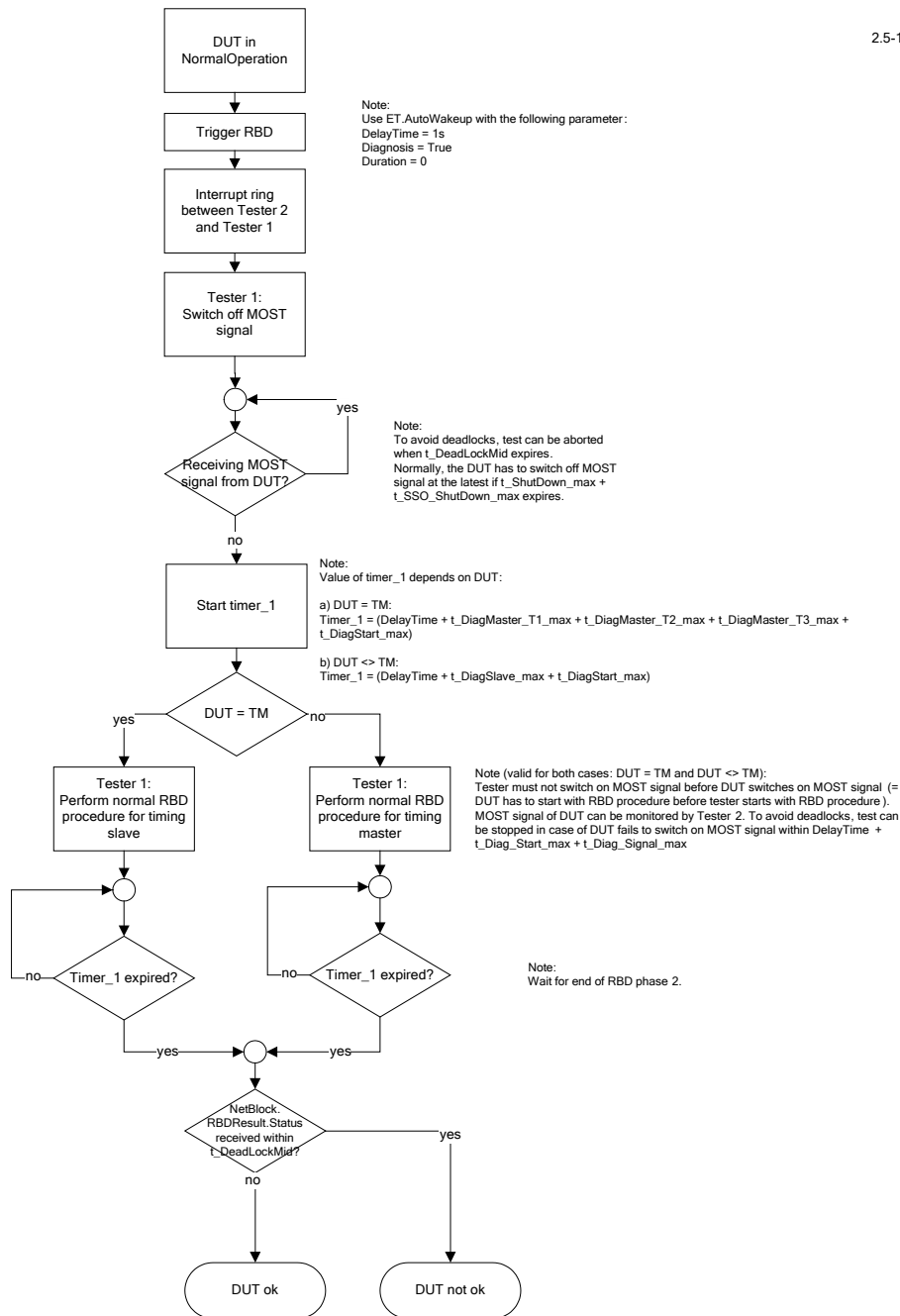


Figure 3-33 Flow of RBD interruption behind DUT test.

3.9 System Configuration

3.9.1 Configuration / System Configuration (NWM)

3.9.1.1 NWM Address Initialization Test (2.6.1-1)

Name of test	NWM address initialization test 2.6.1-1
Reference to MOST Specification	
Value of Interest	Address of DUT
Start Conditions	DUT in NormalOperation Ring closed
Test description	During NO, the tester stores the address of the DUT. After the next wake-up, the DUT has to have still the same address. The DUT is disconnected from power until the buffer capacitors are empty (at least 1 minute). After the next wake-up, the DUT must not use address "uninitialized_node_address". The address, used by the DUT has to be valid.
Experimental set-up	- Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in spy mode
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT fails to store its address during power-off mode. DUT not ok (2): The DUT uses address "uninitialized_node_address" at first power-on. DUT not ok (3): The DUT uses an invalid address at first power-on (address not within valid range).

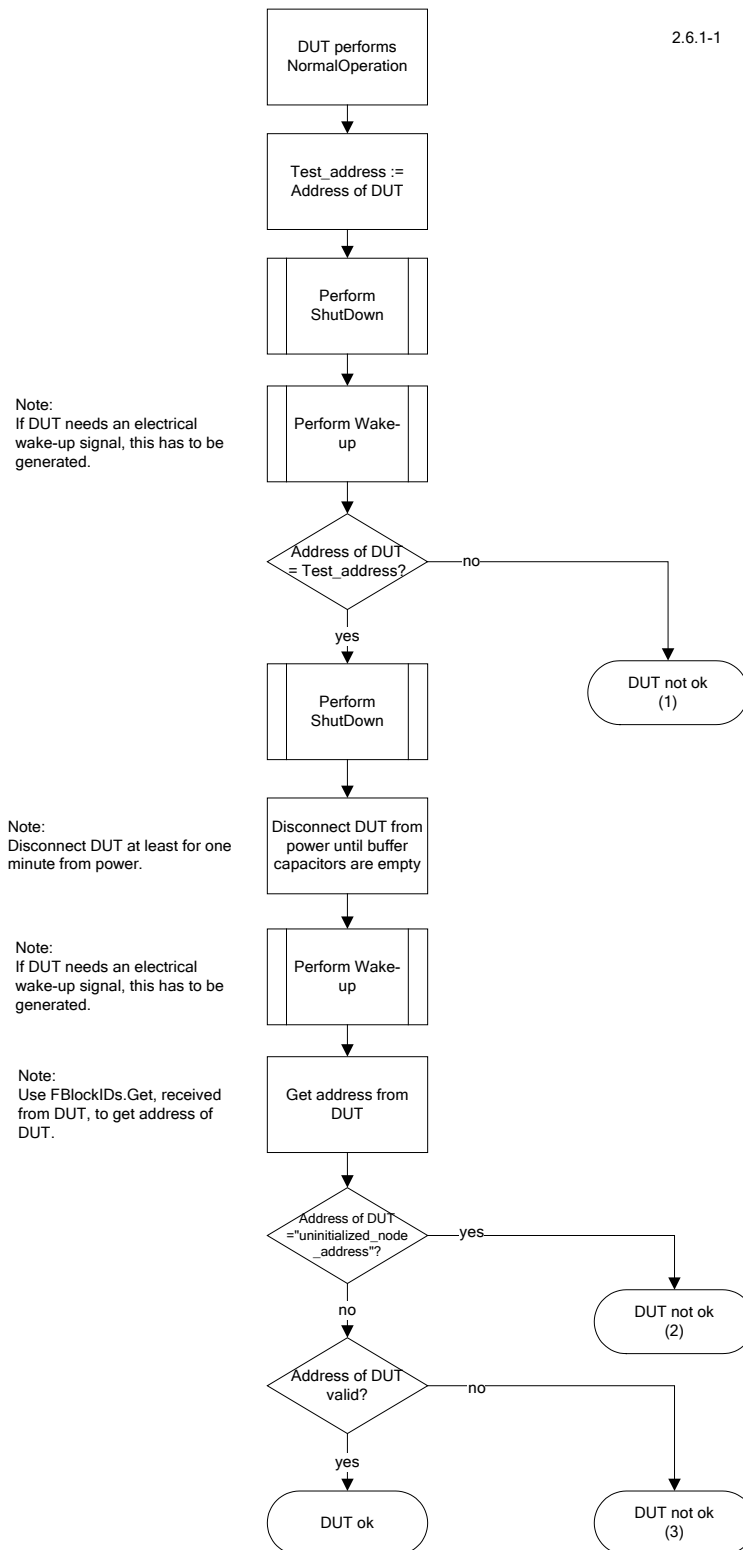


Figure 3-34 Flow of NWM address initialization test.

3.9.1.2 FBlock Polling Test (2.6.2-1)

Name of test	FBlock polling test 2.6.2-1
Reference to MOST Specification	3.1.3.3.4.1 Configuration Request Description
Value of Interest	t_WaitForAnswer FBlockIDs.Get Configuration.Status(ok)
Start Conditions	DUT: NetInterface Off Ring closed
Test description	Tester 2 does not respond to the FBlock check. As soon as the DUT asks tester 2, t_WaitForAnswer_min will be started. Tester 1 has to respond to the FBlock request by the DUT normally. As the CR was build-up successfully, the DUT has to broadcast Configuration.Status(Ok) not before t_WaitForAnswer_min expires.
Experimental set-up	- Tester 1 in slave mode or in master mode (depends on DUT) - Tester 2 in slave mode
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT fails to check the FBlocks of Tester 2. DUT not ok (2): The DUT fails to check the FBlocks of Tester 1. DUT not ok (3): The DUT fails to wait long enough for response of tester 1 t_WaitForAnswer_min). DUT not ok (4): The DUT fails to send Configuration.Status(Ok).

2.6.2-1

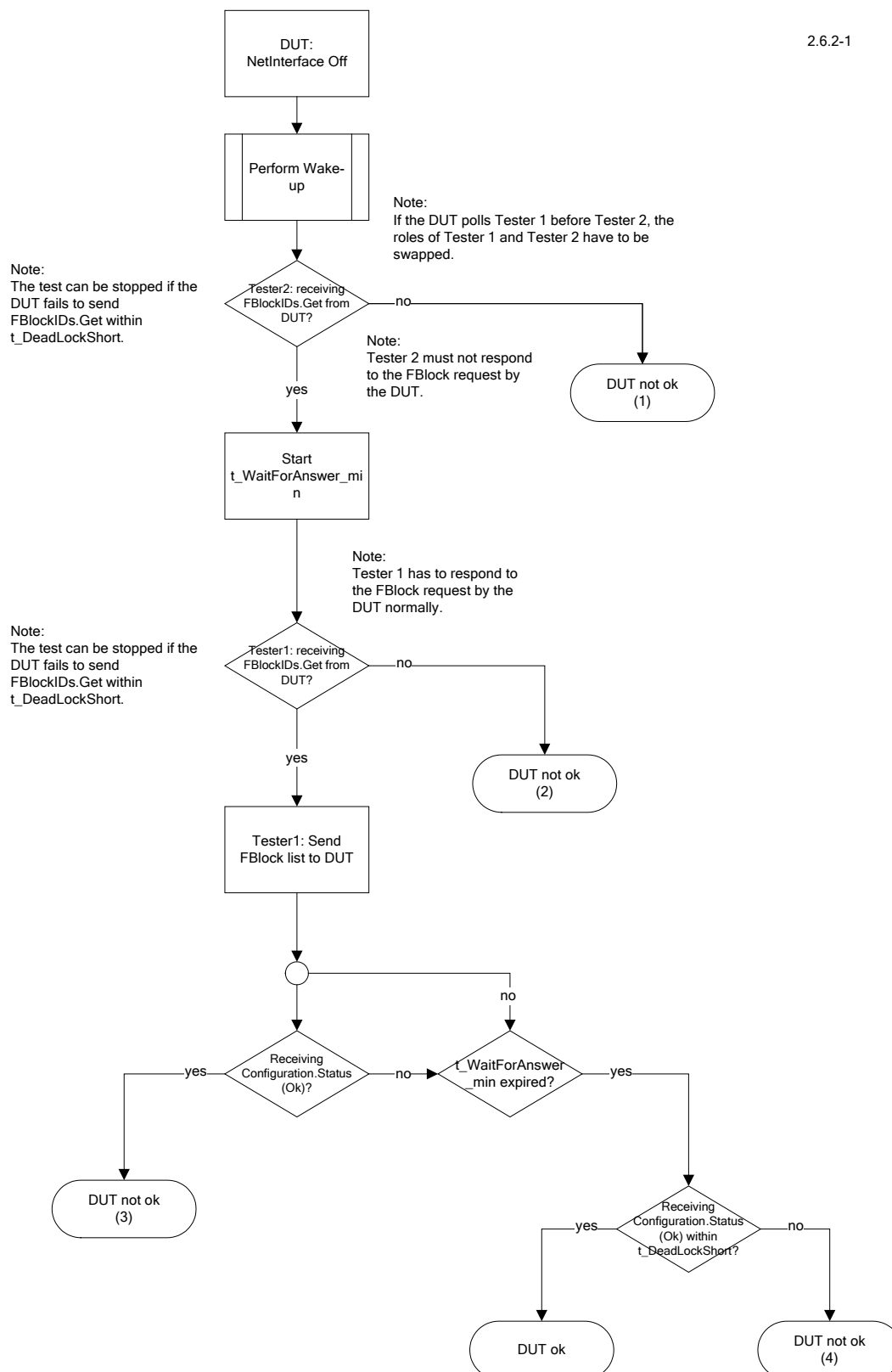


Figure 3-35 Flow of FBlock polling test.

3.9.1.3 Device Ignore Test (a) (2.6.2-3a)

Name of test	Device Ignore test (a) 2.6.2-3a
Reference to MOST Specification	3.1.3.3.4.5 Network Slave Continuous causes for System State NotOk 3.1.3.3.5.1 Un-initialized Logical Node Address
Value of Interest	CR Configuration.Status(NotOk) Configuration.Status(Ok)
Start Conditions	DUT: NetInterface Off Ring closed
Test description	After wake-up, the DUT has to detect that Tester 1 has address "uninitialized_node_address" and has to broadcast Config(NotOK). The Tester does not change its address. The DUT has to check the tester two more times. Then it has to ignore it until the next NCE or system startup (the tester must not be stored into the CR) and has to send Configuration.State(Ok).
Experimental set-up	- Tester 1 in slave mode or in master mode (depends on DUT); Address="uninitialized_node_address" - Tester 2 in slave mode
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT fails to check Tester 1. DUT not ok (2): DUT fails to broadcast Configuration.Status(NotOK). DUT not ok (3): DUT checks the Tester less than three times. DUT not ok (4): DUT fails to broadcast Configuration.Status(OK). DUT not ok (5): DUT stores Tester 1 into CR.

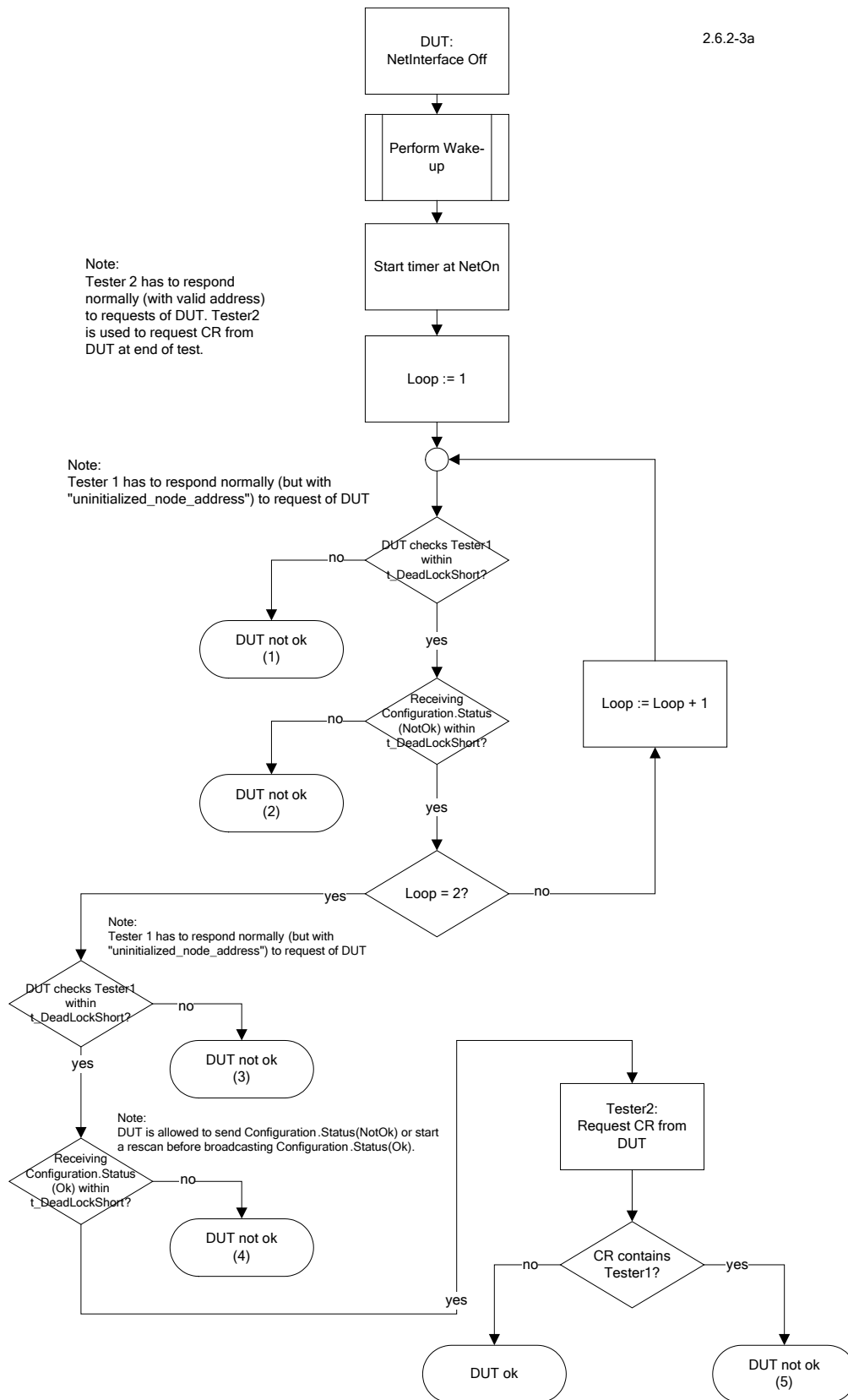


Figure 3-36 Flow of Device Ignore test (a).

3.9.1.4 Device Ignore Test (b) (2.6.2-3b)

Name of test	Device Ignore test (b) 2.6.2-3b
Reference to MOST Specification	3.1.3.3.4.4 Retries of Non Responding Network Slaves
Value of Interest	t_DelayCfgRequest1 t_DelayCfgRequest2 t_WaitForAnswer CR
Start Conditions	DUT: NetInterface Off Ring closed
Test description	After wake-up, the DUT has to detect that Tester 1 does not respond to its requests. The DUT has to check tester 1 at least 10 times with short interval (t_WaitForAnswer + t_DelayCfgRequest1). From the 23. Check on, the DUT has to use long check interval (t_WaitForAnswer + t_DelayCfgRequest2). The DUT must not store Tester 1 into the CR. As soon as the Tester 1 responds to request from DUT, the DUT has to store Tester 1 into the CR.
Experimental set-up	- Tester 1 in slave mode; bypass closed - Tester 2 in slave mode or in master mode (depends on DUT) behaves like normal MOST device
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	Not responding to requests of DUT means: Tester acknowledges reception but does not respond to request from DUT. Test does not focus on border between 20. Check and 21. Check. Test only checks whether DUT changes from short check interval to long check interval. Tester2 has to prevent potential shutdown during test. It has to send ShutDown.Result(Suspend) to the PM in case receiving ShutDown.Start(Query). Note for MOST50 ePHY: Non-Ack-OPTypes are allowed; has to be considered during testing.
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT fails to check Tester 1. DUT not ok (2): DUT checks Tester1 not within t_DelayCfgRequest1_min/max. DUT not ok (3): DUT checks Tester1 not within t_DelayCfgRequest2_min/max. DUT not ok (4): DUT erroneously stores Tester 1 into the CR. DUT not ok (5): DUT fails to detect delayed responding Tester 1. DUT not ok (6): DUT fails to store Tester 1 into the CR.

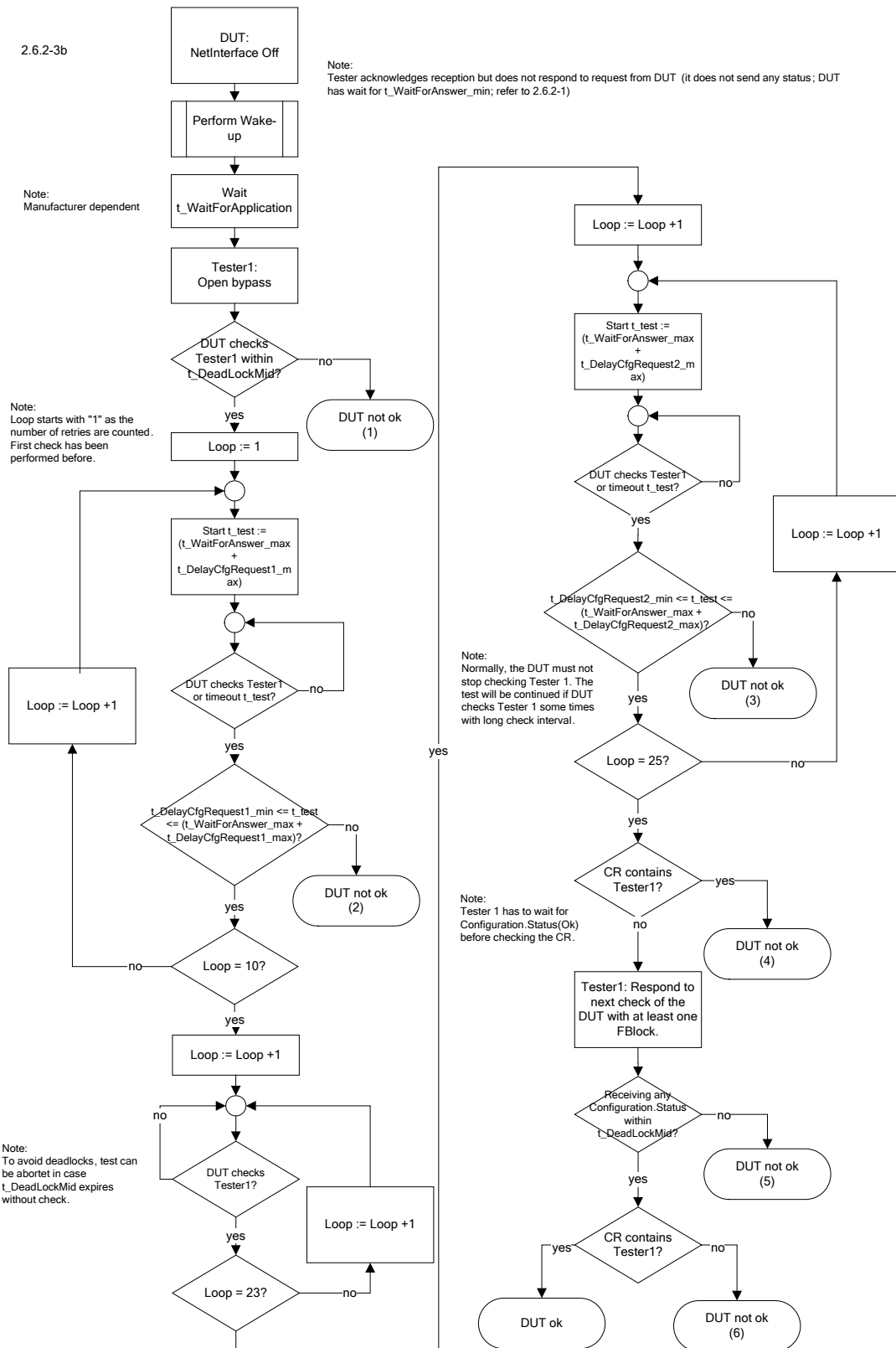


Figure 3-37 Flow of Device Ignore test (b).

3.9.1.5 Device Integration Test (a) (2.6.2-4a)

Name of test	Device Integration test (a) 2.6.2-4a
Reference to MOST Specification	3.1.3.3.4 Scanning the System (System Scan)
Value of Interest	CR FBlockIDs.Get Configuration.Status(ok)
Start Conditions	DUT: NetInterface Off Ring closed
Test description	After wake-up, the DUT has to integrate the tester 1 into the CR and send Config(Ok).
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT); valid address. It has to respond to FBlockIDs.Get from DUT like a normal MOST device - Tester 2 in spy mode
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT fails to send Config(Ok) right in time. DUT not ok (2): DUT fails to store Tester 1 into the CR.

2.6.2-4a

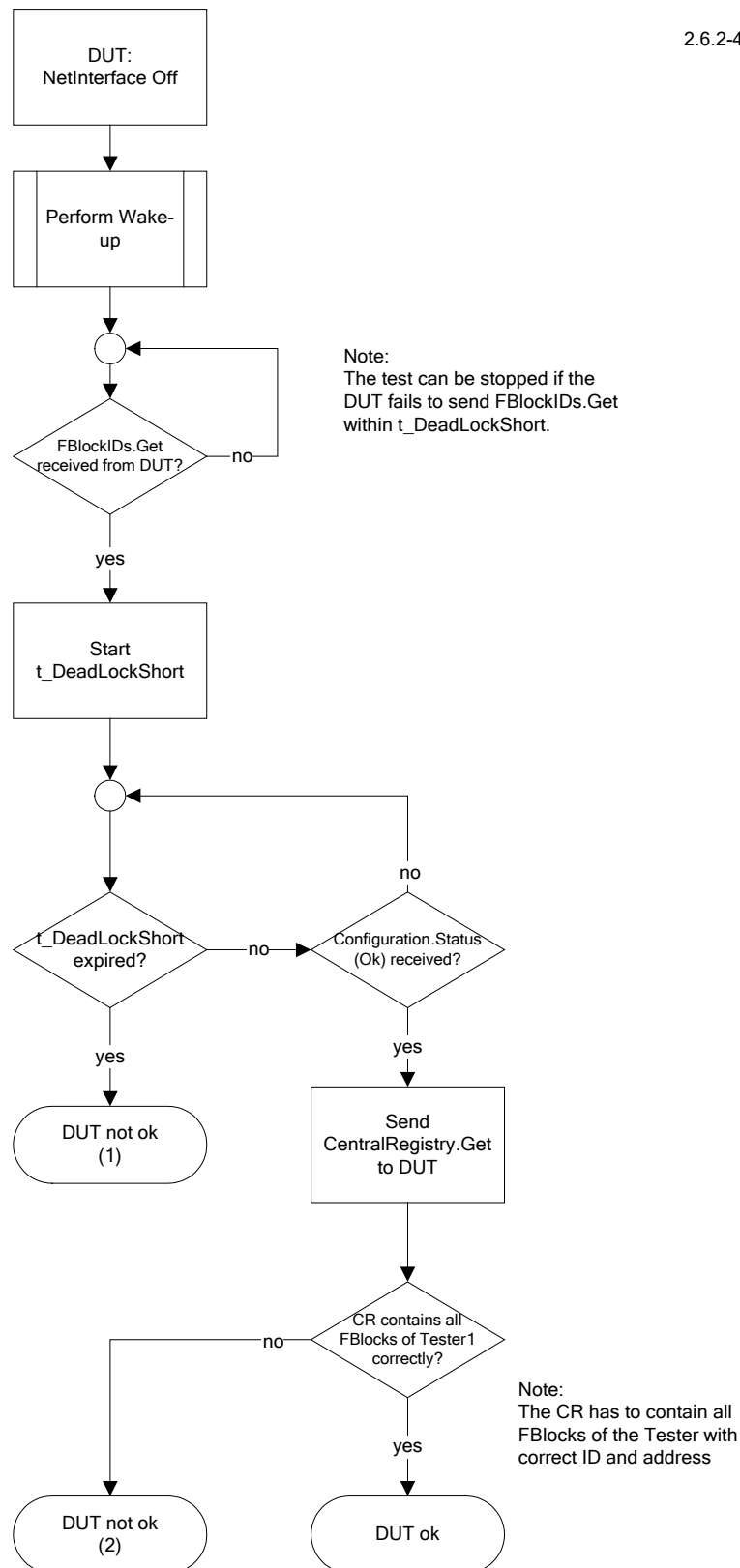


Figure 3-38 Flow of Device Integration test (a).

3.9.1.6 Device Integration Test (b) (2.6.2-4b)

Name of test	Device Integration test (b) 2.6.2-4b
Reference to MOST Specification	3.1.3.3.2 Central Registry 3.1.3.3.6.2 Appearing Function Blocks in System State OK
Value of Interest	CR Configuration.Status(Ok/NewExt)
Start Conditions	DUT: NetInterface Off Ring closed
Test description	At this test, Tester 1 provides more FBlocks than the DUT is able to store into the CR. To fill up the CR, Tester1 provides step by step single FBlocks until DUT does not broadcast any ConfigurationStatus(NewExt,...) message. All FBlocks of Tester1 that have been accepted by the DUT has to be stored into the CR at end of test.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT); valid address. - Tester 2 in slave mode; valid address.
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	If DUT supports dynamic CR (detectable via Fblock ET), the testers must have 257 FBlocks.
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT fails to perform initial scan and/or sending Configuration.Status(Ok) right in time. DUT not ok (2): DUT deletes FBlocks of Tester1 from CR. DUT not ok (3): DUT fails to store all successfully registered FBlocks of Tester1 into the CR.

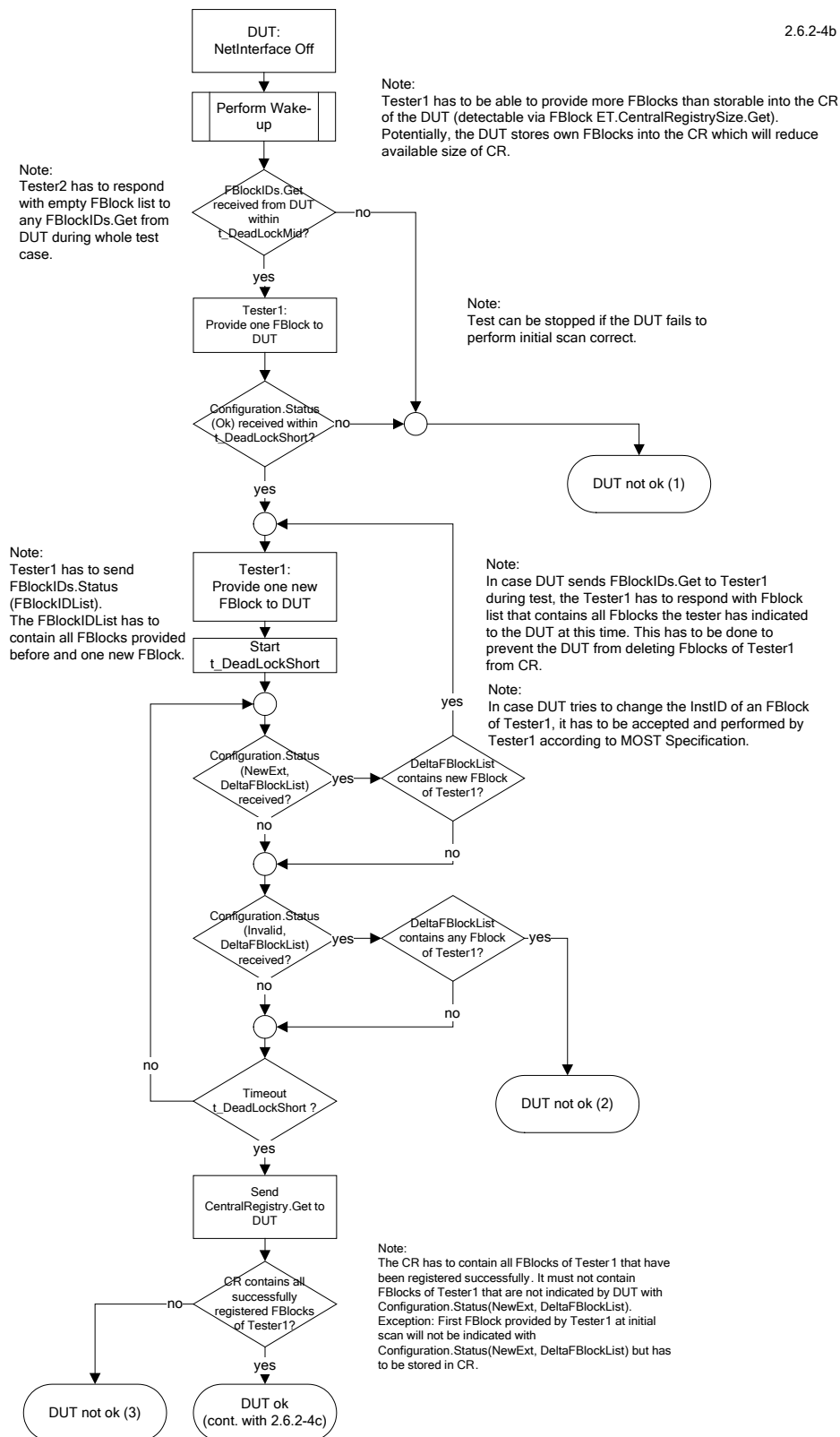


Figure 3-39 Flow of Device Integration test (b).

3.9.1.7 Device Integration Test (c) (2.6.2-4c)

Name of test	Device Integration test (c) 2.6.2-4c
Reference to MOST Specification	3.1.3.3.2 Central Registry 3.1.3.3.6.2 Appearing Function Blocks in System State OK
Value of Interest	CR Configuration.Status(NewExt)
Start Conditions	Continued from 2.6.2-4b; "DUT Ok"
Test description	At this test (continuing from 2.6.2-4b; "DUT Ok"), Tester 2 provides some FBlocks to the DUT until CR fails to store further FBlocks of Tester2. To fill up the CR, Tester2 provides step by step single FBlocks until DUT does not broadcast any ConfigurationStatus(NewExt,...) message. All FBlocks of Tester2 that have been accepted by the DUT has to be stored into the CR at end of test. Additionally, all FBlocks of Tester1 (based on test case 2.6.2-4b) have to be stored in the CR, too.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT); valid address. - Tester 2 in slave mode; valid address.
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	If DUT supports dynamic CR (detectable via Fblock ET), the testers must have 257 FBlocks.
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT deletes FBlocks of Tester1 or Tester2 from CR. DUT not ok (2): DUT fails to store all successfully registered FBlocks of Tester1 and Tester2 into the CR.

2.6.2-4c

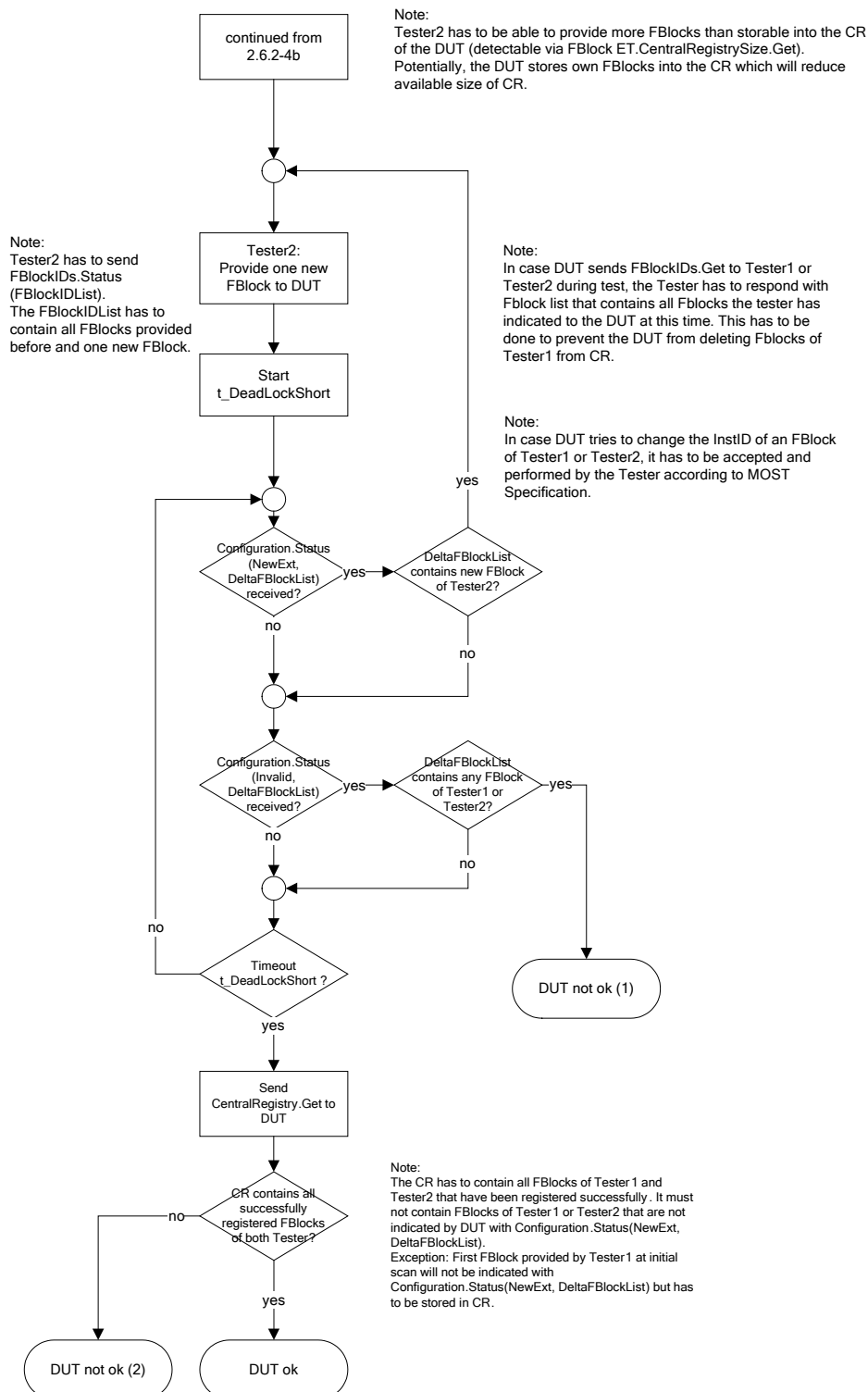


Figure 3-40 Flow of Device Integration test (c).

3.9.1.8 Config(Ok) Delay Test (2.6.2-5)

Name of test	Config(Ok) Delay test 2.6.2-5
Reference to MOST Specification	3.1.3.3.1.2 Setting the System State to NotOk 3.1.3.3.4.3 Non Responding Network Slaves
Value of Interest	t_WaitBeforeRescan t_WaitForAnswer
Start Conditions	DUT in NormalOperation
Test description	Tester 2 triggers Configuration.Status(NotOk) by sending FBlockIDs.Status to DUT, using sender address 0xFFFF. The DUT has to send Configuration.Status(NotOk). Then it has to start a system scan, before t_WaitBeforeRescan_max expires. During this system scan, tester 2 must not respond to DUT. The DUT must not send Configuration.Status(Ok) before t_WaitForAnswer_max expires.
Experimental set-up	- Tester 1 in slave mode - Tester 2 in slave mode
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	Tester 1 has to behave like normal MOST device
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT fails to send Configuration.Status(NotOk). DUT not ok (2): The DUT starts system scan too late. DUT not ok (3): DUT sends Configuration.Status(Ok) too early.

2.6.2-5

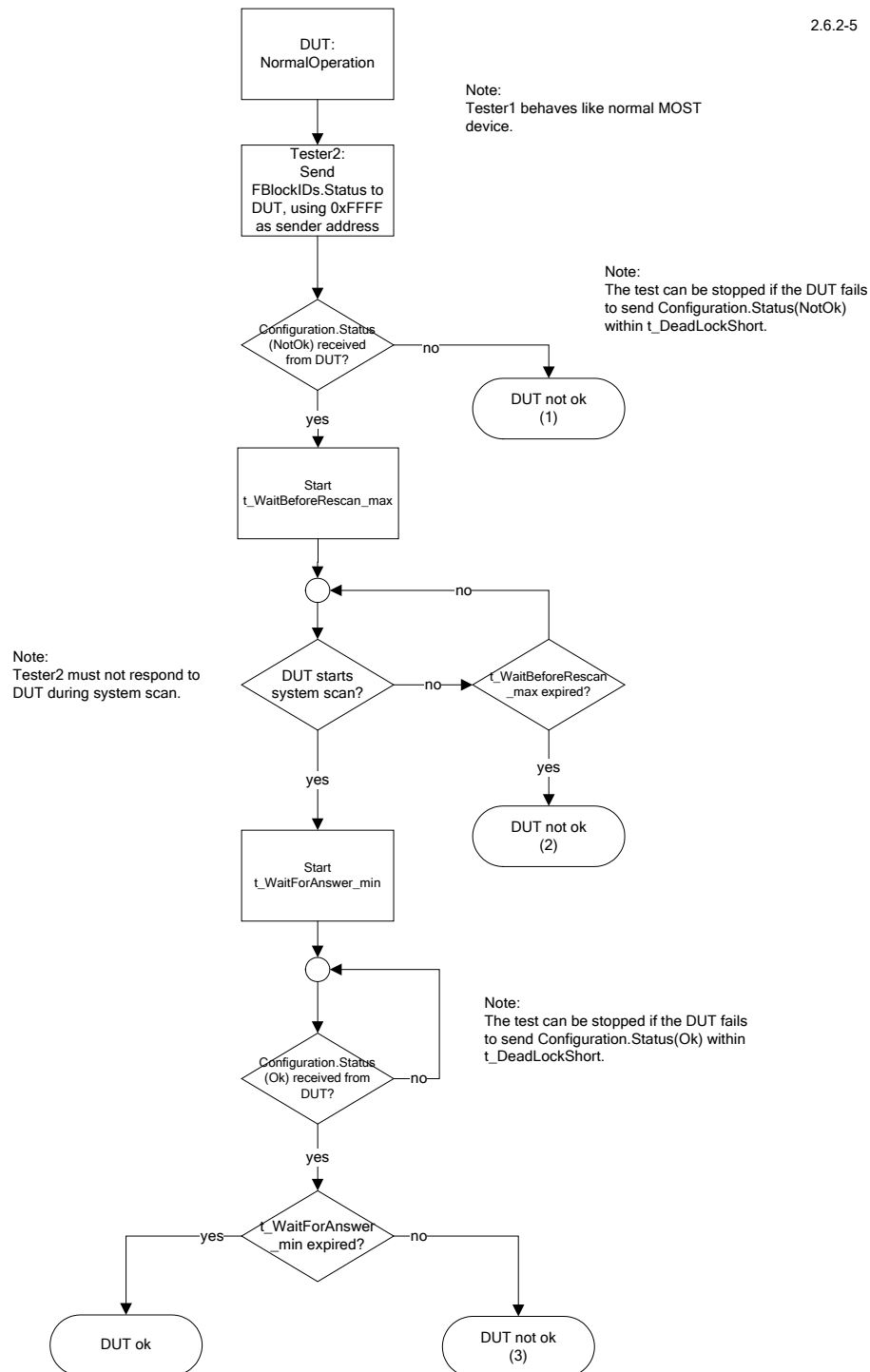


Figure 3-41 Flow of Config(Ok) Delay test.

3.9.1.9 Double FBlock Test (2.6.2-6)

Name of test	Double FBlock test 2.6.2-6
Reference to MOST Specification	3.1.3.3.5.4 Duplicate InstID Registrations
Value of Interest	CR FBlockIDs.Get Configuration.Status(Ok)
Start Conditions	DUT: NetInterface Off Ring closed
Test description	After wake-up, the DUT has to perform system scan. It has to scan the devices in correct order. Tester 1 and Tester 2 send the same FBlock-ID, Inst-ID to the DUT. The DUT has to detect the identical FBlocks and has to try to change the InstID of the FBlock of Tester 1 or tester 2. The tester must not change its InstID. As soon as the DUT sends Config(ok), the CR will be checked. It must contain the "normal FBlocks" of both tester. It must not contain both "fix FBlocks".
Experimental set-up	- Tester 1 in master mode or slave mode; depends on DUT - Tester 2 in slave mode
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	During system scan, the DUT could start checking tester 1 before tester 2 responds. Each tester contain 2 FBlocks. One with normal behaviour (named "normal FBlock") that will change its InstID if commanded by the DUT and one FBlock which will not change its InstID if commanded (named "fix FBlock"). For the "fix FBlocks", both tester have to use same FBlockID and InstID. For the "normal FBlock" they have to use different FBlockIDs and different InstIDs.
Results	DUT ok (1): The DUT has passed the test. DUT ok (2): "fix FBlock" neither of Tester1 nor of Tester2 stored in CR of DUT DUT not ok (1): DUT fails to check Tester 2. DUT not ok (2): DUT fails to check Tester 1. DUT not ok (3): DUT fails to change FBlock-ID of Tester 1 or Tester 2. DUT not ok (4): DUT fails to store "normal FBlock" of tester into the CR. DUT not ok (5): DUT stores "fix FBlocks" of both tester into CR.

2.6.2-6

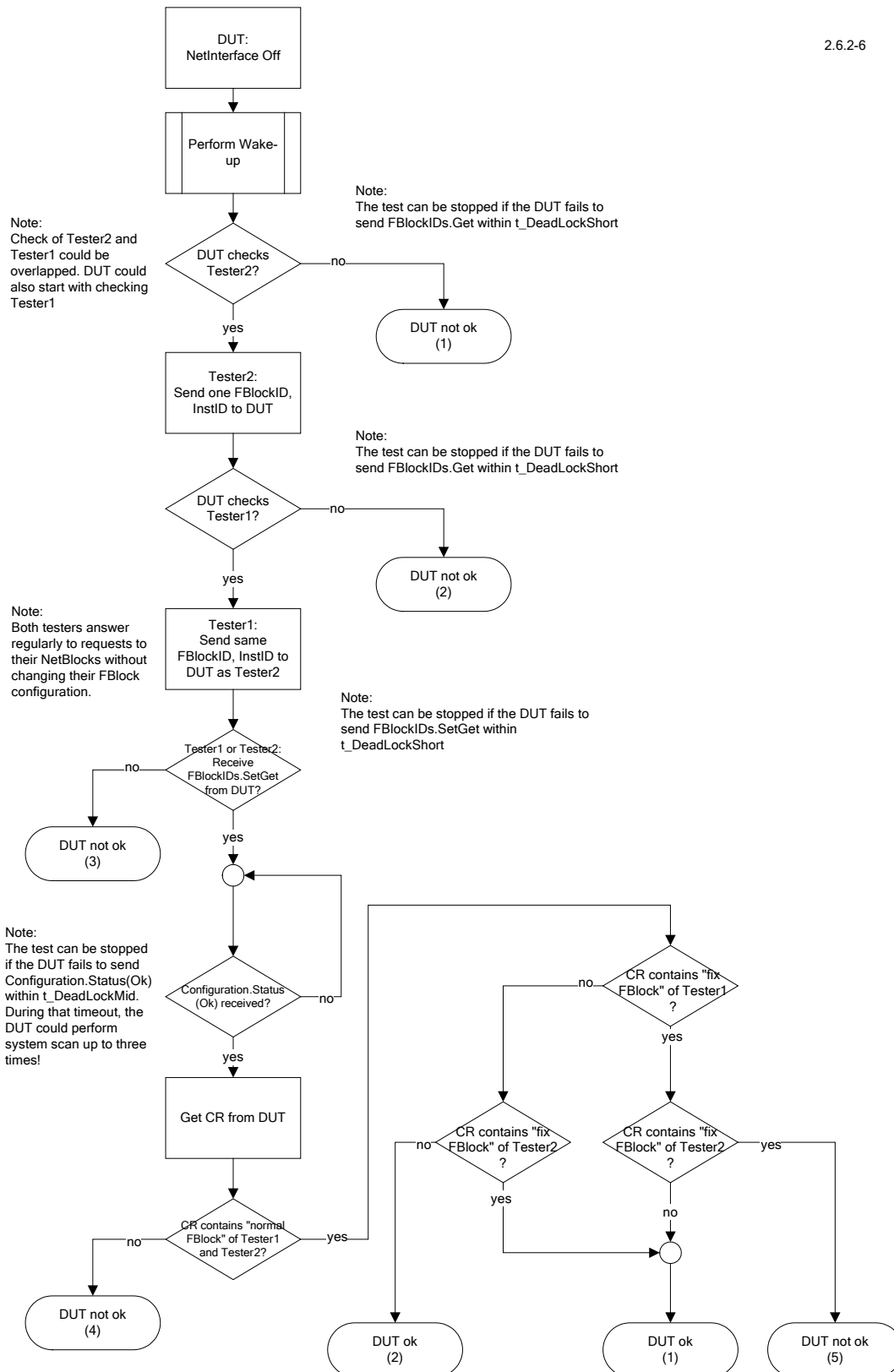


Figure 3-42 Flow of Double FBlock test.

3.9.1.10 Config(New) Order Test (2.6.2-7)

Name of test	Config(New) Order test 2.6.2-7
Reference to MOST Specification	3.1.3.2 System States
Value of Interest	Configuration.Status
Start Conditions	DUT in NormalOperation
Test description	Tester 2 triggers Config(NotOk) by sending FBlockIDs.Status to DUT, using sender address 0xFFFF. The DUT has to send Configuration.Status(NotOk). During the system scan, tester 1 has to indicate an additional FBlock and one missing FBlock (compared with initial system scan). Tester 2 must not respond to the system scan. The DUT has to broadcast Configuration.Status(Ok). Before broadcasting Configuration.Status(Ok), the DUT must not broadcast Configuration.Status(NewExt / New / Invalid).
Experimental set-up	- Tester 1 in slave mode - Tester 2 in slave mode
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	Tester 1 has to behave like normal MOST device
Results	DUT ok: The DUT has passed the test. DUT not ok (1): The DUT fails to send Configuration.Status(NotOk). DUT not ok (2): DUT sends Configuration.Status(NewExt) or Configuration.Status(Invalid) before sending Configuration.Status(Ok).

2.6.2-7

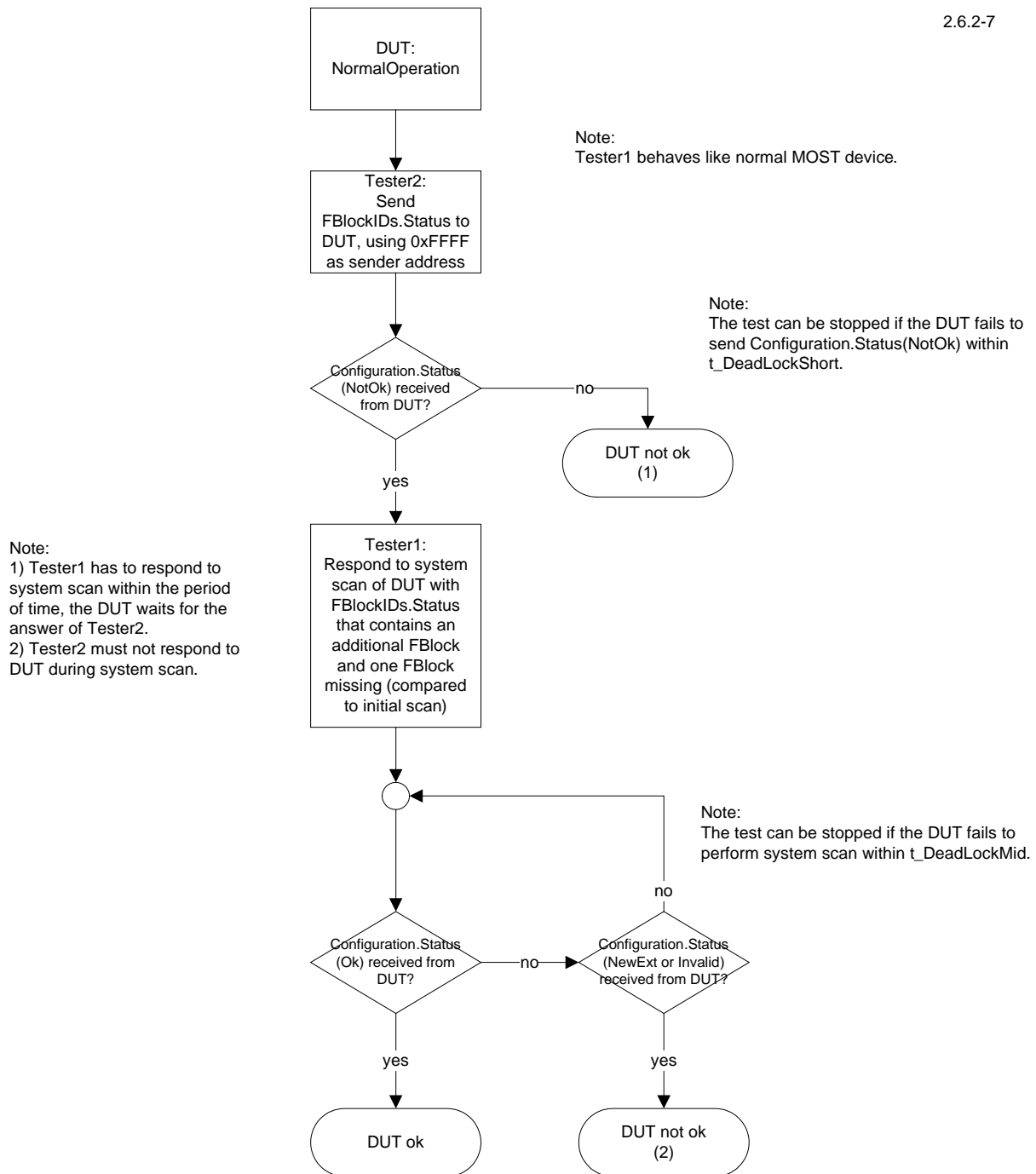


Figure 3-43 Flow of Config(New) Order test.

3.9.1.11 FBlock Status Change Detection Test (2.6.3-1)

Name of test	FBlock Status Change Detection test 2.6.3-1
Reference to MOST Specification	3.1.3.3.6.1 Disappearing Function Blocks in System State Ok
Value of Interest	CR Configuration.Status(Invalid)
Start Conditions	DUT in NormalOperation Ring closed
Test description	The Tester 1 changes the state of an preregistered FBlock from available to unavailable and indicates it to the DUT. The DUT has to broadcast "Configuration.Status(Invalid)" and delete it from the CR.
Experimental set-up	- Tester 1 in slave mode or in master mode (depends on DUT); valid address - Tester 2 in spy mode
Device type	NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT fails to broadcast "Configuration.Status(Invalid)". DUT not ok (2): DUT fails to delete unavailable FBlock from CR.

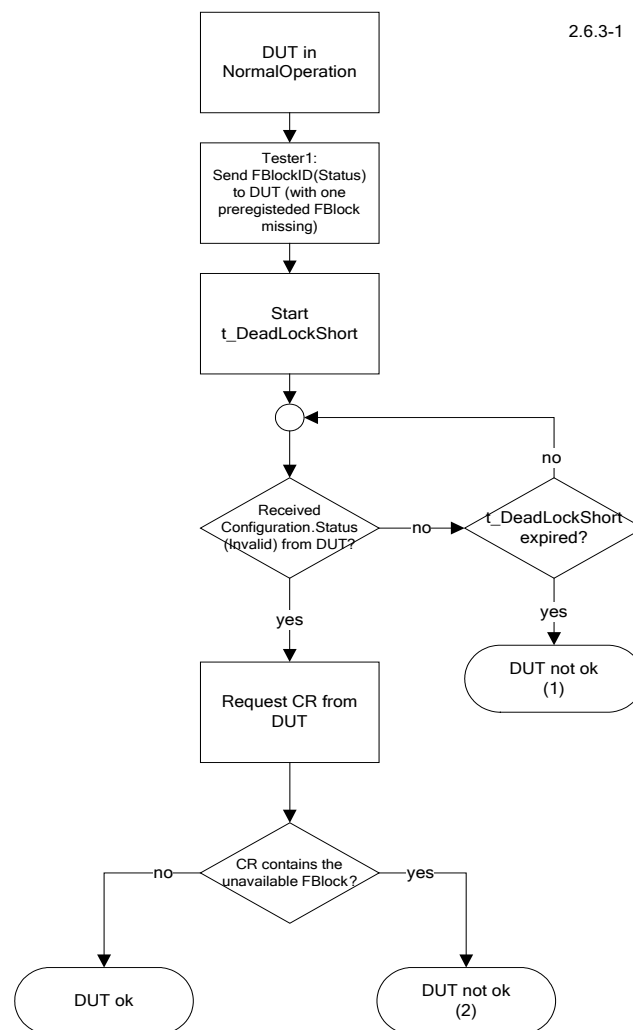


Figure 3-44 Flow of FBlock Status Change Detection test.

3.9.2 Configuration / System Configuration (Slave)

3.9.2.1 Address Initialization Test (2.6.4-1)

Name of test	Address Initialization test 2.6.4-1
Reference to MOST Specification	3.1.3.4.2 Specific Startup Behavior
Value of Interest	Address of DUT
Start Conditions	DUT in NormalOperation Ring closed
Test description	The address of the DUT will be set to a value, different from default address of DUT. The address of the DUT is stored. After ShutDown, disconnecting from power (for at least 1 minute) and Wake-up, the DUT has to use valid node address (in static or dynamic address range) or "uninitialized_node_address". Any other address is not allowed.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices except NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices without NWM
Note	
Results	DUT ok (1): The DUT has passed the test (uses valid node address). DUT ok (2): The DUT has passed the test (uses "uninitialized_node_address" before deriving new address from ring position). DUT not ok (1): DUT changes address between two operation cycles. DUT not ok (2): DUT neither uses "uninitialized_node_address" nor valid node address.

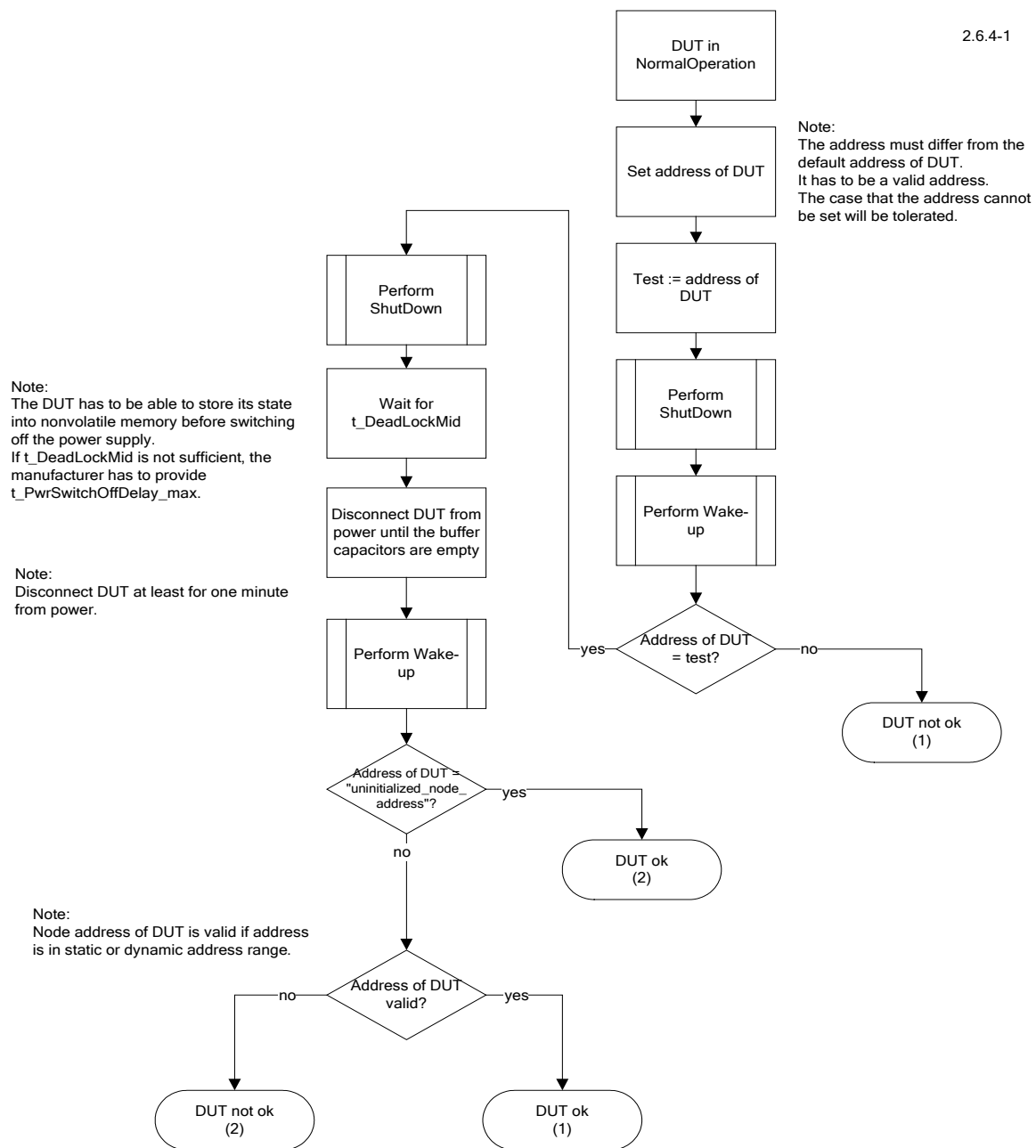


Figure 3-45 Flow of Address Initialization test.

3.9.2.2 NWM Address Storage Test (2.6.4-3)

Name of test	NWM Address Storage test 2.6.4-3
Reference to MOST Specification	3.1.3.4.2.3 Deriving the Logical Node Address of the Network Master
Value of Interest	NWM address, used by DUT CentralRegistry.Get
Start Conditions	DUT: NetInterface Off Ring closed
Test description	After the DUT has been woken up, the tester 1 broadcasts some Configuration.Status message. The DUT has to derive the address of the tester 1. The DUT is triggered to send any message to the NWM (tester 1) to check whether the DUT has derived the address correctly. After a shutdown, the address of tester 1 is changed and the same procedure is performed, again. The DUT has to respond to the new address of tester 1.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices that use the NWM during NormalOperation (e.g. each device which uses the CR) except the NWM. [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices without NWM (that are using the NWM/CR during NormalOperation)
Note	The test has to be performed three times, using Configuration.Status(Ok), Configuration.Status(NewExt) and Configuration.Status(Invalid). In case of "NewExt", an empty FBlock list has to be sent to ensure the DUT requests the CR from NWM. In case of "Invalid", the FBlock.InstID 0xC8.0x01 has to be used.
Results	DUT ok (1): The DUT has passed the test (CR not requested). DUT ok (2): The DUT has passed the test (CR not requested). DUT ok (3): The DUT has passed the test. DUT not ok: DUT stored address of NWM during Power-Off mode.

Note:
Between test loops, switch off MOST signal
without sending ShutDown.Start(Execute), i.e.
irregular shutdown.

2.6.4-3

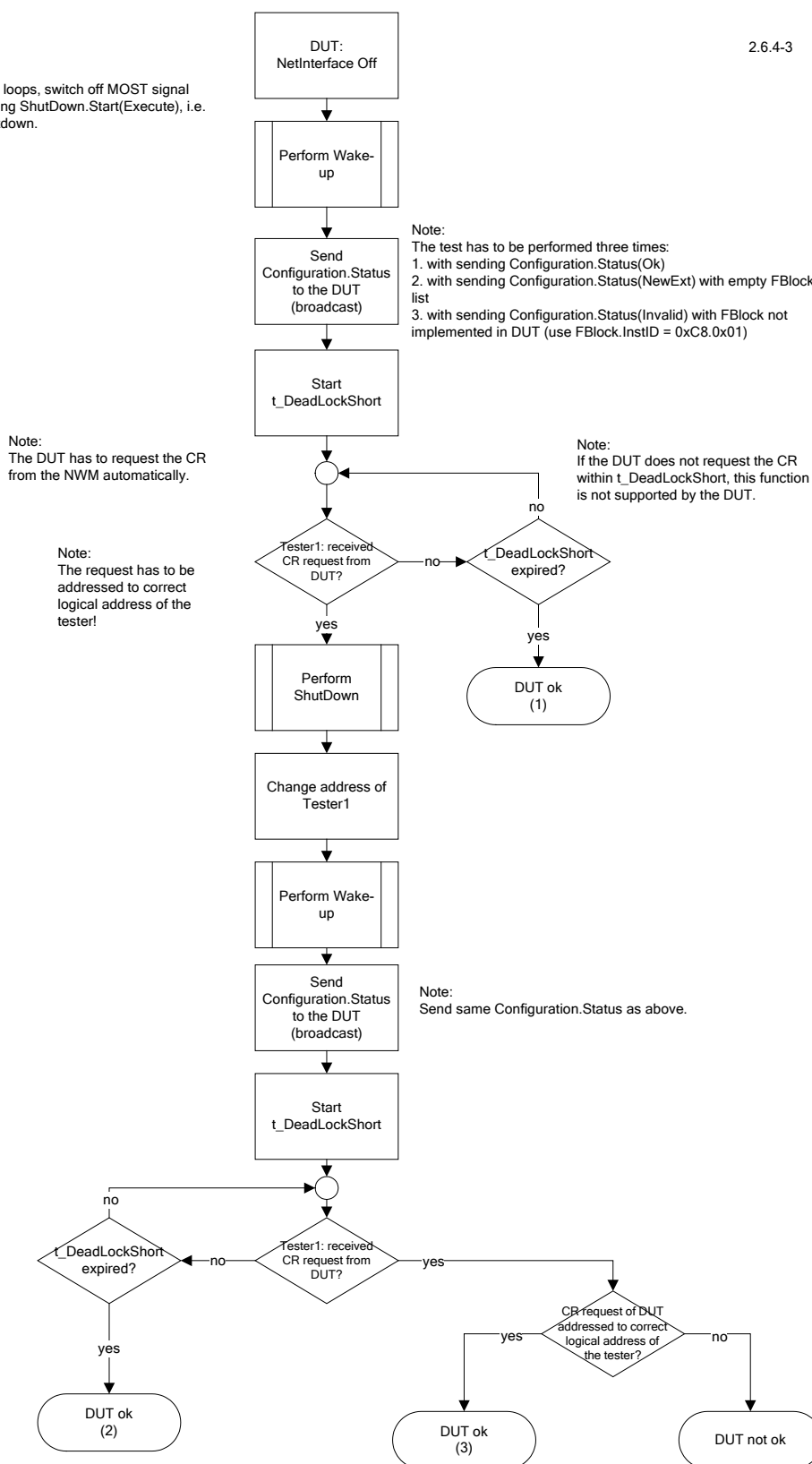


Figure 3-46 Flow of NWM Address Storage test.

3.9.2.3 Address Re-Initialization Test (2.6.4-4)

Name of test	Address Re-Initialization test 2.6.4-4
Reference to MOST Specification	3.1.3.4.2 Specific Startup Behavior
Value of Interest	Address of DUT CentralRegistry.Get
Start Conditions	DUT in NormalOperation Ring closed
Test description	During runtime, the position of the DUT is changed (by switching Tester 1 from spy mode into slave mode). Then, tester 2 sends "Configuration.Status(NotOK)". If the address of the DUT is in dynamic address range, it has to derive a new address from its ring position, else it has to continue using its old address. If the DUT supports DR, Tester 2 sends "Configuration.Status(ok)". The DUT has to send "CentralRegistry(Get)" to Tester 2 to rebuild its DR.
Experimental set-up	- Tester 1 in spy mode - Tester 2 in master mode
Device type	All devices except NWM and TM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices without NWM and TM
Note	
Results	DUT ok (1): The DUT has passed the test. DUT ok (2): The DUT has passed the test (does not support DR). DUT not ok (1): DUT (dyn. Address) fails to change address in case of receiving Config(NotOK). DUT not ok (2): DUT (dyn address) fails to derive new address from ring position. DUT not ok (3): DUT (not dyn. Address) fails to use previous address.. DUT not ok (4): DUT fails to delete / rebuild DR (if supported).

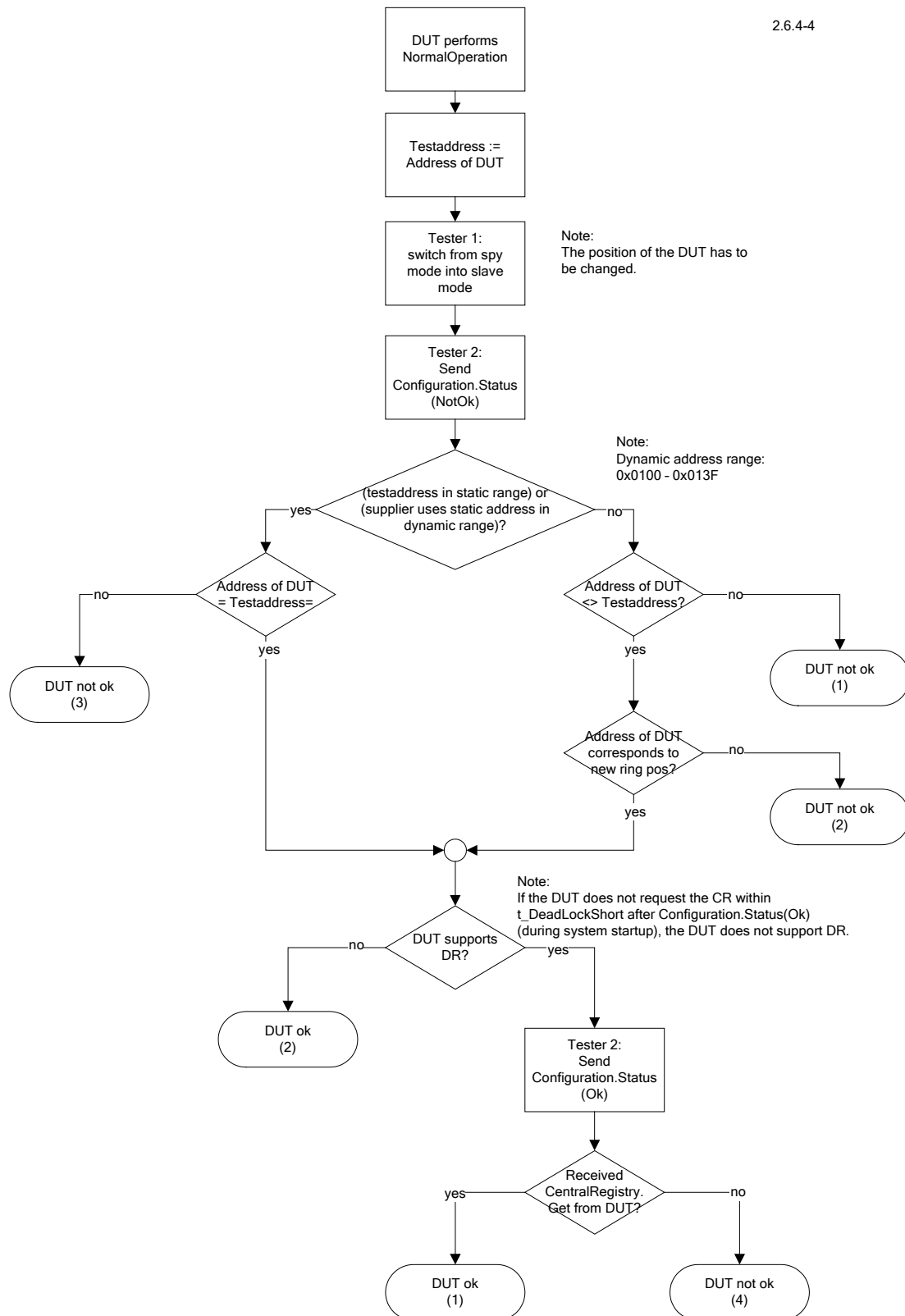


Figure 3-47 Flow of Address Re-Initialization test.

3.9.2.4 FBlock Response Time Test (2.6.4-8)

Name of test	FBlock Response Time test 2.6.4-8
Reference to MOST Specification	3.1.3.4.3.3 Responding to Configuration Requests by the Network Master
Value of Interest	t_Answer
Start Conditions	DUT: NetInterface Off Ring closed
Test description	After wake-up, the tester sends "FBlockIDs.Get" to the DUT. It has to respond within t_Answer_max.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices except NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices without NWM
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok: DUT fails to respond within t_Answer.

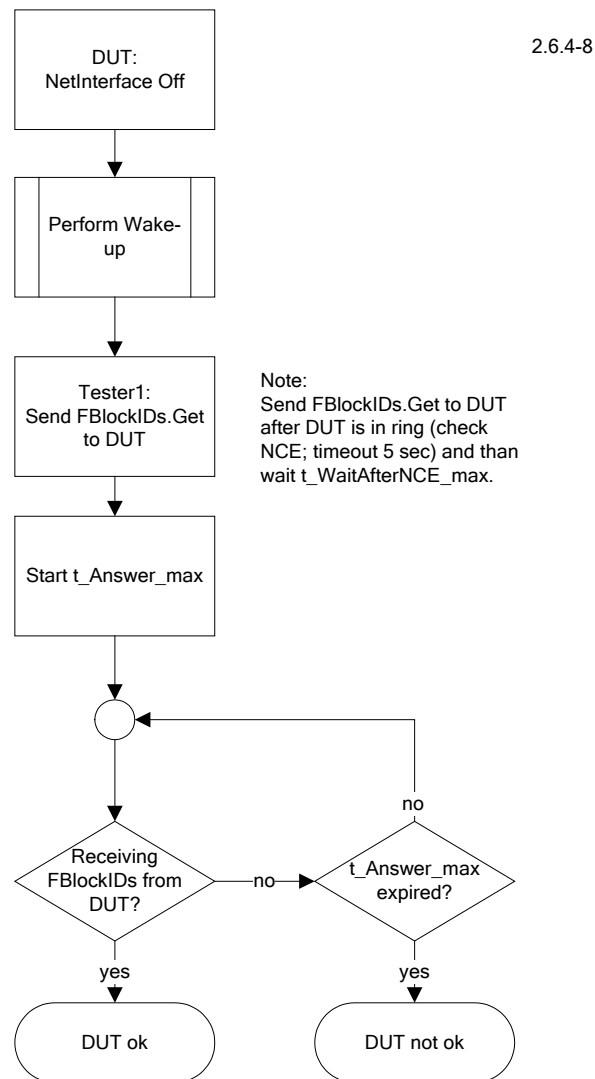


Figure 3-48 Flow of FBlock Response Time test.

3.9.2.5 Control Message Suppression Test (2.6.4-9)

Name of test	Control message suppression test 2.6.4-9
Reference to MOST Specification	3.1.3.4.3.2 Behavior in System State NotOk
Value of Interest	Control message from DUT
Start Conditions	DUT NetInterface Off
Test description	The tester performs a wake-up. It does not send any message to the DUT during start-up. The DUT must not send any control message to the tester.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT); NWM - Tester 2 in spy mode
Device type	All devices except NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices without NWM
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT sends control message without receiving FBlockIDs.Get

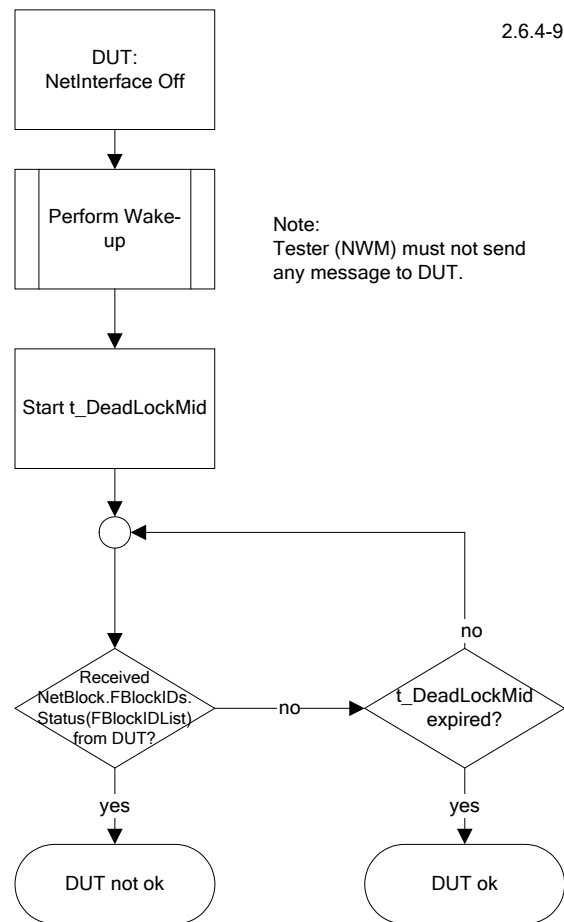


Figure 3-49 Flow of control message suppression test.

3.9.3 Configuration / System Configuration (General)

3.9.3.1 InstID Wildcard Test (2.6.4-10)

Name of test	InstID Wildcard Test 2.6.4-10
Reference to MOST Specification	2.2.3.3.6 InstID Wildcards
Value of Interest	FktIDs.Status InstID
Start Conditions	DUT in NormalOperation
Test description	<p>The tester sends FktIDs.Get to the DUT, using wildcard 0x00 for InstID. The DUT has to respond FktIDs.Status. The respond of the DUT must not contain any wildcard as InstID.</p> <p>The tester sends FktIDs.Get to the DUT, using wildcard 0xFF for InstID. The DUT has to respond FktIDs.Status of all implemented instances of the current FBlock. The respond of the DUT must not contain any wildcard as InstID.</p>
Experimental set-up	<ul style="list-style-type: none"> - Tester in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	<p>All devices</p> <ul style="list-style-type: none"> [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices
Note	<p>The test has to be performed with every single registered FBlock of the DUT, except NetBlock, FBlock ET and PM.</p> <p>All implemented instances of an FBlock of the DUT can be detected by the tester by sending FBlockIDs.Get to the DUT.</p>
Results	<p>DUT ok: The DUT has passed the test.</p> <p>DUT not ok (1): The DUT responds with Error "InstID not available"</p> <p>DUT not ok (2): The DUT uses wildcard for responding FktIDs.Status.</p> <p>DUT not ok (3): DUT fails to respond with FktIDs.Status of all implemented instances of the current FBlock.</p> <p>DUT not ok (4): The DUT uses wildcard for responding FktIDs.Status.</p>

2.6.4-10

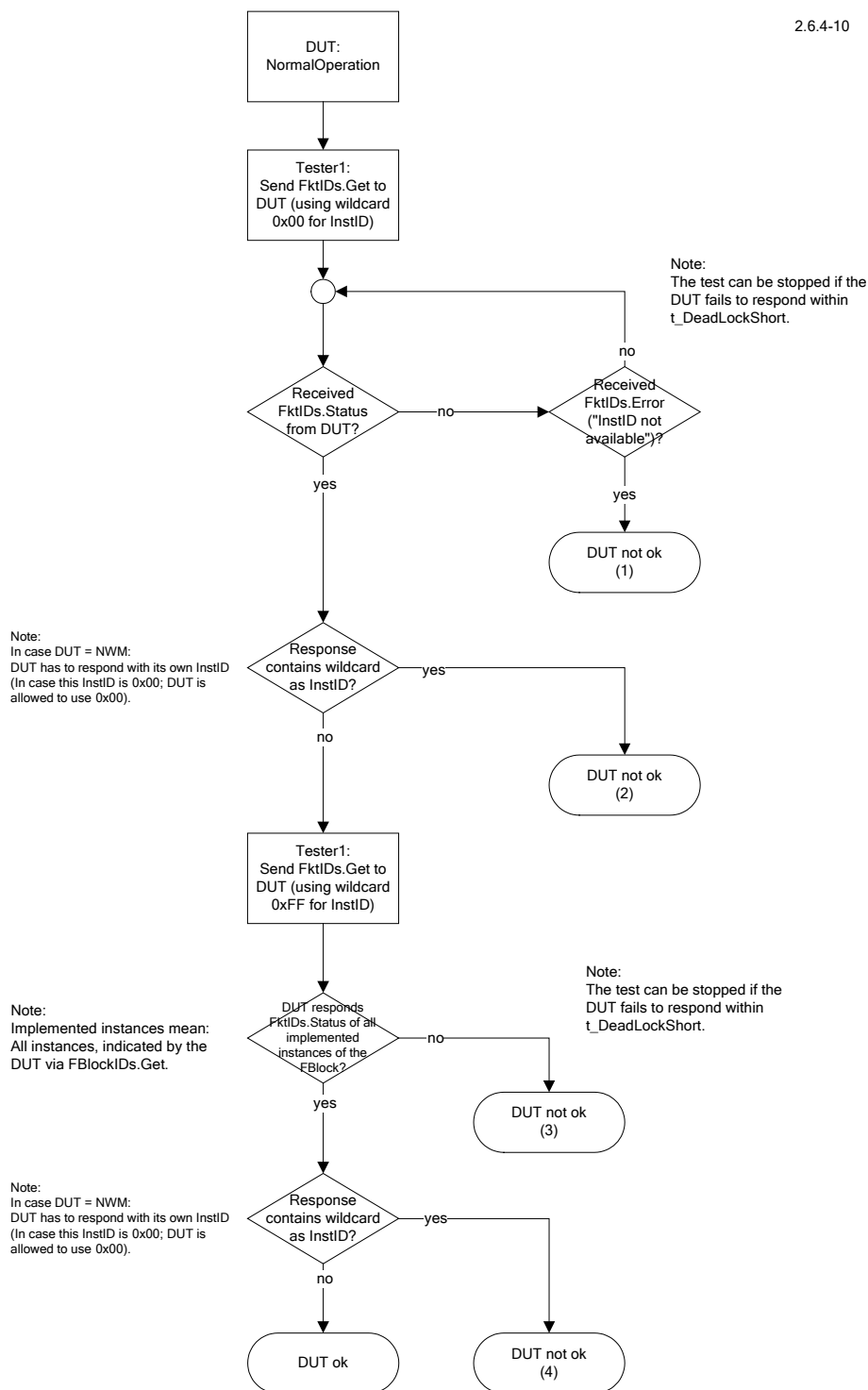


Figure 3-50 Flow of InstID Wildcard test.

3.10 Node Addressing

3.10.1.1 Node Addressing Test (2.7-1)

Name of test	Node Addressing test 2.7-1
Reference to MOST Specification	3.2.2 Addressing
Value of Interest	Node Position Address of DUT Groupcast Address of DUT Broadcast Address of DUT
Start Conditions	DUT in NormalOperation Ring closed
Test description	At the beginning of the test, the ring is shut down and the position of the DUT is changed (Tester 1: spy mode → slave mode). At the next wake-up, the DUT has to derive the new Node Position Address from its new ring position. The Node Position Address, the Logical Node Address, the Group Address and the Broadcast Address (blocking and unblocking) of the DUT are tested by sending a message to the corresponding address and checking the respond.
Experimental set-up	- Tester 1 in spy mode; it must not respond to group cast messages. - Tester 2 in master mode or slave mode (depends on DUT).
Device type	All devices except NWM [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices without NWM.
Note	
Results	DUT ok (1): The DUT has passed the test. DUT not ok (1): DUT fails to respond to "Node position addressing". DUT not ok (2): DUT fails to respond to "Logical Node addressing". DUT not ok (3): DUT fails to report the correct group address (differs from manufacturer information). DUT not ok (4): DUT fails to respond to "group addressing". DUT not ok (5): DUT fails to respond to "broadcast addressing" (blocking broadcast). DUT not ok (6): DUT fails to respond to "broadcast addressing" (unblocking broadcast).

2.7-1

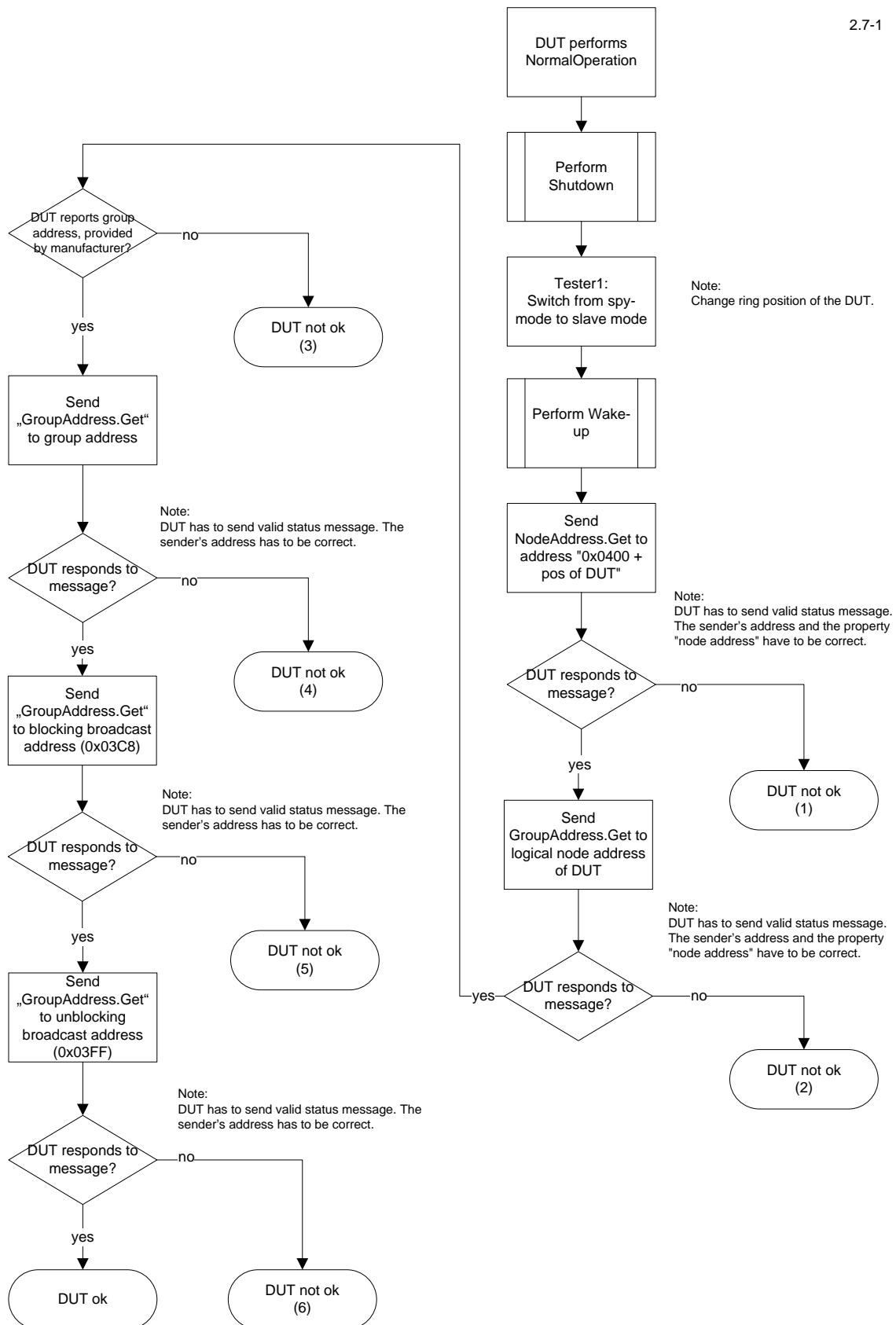


Figure 3-51 Flow of Node Addressing test.

3.10.1.2 Broadcast Error Test (2.8.4-9)

Name of test	Broadcast error test 2.8.4-9
Reference to MOST Specification	2.2.3.5.1 Error
Value of Interest	error respond of broadcast message
Start Conditions	DUT in NormalOperation
Test description	Tester 1 produces an error within the DUT by broadcasting message "ET.InstID(0x00).FktID(0x3c8).SetGet" to the DUT. The DUT must not return any error. The not implemented function will be addressed normally and DUT has to respond with error.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices
Note	
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT responds error on broadcast messages DUT not ok (2): DUT fails to respond with error if not implemented function is addressed normally

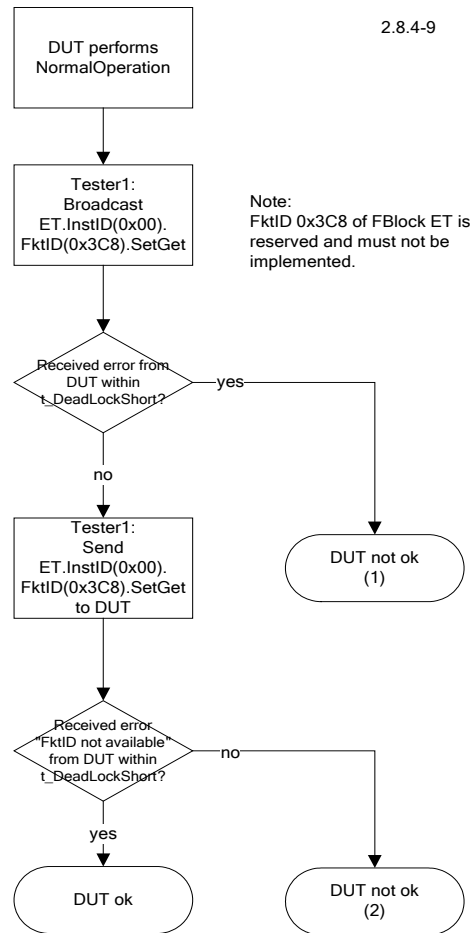


Figure 3-52 Flow of Broadcast error test.

3.11 Other Functions

3.11.1 Notification Matrix

3.11.1.1 Notification Matrix Storage Test (NWM) (2.8.3-1a)

Name of test	Notification Matrix Storage test (NWM) 2.8.3-1a
Reference to MOST Specification	2.2.5 Handling Message Notification
Value of Interest	Notification Matrix t_Property
Start Conditions	DUT in NormalOperation Ring closed
Test description	<p>The tester enters itself into the Notification Matrix of the DUT by using Notification.Set(All). If the DUT supports notification, it must not generate a notification error. If the tester triggers a "Configuration.Status(NewExt)" (by sending new FBlockID lists with new FBlockIDs to the DUT), the DUT must not delete the entries from the Notification Matrix. Tester 2 leaves the ring by switching from test mode into spy mode. The DUT has to send "Config(Invalid)" and its Notification Matrix must not be empty. Tester 2 changes its address to "uninitialized_node_address" and enters the ring, again (test mode). The DUT has to broadcast "Config(NotOK)" and its Notification Matrix must not contain any entry of the tester. Tester 1 writes some entry to the Notification Matrix of the DUT. It triggers Config(ok). If DUT sends Config.Status(ok), the Notification Matrix will be checked again. It must contain all entries, made by the tester. Tester performs a ShutDown. At the next wake-up, the Notification Matrix of the DUT must not contain any entry of the tester.</p>
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in slave mode or master mode (depends on DUT) - Tester 2 in test mode
Device type	<p>NWM</p> <ul style="list-style-type: none"> [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing NWM
Note	<p>The test has to be performed with every FBlock and function of the DUT, registered in the CR except FBlock ET and FBlock PM. All FBlocks have to be tested within one testloop (no separate testloop for every FBlock) to increase stress of DUT. At the beginning of the test, the notification of the DUT must not contain any entry of the tester. Test is based on generation of NCEs so the DUT has to check the SystemConfiguration on each NCE and the tester 2 has to set the FBlocks and address accordingly.</p>
Results	<p>DUT ok (1): The DUT has passed the test. It does not support notification. DUT ok (2): The DUT has passed the test. DUT not ok (1): DUT fails to store any entry into its notification matrix DUT not ok (2): DUT deletes Notification Matrix in case of Config(NewExt). DUT not ok (3): DUT deletes Notification Matrix in case of Config(Invalid). DUT not ok (4): DUT fails to delete the Notification Matrix in case of Config(NotOK). DUT not ok (5): DUT deletes Notification Matrix in case of Config(Ok) during SystemState(Ok). DUT not ok (6): DUT fails to delete the Notification Matrix after NetOn.</p>

2.8.3-1a

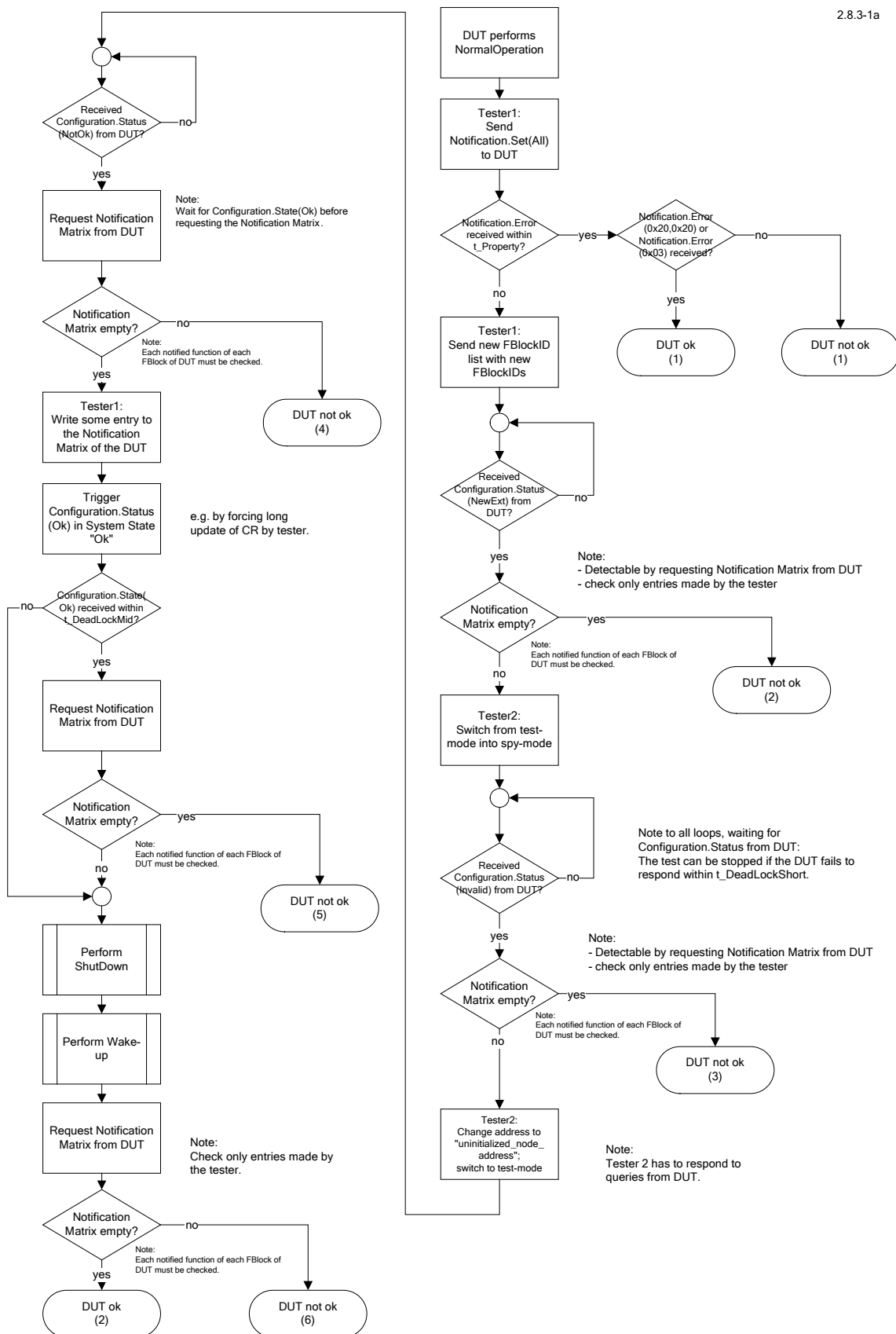


Figure 3-53 Flow of Notification Matrix Storage test (NWM).

3.11.1.2 Notification Matrix Storage Test (Slave) (2.8.3-1b)

Name of test	Notification Matrix Storage test (Slave) 2.8.3-1b
Reference to MOST Specification	2.2.5 Handling Message Notification
Value of Interest	Notification Matrix t_Property
Start Conditions	DUT in NormalOperation Ring closed
Test description	<p>The tester enters itself into the Notification Matrix of the DUT by using Notification.Set(All). If the DUT supports notification, it must not generate a notification error. If the tester sends "Config(Ok)", the DUT must not delete the entries from the Notification Matrix. If the tester sends "Config(NewExt)", the DUT must not delete the entries from the Notification Matrix. If the tester sends "Config(Invalid)", the DUT must not delete the entries from the Notification Matrix. If the tester sends "Config(NotOK)", the DUT has to delete the entries of the tester from the Notification Matrix. Tester 1 writes some entry to the Notification Matrix of the DUT and performs a ShutDown. At the next wake-up, the Notification Matrix of the DUT must not contain any entry of the tester.</p>
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	<p>All devices except NWM</p> <p>[x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY</p>
Multi Node Device	All devices without NWM
Note	<p>The test has to be performed with every FBlock and function of the DUT, registered in the CR except FBlock ET and FBlock PM. All FBlocks have to be tested within one testloop (no separate testloop for every FBlock) to increase stress of DUT.</p> <p>At the beginning of the test, the notification of the DUT must not contain any entry of the tester.</p>
Results	<p>DUT ok (1): The DUT has passed the test. It does not support notification. DUT ok (2): The DUT has passed the test. DUT not ok (1): DUT fails to store any entry into its notification matrix DUT not ok (2): DUT deletes Notification Matrix if Config(Ok) has been received. DUT not ok (3): DUT deletes Notification Matrix if Config(NewExt) has been received. DUT not ok (4): DUT deletes Notification Matrix if Config(Invalid) has been received. DUT not ok (5): DUT fails to delete the Notification Matrix if Config(NotOK) has been received. DUT not ok (6): DUT fails to delete the Notification Matrix after NetOn.</p>

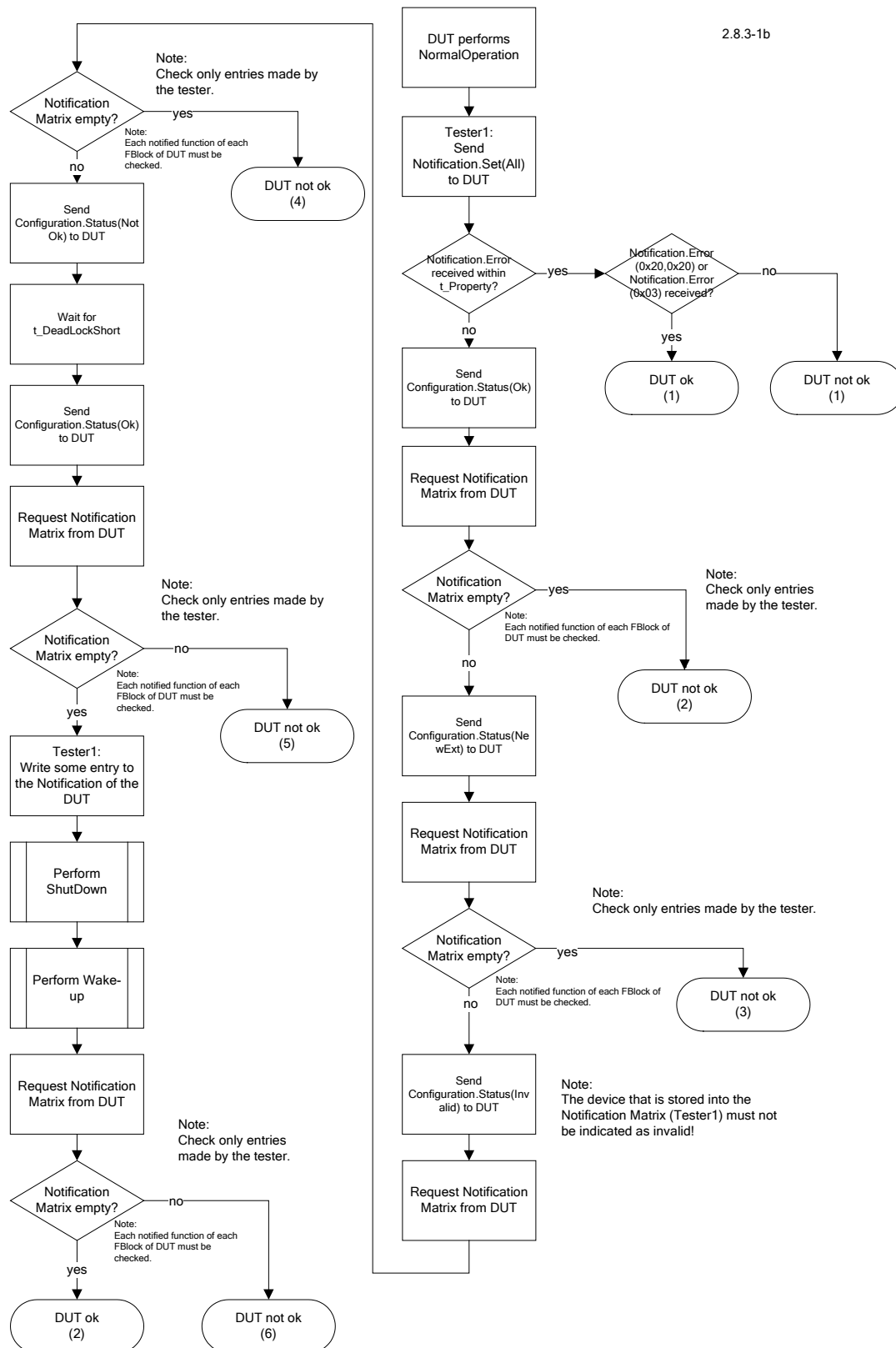


Figure 3-54 Flow of Notification Matrix Storage test (Slave).

3.11.1.3 NotificationCheck Test (2.8.3-2)

Name of test	NotificationCheck test 2.8.3-2
Reference to MOST Specification	2.2.3.4 FktID
Value of Interest	NotificationMatrix
Start Conditions	DUT in NormalOperation
Test description	The tester1 requests the FktID list of DUT to check whether NotificationCheck is supported. If the DUT supports NotificationCheck, the tester1 enters itself into all properties of the current FBlock to the DUT that support notification. Then the tester sends Notification.Check to the DUT. The DUT has to respond a list that contains all functions supporting notification.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in slave mode
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices
Note	The test has to be performed with every FBlock of the DUT, registered in the CR except PM.
Results	DUT ok (1): The DUT has passed the test. DUT ok (2): The DUT has passed the test (does not support NotificationCheck). DUT not ok (1): The DUT fails to respond with list, containing all functions that support notification. DUT not ok (2): Responded list from DUT (for tester2) not empty. DUT not ok (3): DUT fails to respond to FktIDs.Get in time.

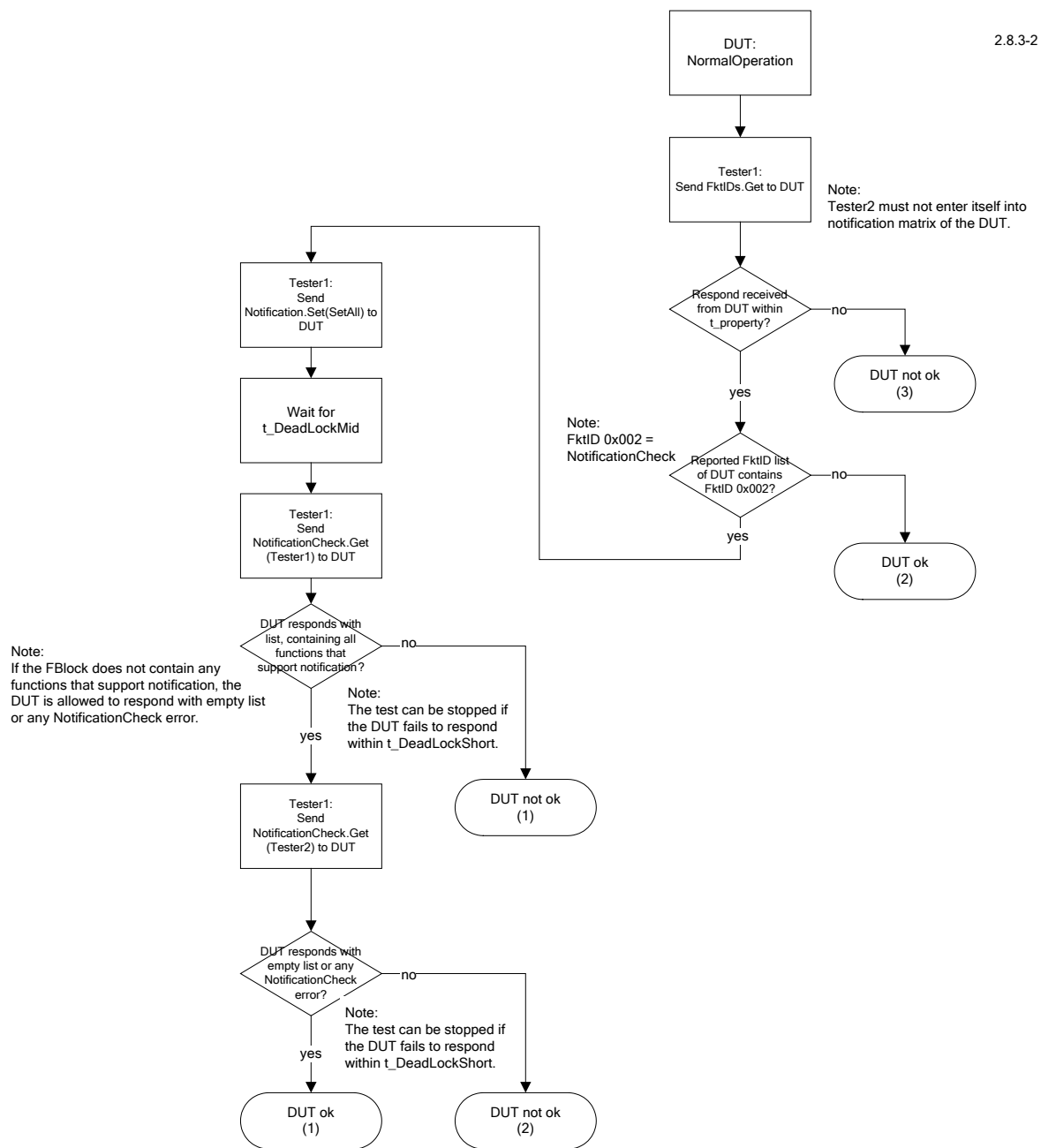


Figure 3-55 Flow of NotificationCheck test.

3.11.1.4 Notification Matrix Double Entry Test (2.8.3-7)

Name of test	Notification Matrix Double Entry test 2.8.3-7
Reference to MOST Specification	2.2.5 Handling Message Notification
Value of Interest	Notification Matrix t_WaitforProperty
Start Conditions	DUT in NormalOperation Ring closed
Test description	To get all functions that could be used for notification, a "Notification.Set(All)" is sent to the DUT. If the DUT supports notification, it must not generate a notification error. The tester sends "Notification.Set(All)" to the DUT, again (try to generate double entries). The DUT has to acknowledge the double entries normally but it must not store it into the Notification Matrix. If the tester deletes itself from the Notification Matrix, the Notification Matrix must not contain any entry of the tester.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices
Note	The test has to be performed with every registered FBlock and function of the DUT, registered in the CR except FBlock ET and FBlock PM. All FBlocks have to be tested within one testloop (no separate testloop for every FBlock) to increase stress of DUT. To check whether the DUT supports notification, the FBlock ET could be used to check the size of the notification matrix. At the beginning of the test, the notification matrix of the DUT must not contain any entry of the tester.
Results	DUT ok (1): The DUT has passed the test. It does not support notification. DUT ok (2): The DUT has passed the test. DUT not ok (1): DUT fails to store any entry into its notification matrix. DUT not ok (2): DUT fails to acknowledge entries to the Notification Matrix within t_WaitforProperty. DUT not ok (3): DUT stores double entries into the Notification Matrix. DUT not ok (4): DUT fails to delete devices from the Notification Matrix.

2.8.3-7

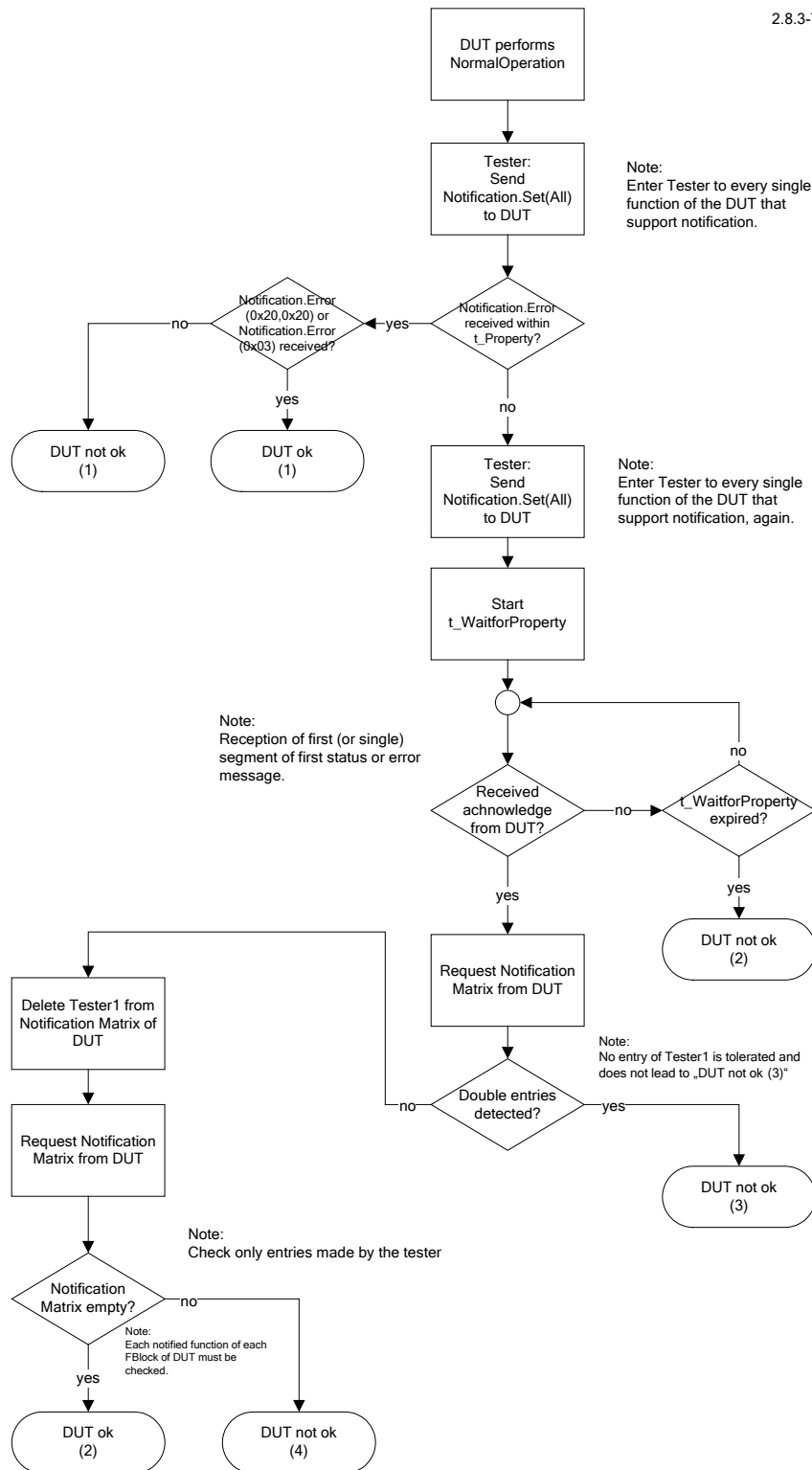


Figure 3-56 Flow of Notification Matrix Double Entry test.

3.11.1.5 Notification Error Test (2.8.3-10)

Name of test	Notification Error test 2.8.3-10
Reference to MOST Specification	2.2.5 Handling Message Notification
Value of Interest	Notification Matrix Notification.Error
Start Conditions	DUT in NormalOperation Ring closed
Test description	<p>The tester requests a list of all function-IDs of the DUT (by means of "FktIDs.Get"). It tries to register itself to a function of the DUT that does not exist. The DUT has to respond with a "Notification Error" if the FBlock supports notification.</p> <p>The tester sends "Notification.Set(All)" to the DUT to get a list of all functions that support notification. It tries to register to a function that does not support notification. The DUT has to respond with an error.</p> <p>The tester tries to read out the "notification state" of a function that does not support notification. The DUT has to respond with an error.</p> <p>The notification matrix of the DUT will be cleared by means of "Notification.Set(ClearAll)" and the tester tries to delete an existing, not notified function from the notification matrix of the DUT. The DUT must not return any error.</p>
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	<p>All devices</p> <p>[x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY</p>
Multi Node Device	All devices
Note	<p>The test has to be performed with every FBlock of the DUT, registered in the CR except FBlock ET and FBlock PM.</p> <p>All FBlocks have to be tested within one testloop (no separate testloop for every FBlock) to increase stress of DUT.</p>
Results	<p>DUT ok (1): The DUT has passed the test.</p> <p>DUT ok (2): The DUT has passed the test (FB does not support notification).</p> <p>DUT not ok (1): DUT fails to send any error if function does not exist.</p> <p>DUT not ok (2): DUT fails to send "Notification Error" if function does not support notification.</p> <p>DUT not ok (3): DUT fails to send "Notification Error(0x07,0x1)" if requested function does not support notification.</p> <p>DUT not ok (4): DUT responds with error in case of deletion of a not notified function.</p>

2.8.3-10

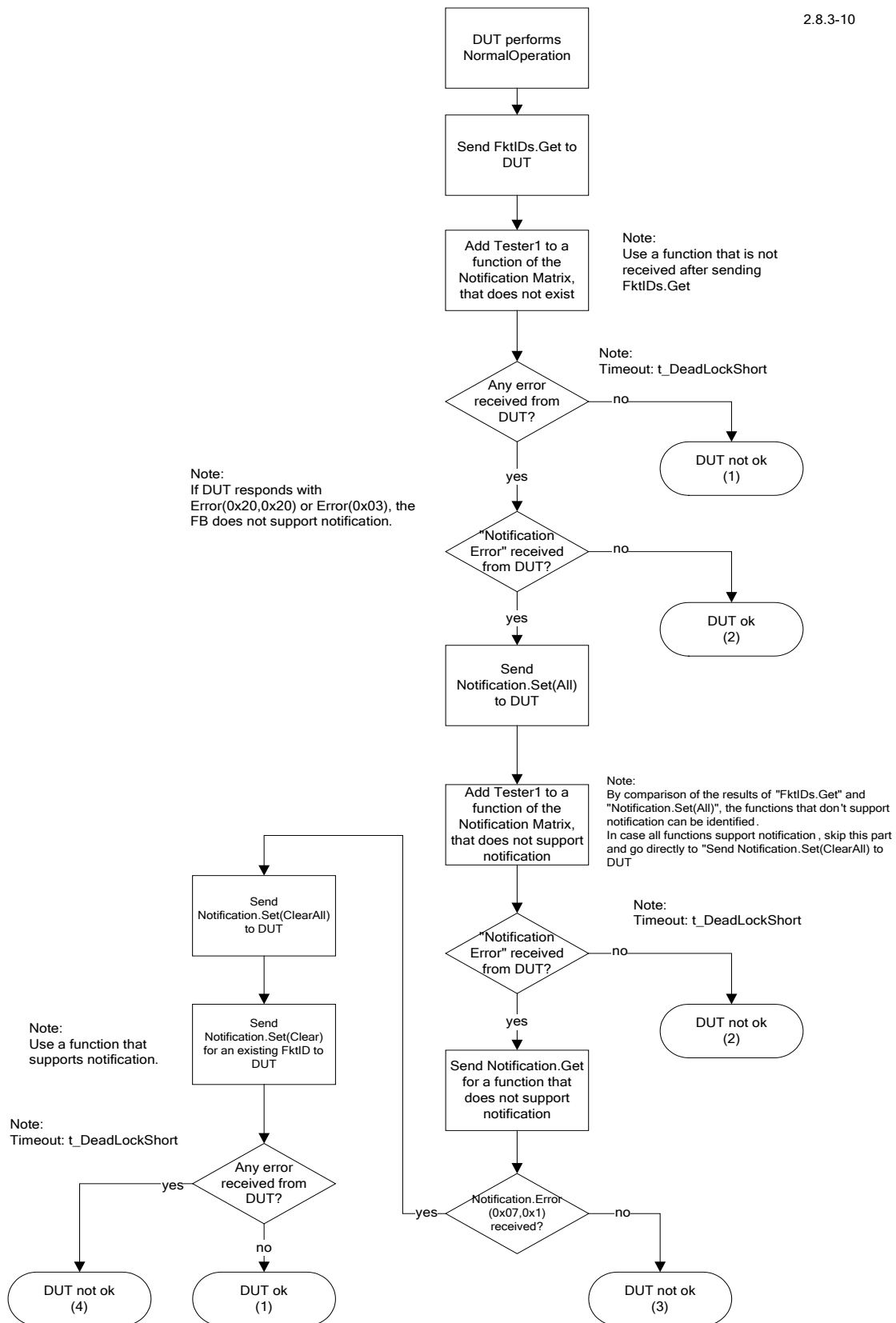


Figure 3-57 Flow of Notification Error test.

3.11.2 Message Segmentation

3.11.2.1 Segmented Message Rejection Test (2.8.4-1)

Name of test	Segmented Message Rejection test 2.8.4-1
Reference to MOST Specification	2.2.3.5.1 Error
Value of Interest	Segmentation Error t_WaitForNextSegment
Start Conditions	DUT in NormalOperation Ring closed
Test description	The tester sends the first part of a segmented message ("ET.EchoMessage.StartResult()") to the DUT. If the DUT does not support segmented messages, it has to respond with a "segmentation error(06)". If the DUT supports segmented messages, it has to respond with "Segmentation Error(05)".
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices
Note	Timer Values for t_WaitForNextSegment are: min. 4950 typ. 5000 max. 10150 ms.
Results	DUT ok (1): The DUT has passed the test (DUT does not support segmented messages). DUT ok (2): The DUT has passed the test (DUT supports segmented messages). DUT not ok(1): DUT fails to send "Segmentation Error" in time. DUT not ok(2): DUT sends "Segmentation Error" too early.

2.8.4-1

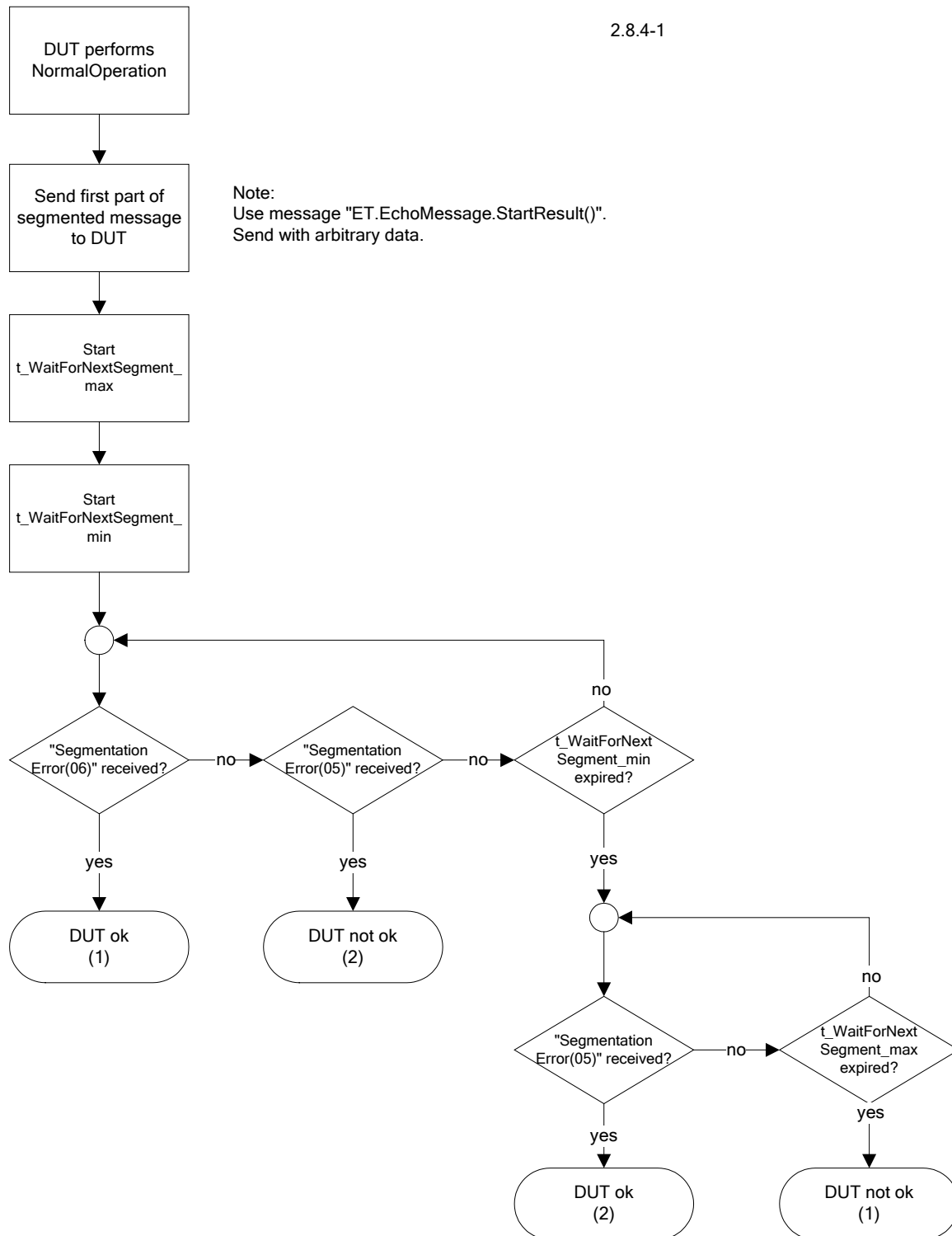


Figure 3-58 Flow of Segmented Message Rejection test.

3.11.2.2 Segmented Message Sending Test (2.8.4-2)

Name of test	Segmented Message Sending test 2.8.4-2
Reference to MOST Specification	3.2.5.2 Application Message Service (AMS)
Value of Interest	Segmented message from DUT
Start Conditions	DUT in NormalOperation Ring closed
Test description	The tester triggers the DUT (via "FBlock ET"; SendMessage) to send a segmented message to the tester. The DUT has to send the message correctly: <ul style="list-style-type: none"> - Sequence of message has to be correct - The whole message has to be sent right in time (t_WaitForNextSegment_max x number of segments) - The telegram types have to be correct (for example 1,2,2,2,3)
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices that support message segmentation. [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices with at least one message segmentation supporting node
Note	Possibly, some devices are able to send segmented messages but not able to receive.
Results	DUT ok: The DUT has passed the test. DUT not ok: The DUT fails to send a segmented message correctly.

Note:
MOST50 ePHY:
Trigger the segmented message via FBlock ET (SendMessage).
Depending on the buffer size given by FB ET the message consists of 2, 3 or 4 segments:
"size = 13 .. 22": 2 segments
"size = 23 .. 33": 3 segments
"size > 33" : 4 segments
In case the MessageBufSize is dynamic (indicated by value 0) the length to be sent by the DUT is 44 bytes (4 segments)

MOST150 oPHY:
Depending on the buffer size given by FB ET the message consists of 2, 3 or 4 segments:
In case the MessageBufSize is dynamic (indicated by value 0) the DUT has to send at least (LAMSmax + 1) bytes and at maximum 4 segments.

Note:
Correctly reception means:
- Sequence of messages has to be correct
- All segments has to be received before the Garbage Collection gets activated (max. t_WaitForNextSegment_max between two segments)
- Telegram types has to be correct (for example 1,2,2,2,3)

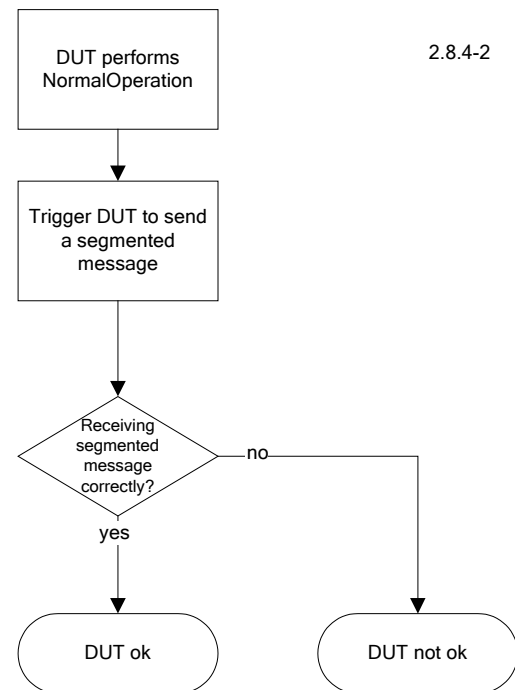


Figure 3-59 Flow of Segmented Message Sending test.

3.11.2.3 Message Segmentation Error Test (2.8.4-3)

Name of test	Message Segmentation Error test 2.8.4-3
Reference to MOST Specification	2.2.3.5.1 Error
Value of Interest	Segmentation Error t_Property
Start Conditions	DUT in NormalOperation Ring closed
Test description	<p>The tester sends a second part of a segmented message ("ET.EchoMessage.StartResult()") to the DUT without sending the first part. The DUT has to respond with "Segmentation Error(01)".</p> <p>The tester sends the first and the third part of a segmented message without sending the second part. The DUT has to respond with "Segmentation Error(03)".</p> <p>The tester sends the first part of a segmented message twice. The DUT has to respond with "Segmentation Error(07)".</p>
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	<p>All devices that support message segmentation.</p> <p>[x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY</p>
Multi Node Device	All devices with at least one message segmentation supporting node
Note	
Results	<p>DUT ok: The DUT has passed the test.</p> <p>DUT not ok (1): DUT fails to detect/indicate missing first part of segm. Message.</p> <p>DUT not ok (2): DUT fails to detect/indicate wrong order of parts of segm. Message.</p> <p>DUT not ok (3): DUT fails to detect/indicate parts of segm. Message that are sent twice.</p>



Figure 3-60 Flow of Message Segmentation Error test.

3.11.2.4 Message Segmentation Buffer Test (2.8.4-7)

Name of test	Message Segmentation Buffer test 2.8.4-7
Reference to MOST Specification	3.2.5.2 Application Message Service (AMS)
Value of Interest	Segmentation Error
Start Conditions	DUT in NormalOperation Ring closed
Test description	<p>If DUT does not support dynamic buffer size, the tester sends a segmented message ("ET.EchoMessage(0x208).Data" with correct InstID for the DUT) to the DUT that exceeds the supported buffer size. The DUT has to respond with "Segmentation Error(0x02)".</p> <p>The tester sends a segmented message with more than 256 segments (but smaller than the supported buffer size). The DUT must not return any segmentation error.</p> <p>If DUT supports dynamic buffer size, the first part of the test (trigger of "Segmentation Error (0x02)") will be skipped.</p>
Experimental set-up	<ul style="list-style-type: none"> - Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	<p>All devices that support message segmentation.</p> <ul style="list-style-type: none"> [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices with at least one message segmentation supporting node
Note	The tester shall not use TelID 0x4
Results	<p>DUT ok: The DUT has passed the test.</p> <p>DUT not ok (1): DUT fails to return a "buffer size overflow" error.</p> <p>DUT not ok (2): DUT fails to handle more than 256 segments per message.</p>

2.8.4-7

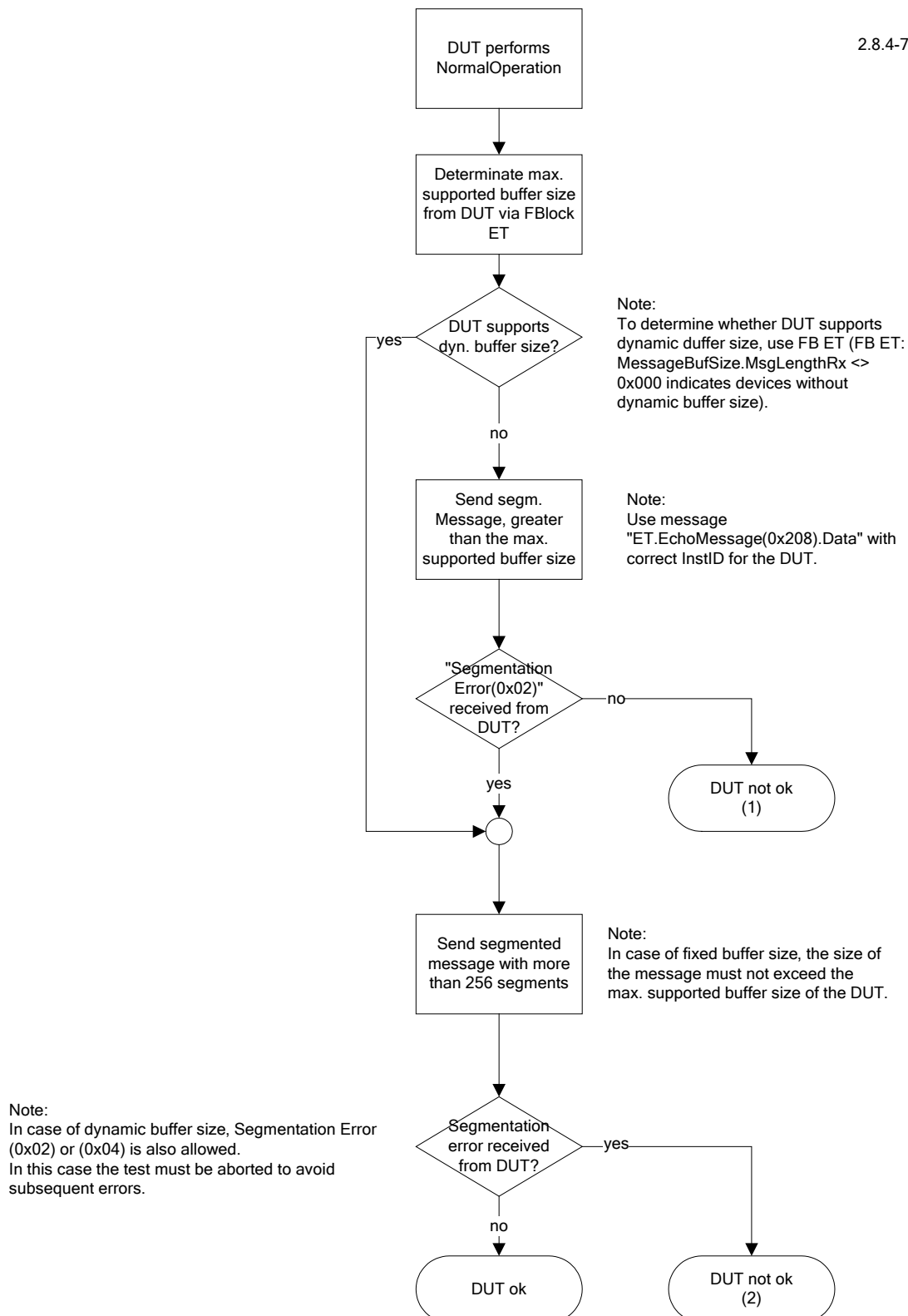


Figure 3-61 Flow of Message Segmentation Buffer test.

3.11.2.5 Parallel Segmented Message Reception Test (2.8.4-8)

Name of test	Parallel Segmented Message Reception test 2.8.4-8
Reference to MOST Specification	3.2.5.2 Application Message Service (AMS)
Value of Interest	Pool overflow error
Start Conditions	DUT in NormalOperation Ring closed
Test description	The tester determines the max. number of simultaneous segmented messages that are supported by the DUT (e.g. by means of the "FBlock ET"). The tester sends as much first parts of segmented messages ("ET.InstID.<FunctionID>.Get") as supported by the DUT (every message is sent by a unique value of <i>FunctionID</i>). The DUT must not return any error. If the tester sends one more message, the DUT has to return a "pool overflow" error ErrorCode 0x0C, ErrorInfo 0x04). The whole test has to be finished before the garbage collection is active.
Experimental set-up	- Tester 1 in master mode or slave mode (depends on DUT) - Tester 2 in spy mode
Device type	All devices that support message segmentation [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices with at least one message segmentation supporting node
Note	Test only applicable for devices without dynamic parallel message buffers (FBlock ET: MessageBufSize.BufferCountRx <> 0x000)
Results	DUT ok: The DUT has passed the test. DUT not ok (1): DUT fails to handle enough segmented message simultaneously. DUT not ok (2): DUT fails to send "pool overflow" error (ErrorInfo 0x04).

Note:
The whole test has to be performed, before
the Garbage Collection of the DUT gets
active.

2.8.4-8

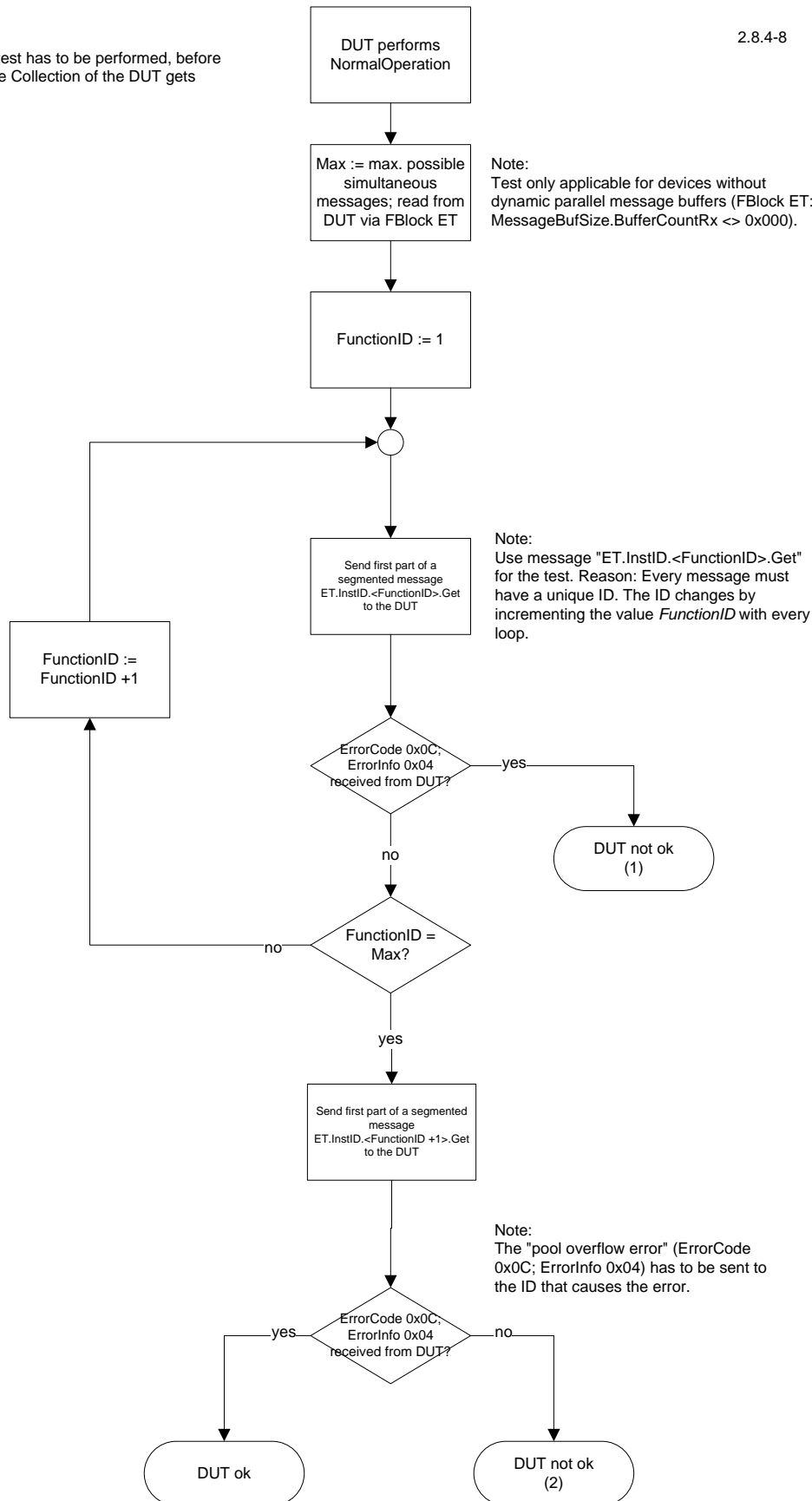


Figure 3-62 Flow of Parallel Segmented Message Reception test.

3.12 Source / Sink Identification

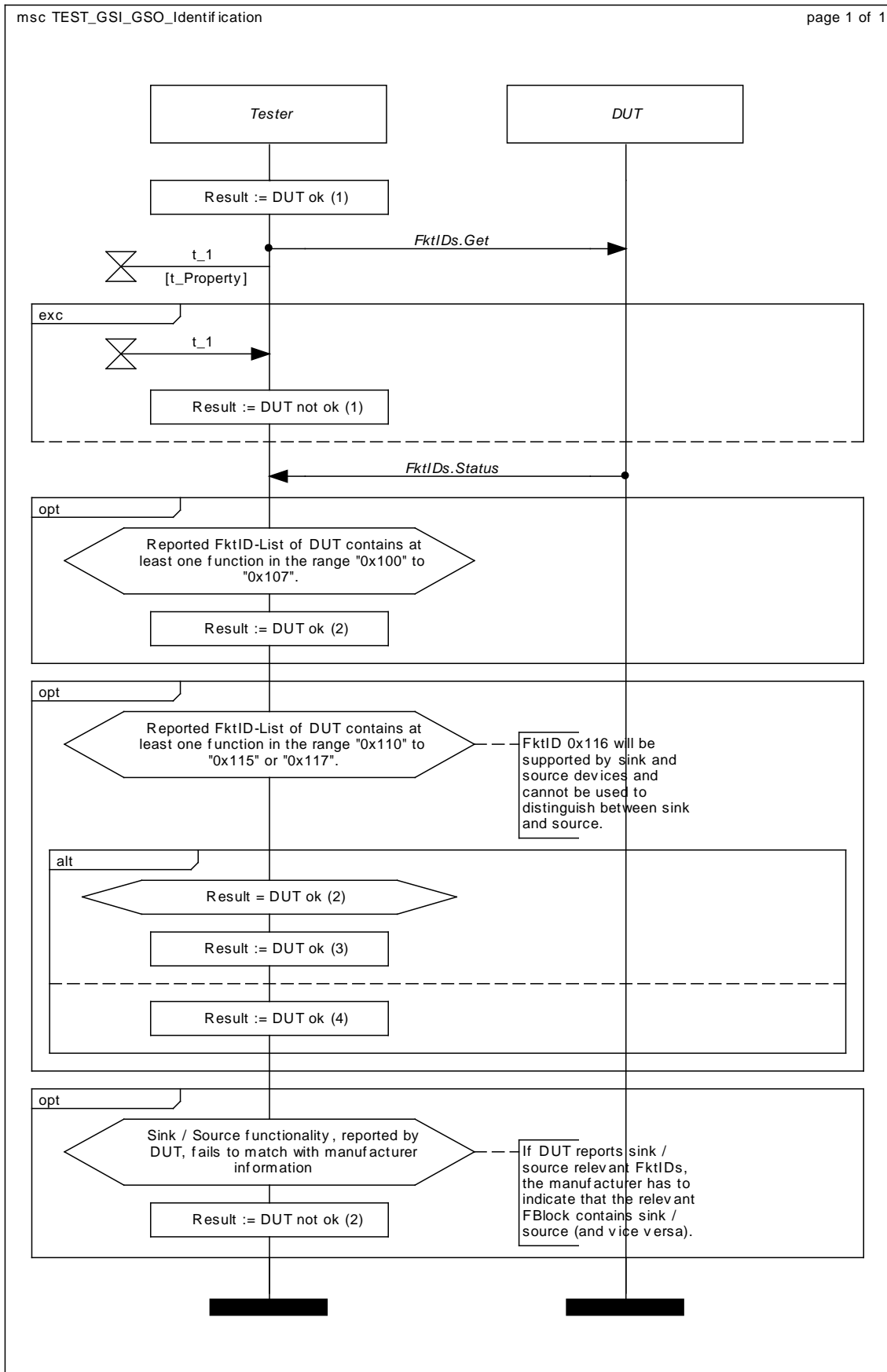
3.12.1.1 TEST_GSI_GSO_Identification (3.0-1)

Name of test	TEST_GSI_GSO_Identification 3.0-1
Reference to MOST Specification	[GFB] 2.1 GeneralFBlock
Value of Interest	t_Property Reported FktIDs
Preconditions	DUT in NormalOperation
Test focus	The tester tries to identify FBlocks of the DUT that contain sink or source functionalities.
Device type	All devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Devices	All devices
Note	<p>The test has to be performed to identify whether the DUT contains sink or source for synchronous streaming data.</p> <p>Test has to be performed with every single FBlock except PM, reported by the DUT by means of FBlockID.Status.</p> <p>The manufacturer has to provide list with all FBlocks of the DUT, containing sink and / or source functionality.</p> <p>DUT with result "DUT ok (1)" must not run through any source or sink related test</p> <p>DUT with result "DUT ok (2)" has to run through all source related tests (relevant FBlocks only)</p> <p>DUT with result "DUT ok (3)" has to run through all source and sink related tests (relevant FBlocks only)</p> <p>DUT with result "DUT ok (4)" has to run through all sink related tests (relevant FBlocks only)</p>
Results	<p>DUT ok (1): FBlock neither contains sink nor source</p> <p>DUT ok (2): FBlock contains source</p> <p>DUT ok (3): FBlock contains source and sink</p> <p>DUT ok (4): FBlock contains sink</p> <p>DUT not ok (1): DUT fails to send FktIDs.Status</p> <p>DUT not ok (2): FBlocks of DUT, containing sinks / sources do not match to manufacturer list</p>

CORE_GSI_GSO

msc TEST_GSI_GSO_Identification

page 1 of 1



3.13 Obligatory Tests For Sink And Source Devices

3.13.1 Sink Devices

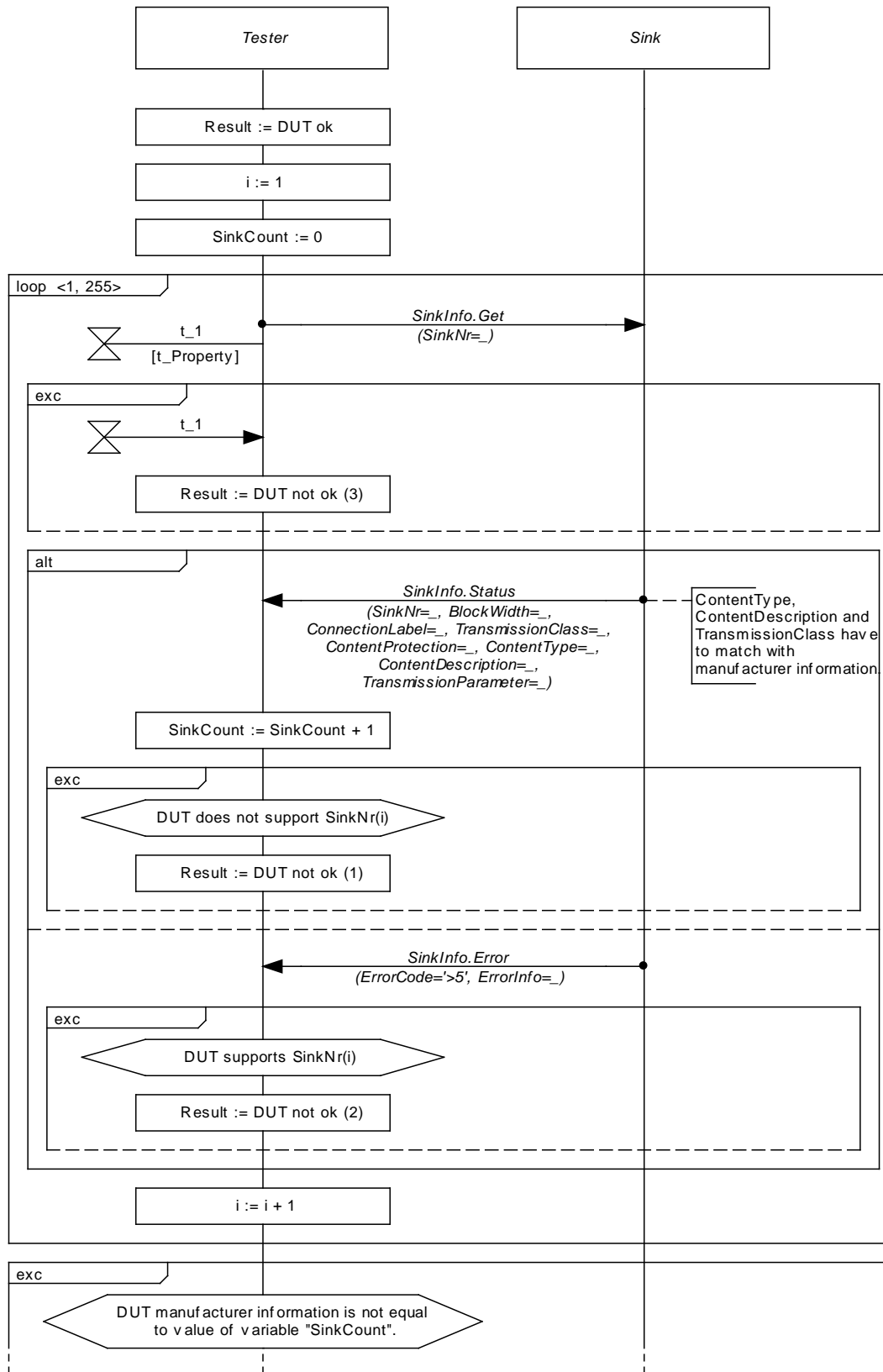
3.13.1.1 TEST_GSI_SinkInfo (3.1-1)

Name of test	TEST_GSI_SinkInfo 3.1-1
Reference to MOST Specification	[MS] para 3.2.7.2.1.2 Streaming Sink [GFB] 2.1.24 SinkInfo
Value of Interest	SinkInfo
Preconditions	DUT in NormalOperation
Test focus	DUT has to respond to SinkInfo.Get correct status information (or Error, if SinkNr not supported).
Device type	All sink devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Devices	All devices containing at least one sink
Note	Test has to be performed with every single SinkNumber (1..255). In cases where the bandwidth of sink cannot be determined because of the data type (ContentType 0xC0 ... 0xEF System Integrator Specific and 0xF0 ... 0xFE Supplier specific), the test case is skipped for this sink. The manufacturer information of each sink number will be checked. If DUT supports sink number, the DUT has to respond the correct ContentType, ContentDescription (data type of the parameter) and TransmissionClass.
Results	DUT ok (1): The DUT has passed the test DUT not ok (1): The DUT responds status although SinkNr not supported DUT not ok (2): The DUT responds with Error although SinkNr supported DUT not ok (3): The DUT fails to respond within t_Property to SinkInfo.Get DUT not ok (5): The DUT manufacturer information is not equal to reported number of sinks

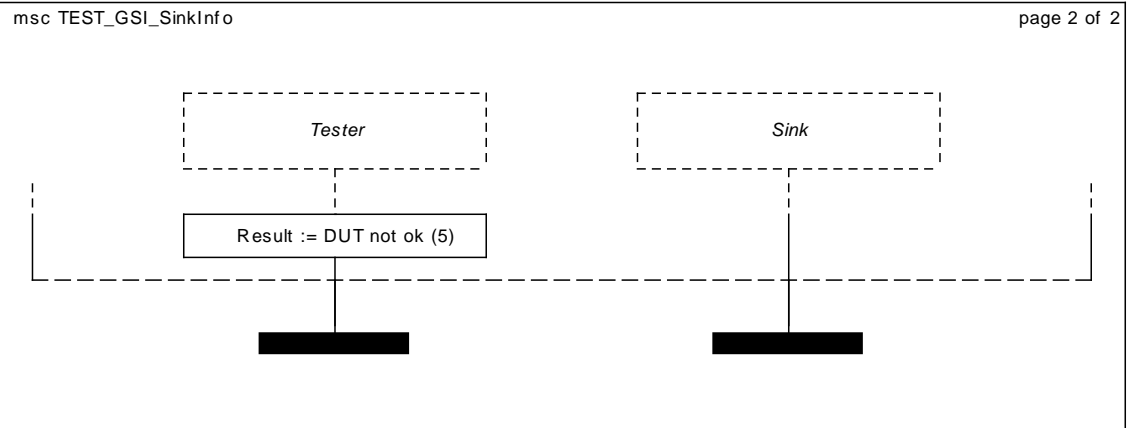
CORE_GSI_GSO

msc TEST_GSI_SinkInfo

page 1 of 2



CORE_GSI_GSO



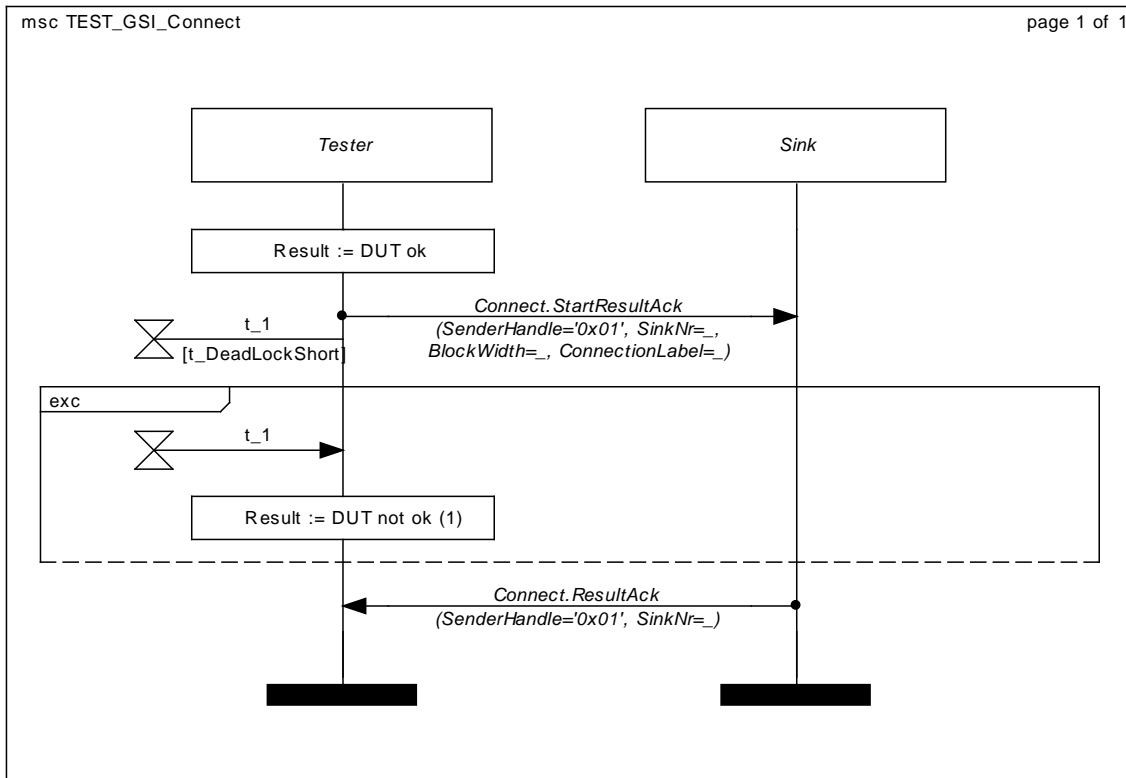
3.13.1.2 TEST_GSI_Connect (3.1-3)

Name of test	TEST_GSI_Connect 3.1-3
Reference to MOST Specification	[MDS] 4.5.1 Connect
Value of Interest	Connect.StartResultAck
Preconditions	DUT in NormalOperation
Test focus	DUT has to process Connect.StartResultAck correctly
Device type	All sink devices, supporting transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Devices	All devices containing at least one sink that supports transmission class "synchronous"
Note	<ul style="list-style-type: none"> - Test only applicable if DUT does not contain CM - Test to be performed with every single SinkNumber, supported by the DUT but Disconnect before in order to avoid resource overflow. - In cases where the bandwidth of sink cannot be determined because of the data type (ContentType 0xC0 ... 0xEF System Integrator Specific and 0xF0 ... 0xFE Supplier specific), the test case is skipped for this sink. - The tester has to ensure that the connections to which a sink has to connect to are in use by a source. (Reason: In some test cases the tester sends the message "Connect.StartResultAck" with BlockWidth and ConnectionLabel to the DUT (sink device) to trigger the DUT to connect to. Practice has shown that some sink devices check the usage of the connections they have to connect to before performing the connection. Only if the TM indicates the connections as allocated by a source, the sink device will connect. This behaviour is neither prescribed nor forbidden by the MOST Specification and has to be tolerated. - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok: The DUT has passed the test DUT not ok (1): The DUT fails to respond within t_DeadLockShort

CORE_GSI_GSO

mssc TEST_GSI_Connect

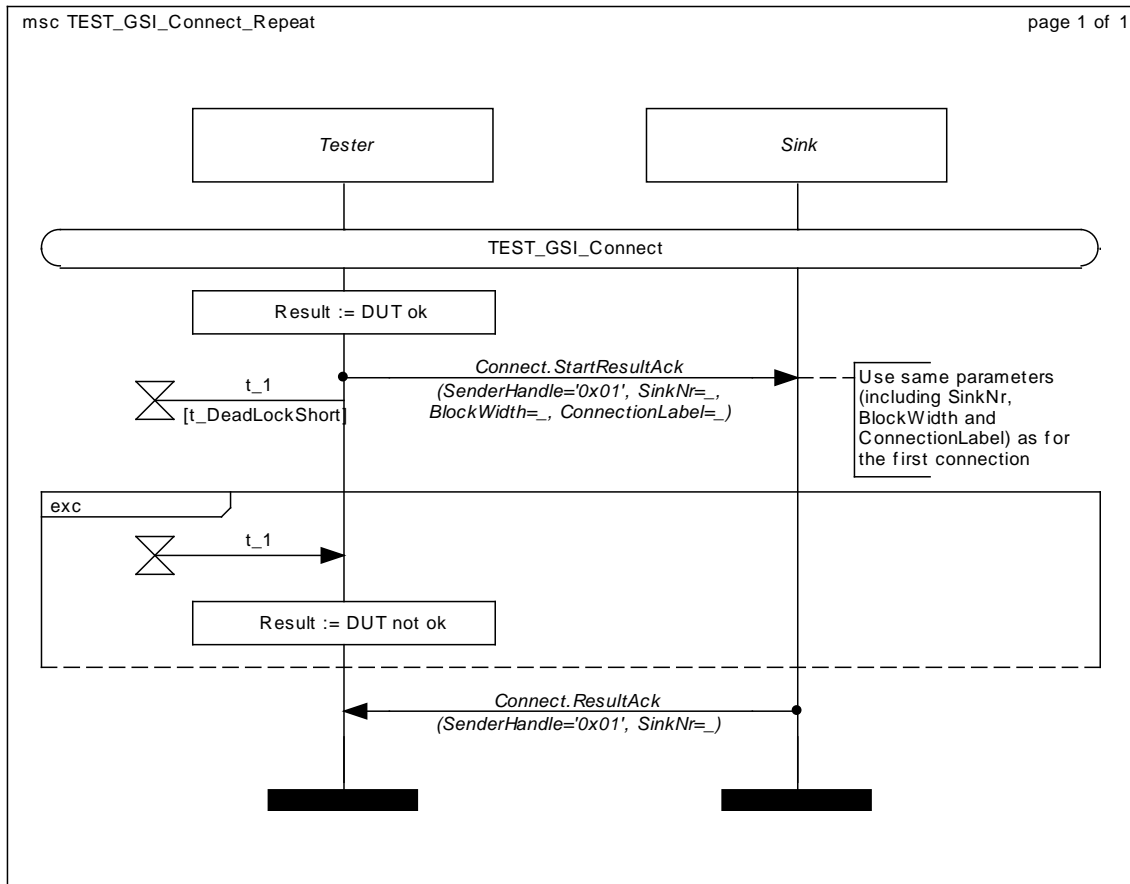
page 1 of 1



3.13.1.3 TEST_GSI_Connect_Repeat (3.1-4)

Name of test	TEST_GSI_Connect_Repeat 3.1-4
Reference to MOST Specification	[MDS] 4.5.1 Connect [MS] 3.2.7.2.1.3 Handling of Double Commands
Value of Interest	Connect.StartResultAck
Preconditions	DUT in NormalOperation
Test focus	DUT has to process repeated Connect.StartResultAck correctly
Device type	All sink devices, supporting transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Devices	All devices containing at least one sink that supports transmission class "synchronous"
Note	<ul style="list-style-type: none"> - Test only applicable if DUT does not contain CM - Test to be performed with every single SinkNumber, supported by the DUT but Disconnect before in order to avoid resource overflow. - In cases where the bandwidth of sink cannot be determined because of the data type (ContentType 0xC0 ... 0xEF System Integrator Specific and 0xF0 ... 0xFE Supplier specific), the test case is skipped for this sink. - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok: The DUT has passed the test DUT not ok: The DUT fails to respond within t_DeadLockShort

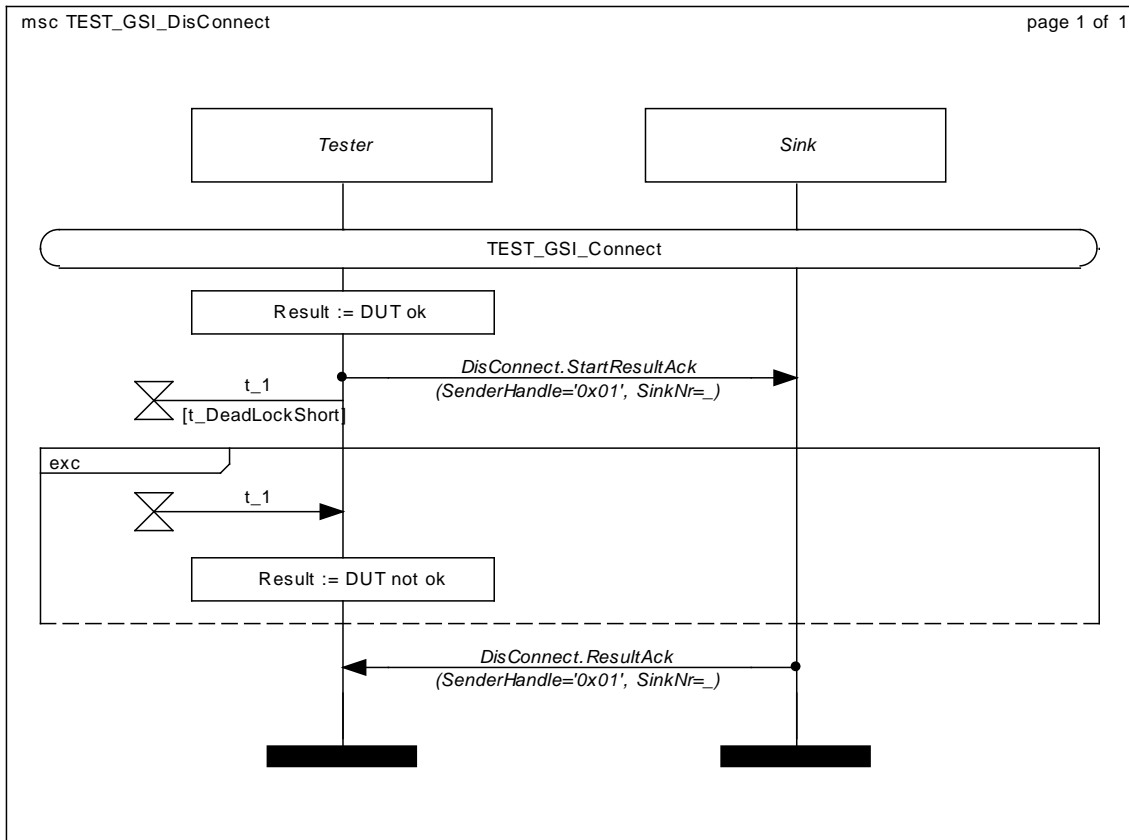
CORE_GSI_GSO



3.13.1.4 TEST_GSI_DisConnect (3.1-5)

Name of test	TEST_GSI_DisConnect 3.1-5
Reference to MOST Specification	[MDS] 4.5.2 Disconnect
Value of Interest	Disconnect.StartResultAck
Preconditions	- DUT in NormalOperation - DUT already connected to a source
Test focus	DUT has to disconnect correctly
Device type	All sink devices, supporting transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Devices	All devices containing at least one sink that supports transmission class "synchronous"
Note	- Test only applicable if DUT does not contain CM - Test to be performed with every single SinkNumber, supported by the DUT. - In cases where the bandwidth of sink cannot be determined because of the data type (ContentType 0xC0 ... 0xEF System Integrator Specific and 0xF0 ... 0xFE Supplier specific), the test case is skipped for this sink. - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok: The DUT has passed the test DUT not ok: The DUT fails to respond to Disconnect.StartResultAck within t_DeadLockShort

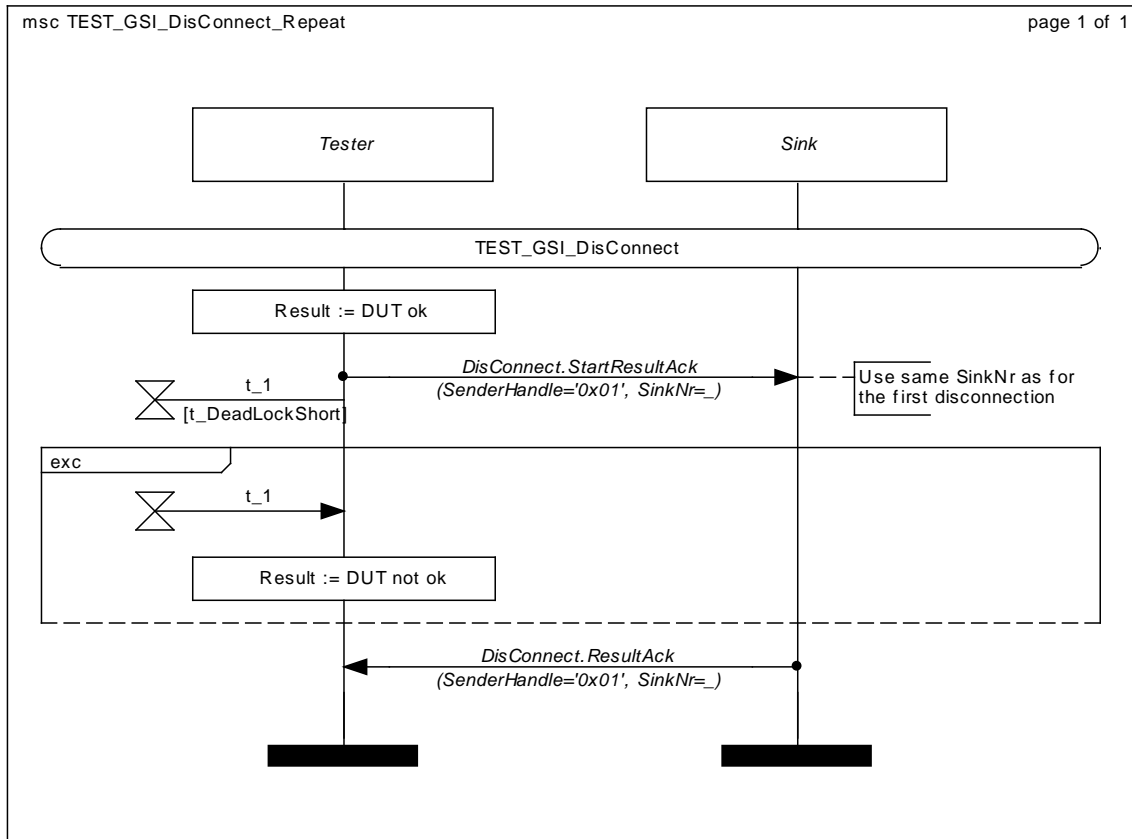
CORE_GSI_GSO



3.13.1.5 TEST_GSI_DisConnect_Repeat (3.1-6)

Name of test	TEST_GSI_DisConnect_Repeat 3.1-6
Reference to MOST Specification	[MDS] 4.5.2 Disconnect [MS] 3.2.7.2.1.3 Handling of Double Commands
Value of Interest	DisConnect.StartResultAck
Preconditions	- DUT in NormalOperation - DUT already connected to a source
Test focus	DUT must not generate any error in case of repeated disconnect command
Device type	All sink devices, supporting transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing at least one sink that supports transmission class "synchronous"
Note	- Test only applicable if DUT does not contain CM - Test to be performed with every single SinkNumber, supported by the DUT. - In cases where the bandwidth of sink cannot be determined because of the data type (ContentType 0xC0 ... 0xEF System Integrator Specific and 0xF0 ... 0xFE Supplier specific), the test case is skipped for this sink. - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok: The DUT has passed the test DUT not ok: The DUT fails to respond to repeated DisConnect.StartResultAck within t_DeadLockShort

CORE_GSI_GSO



3.13.2 Source Devices

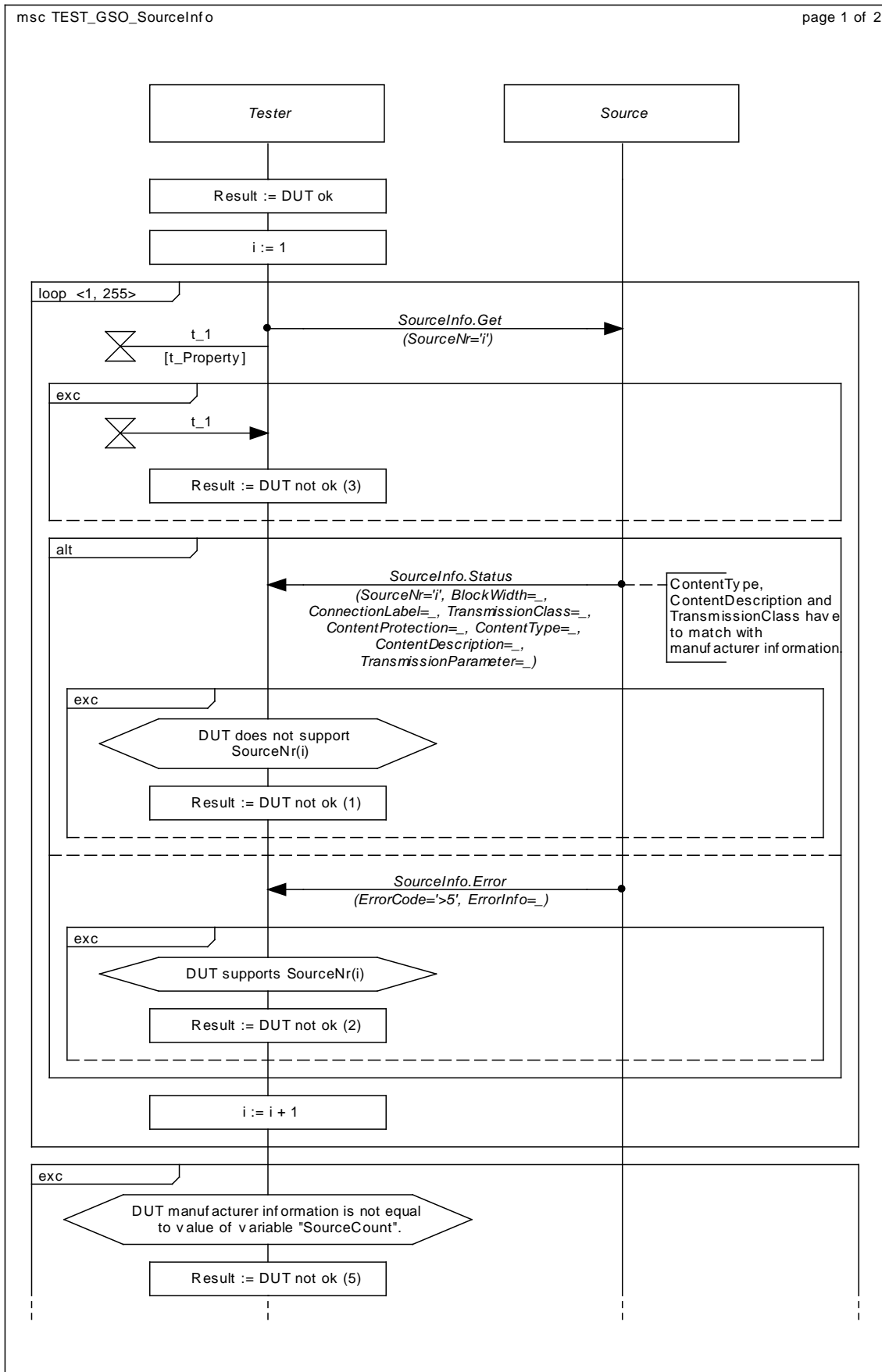
3.13.2.1 TEST_GSO_SourceInfo (3.2-1)

Name of test	TEST_GSO_SourceInfo 3.2-1
Reference to MOST Specification	[MS] 3.2.7.2.1.1 Streaming Source
Value of Interest	t_Property SourceInfo.Status
Preconditions	- DUT in NormalOperation
Test focus	DUT has to respond correct information to SourceInfo.Get.
Device type	All source devices [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing at least one source
Note	Test has to be performed with every single SourceNumber (1..255). The manufacturer information of each source number will be checked. If DUT supports source number, the DUT has to respond the correct ContentType, ContentDescription (data type of the parameter) and TransmissionClass.
Results	DUT ok: The DUT has passed the test DUT not ok (1): The DUT responds SourceInfo.Status although source info not supported. DUT not ok (2): The DUT responds with error. DUT not ok (3): The DUT fails to respond within t_Property DUT not ok (5): The DUT manufacturer information is not equal to reported number of sources

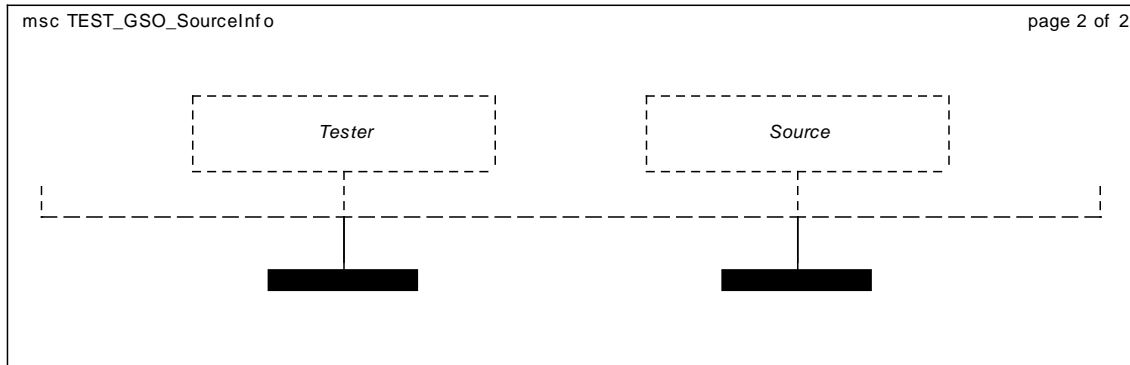
CORE_GSI_GSO

msc TEST_GSO_SourceInfo

page 1 of 2



CORE_GSI_GSO



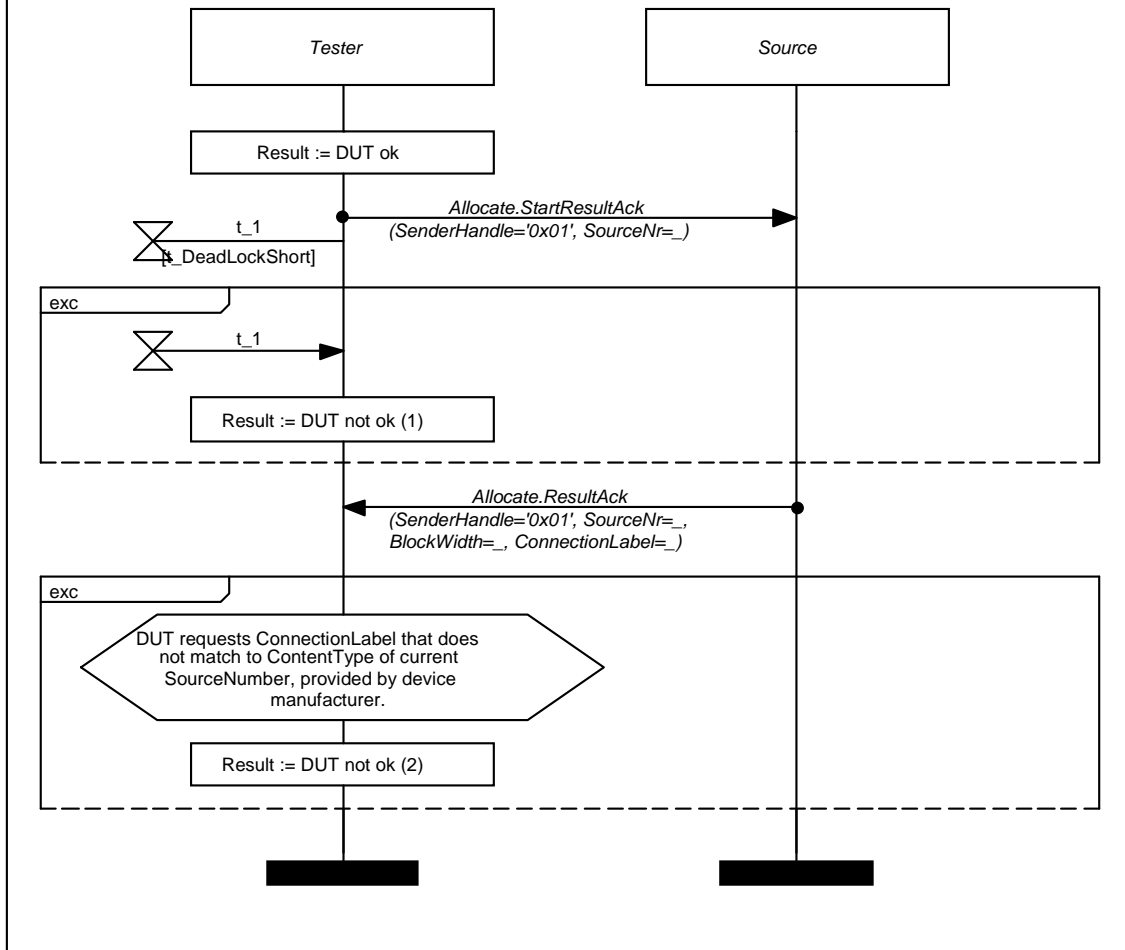
3.13.2.2 TEST_GSO_Allocate (3.2-3)

Name of test	TEST_GSO_Allocate 3.2-3
Reference to MOST Specification	[MDS] 4.3.1.4 Allocate
Value of Interest	Allocate.ResultAck
Preconditions	- DUT in NormalOperation
Test focus	DUT has to process Allocate.StartResultAck correctly
Device type	All source devices, supporting allocate mechanism and transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing at least one source that supports allocate mechanism and transmission class "synchronous"
Note	<ul style="list-style-type: none"> - Test only applicable if DUT does not contain CM - Test to be performed with every single SourceNumber, supported by the DUT. - BlockWidth and ConnectionLabel of DUT has to be provided by device manufacturer - Timeout "t_DeadLockShort" is valid for whole MSC. <p>Bandwidth constraints must be considered when beginning with next source number. Already allocated sources should be deallocated before sending Allocate.StartResultAck to a new SourceNr.</p>
Results	<p>DUT ok: The DUT has passed the test</p> <p>DUT not ok (1): The DUT fails to respond within t_DeadLockShort</p> <p>DUT not ok (2): Requested BlockWidth and ConnectionLabel of DUT does not match to information of DataType of manufacturer.</p>

CORE_GSI_GSO

msc TEST_GSO_Allocate

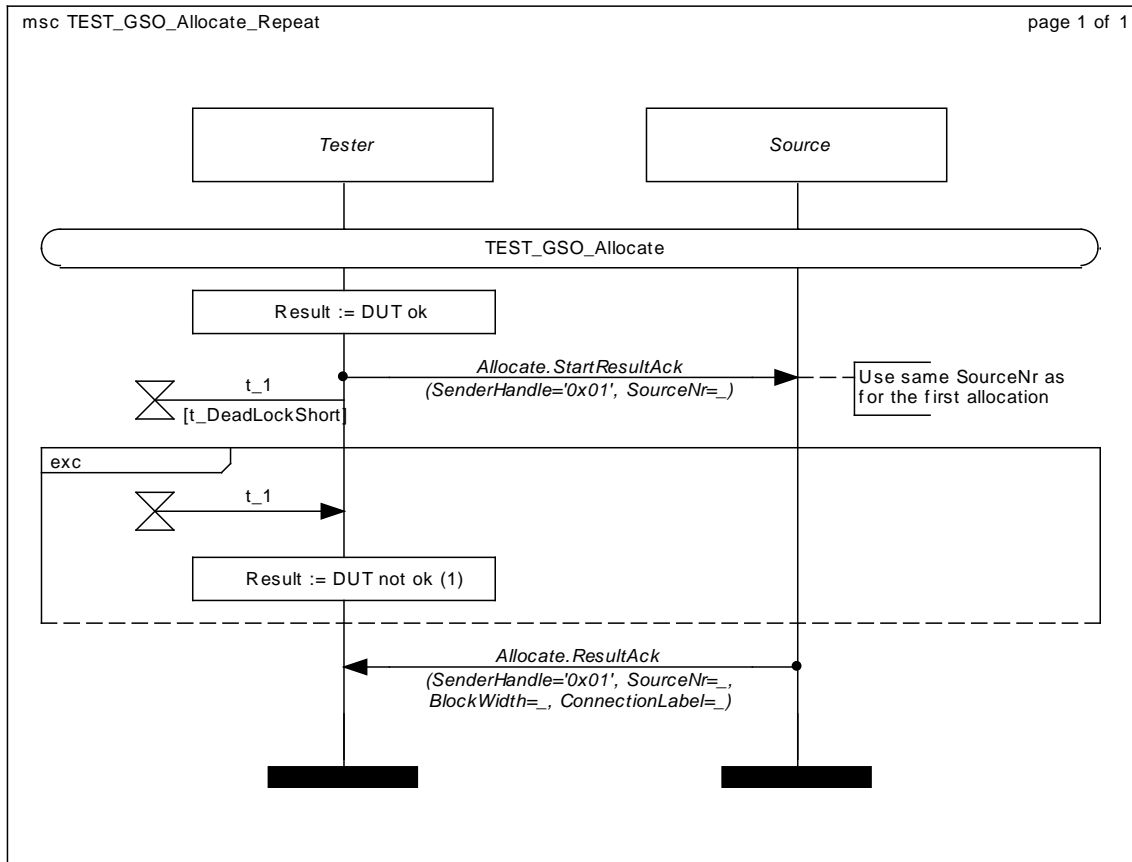
page 1 of 1



3.13.2.3 TEST_GSO_Allocate_Repeat (3.2-4)

Name of test	TEST_GSO_Allocate_Repeat 3.2-4
Reference to MOST Specification	[MDS] 4.3.1.4 Allocate [MS] 3.2.7.2.1.3 Handling of Double Commands
Value of Interest	Allocate.ResultAck
Preconditions	- DUT in NormalOperation
Test focus	DUT has to process repeated Allocate.StartResultAck correctly. It has to respond status without new allocation of connections.
Device type	All source devices, supporting allocate mechanism and transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing at least one source that supports allocate mechanism and transmission class "synchronous"
Note	- Test only applicable if DUT does not contain CM - Test to be performed with every single SourceNumber, supported by the DUT. - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok: The DUT has passed the test DUT not ok (1): The DUT fails to respond within t_DeadLockShort

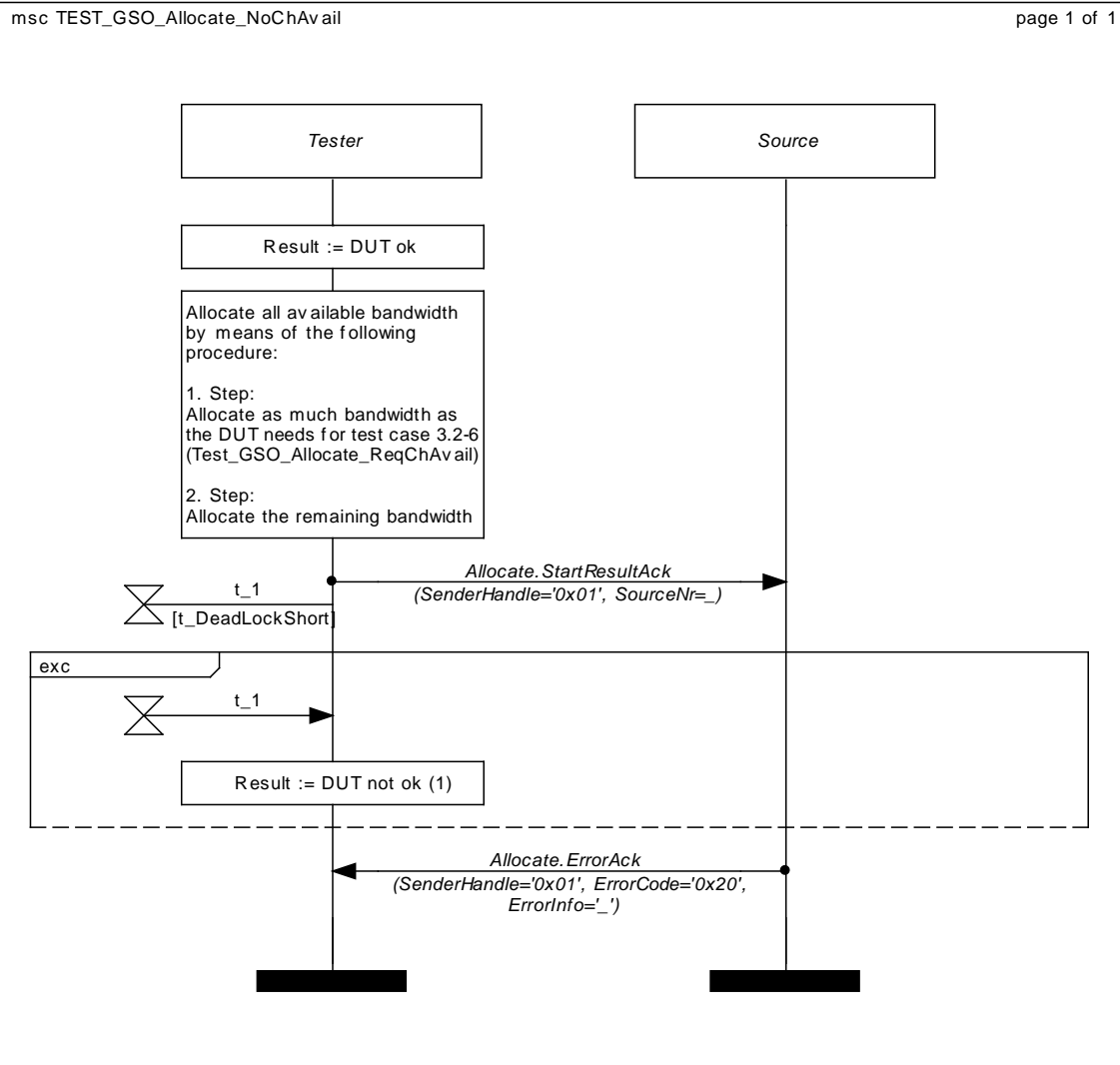
CORE_GSI_GSO



3.13.2.4 TEST_GSO_Allocate_NoChAvail (3.2-5)

Name of test	TEST_GSO_Allocate_NoChAvail 3.2-5
Reference to MOST Specification	[MDS] 4.3.1.4 Allocate [MS] 3.2.7.2.1.1 Streaming Source
Value of Interest	Allocate.ErrorAck
Preconditions	- DUT in NormalOperation
Test focus	DUT has to detect not enough bandwidth available during allocation and has to report correct error message.
Device type	All source devices, supporting allocate mechanism and transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing at least one source that supports allocate mechanism and transmission class "synchronous"
Note	- Test only applicable if DUT does not contain CM - Test to be performed with every single SourceNumber, supported by the DUT. - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok: The DUT has passed the test DUT not ok (1): The DUT fails to respond within t_DeadLockShort

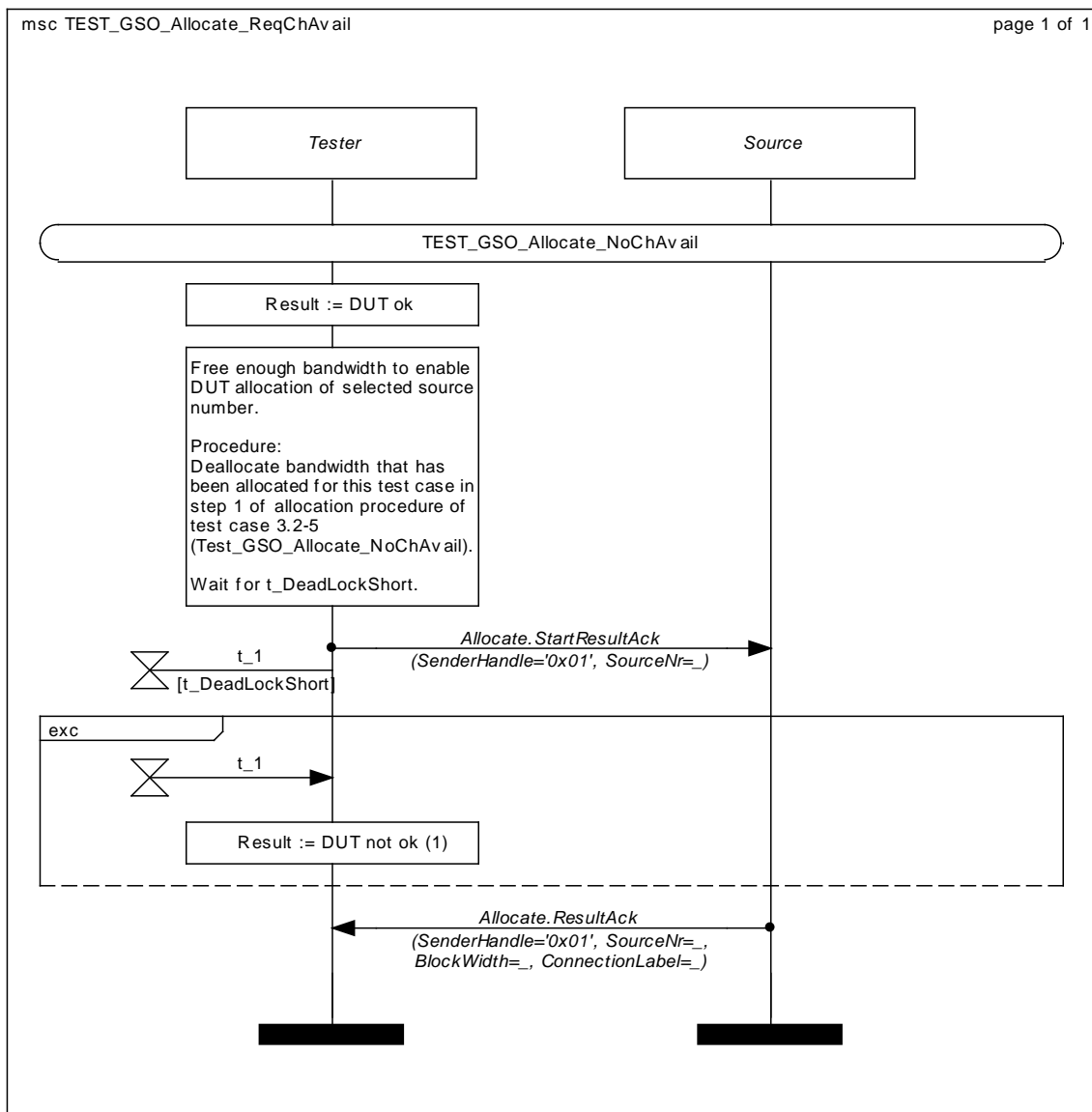
CORE_GSI_GSO



3.13.2.5 TEST_GSO_Allocate_ReqChAvail (3.2-6)

Name of test	TEST_GSO_Allocate_ReqChAvail 3.2-6
Reference to MOST Specification	[MDS] 4.3.1.4 Allocate [MS] 3.2.7.2.1.1 Streaming Source
Value of Interest	Allocate.ResultAck
Preconditions	- DUT in NormalOperation
Test focus	DUT has to be able to allocate connections if exactly required bandwidth is available.
Device type	All source devices, supporting allocate mechanism and transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing at least one source that supports allocate mechanism and transmission class "synchronous"
Note	- Test only applicable if DUT does not contain CM - Test to be performed with every single SourceNumber, supported by the DUT. - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok: The DUT has passed the test DUT not ok (1): The DUT fails to respond within t_DeadLockShort

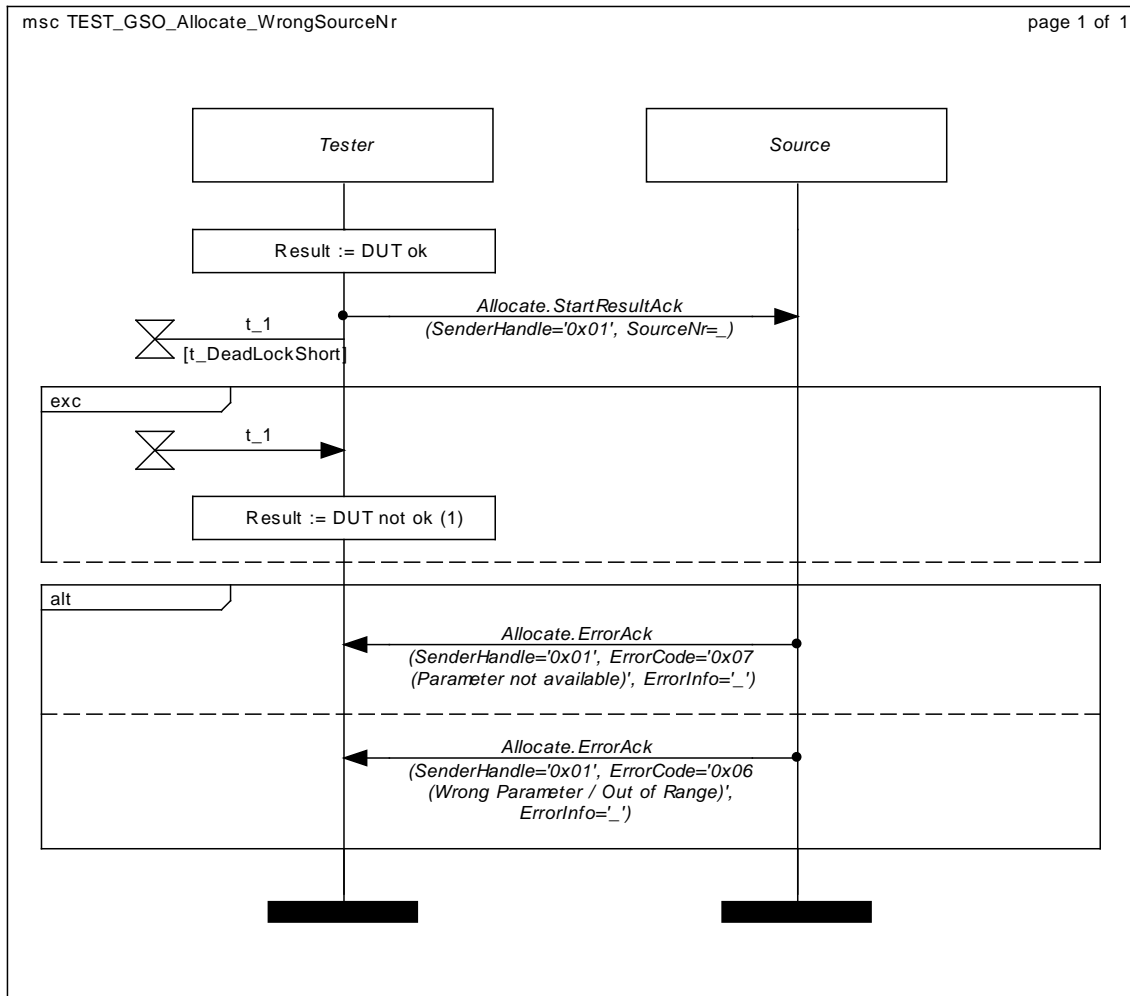
CORE_GSI_GSO



3.13.2.6 TEST_GSO_Allocate_WrongSourceNr (3.2-7)

Name of test	TEST_GSO_Allocate_WrongSourceNr 3.2-7
Reference to MOST Specification	[MDS] 4.3.1.4 Allocate [MS] 3.2.7.2.1.1 Streaming Source
Value of Interest	Allocate.ErrorAck
Preconditions	- DUT in NormalOperation
Test focus	DUT has to be able to detect wrong source number at allocation request and has to respond with error.
Device type	All source devices, supporting allocate mechanism and transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing at least one source that supports allocate mechanism and transmission class "synchronous"
Note	- Test only applicable if DUT does not contain CM - Test to be performed with first SourceNumber, not supported by the DUT. - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok: The DUT has passed the test DUT not ok (1): The DUT fails to respond within t_DeadLockShort

CORE_GSI_GSO



3.13.2.7 TEST_GSO_DeAllocate (3.2-8)

Name of test	TEST_GSO_DeAllocate 3.2-8
Reference to MOST Specification	[MDS] 4.3.1.5.2 Deallocation Procedure [MS] 3.2.7.2.1.1 Streaming Source
Value of Interest	DeAllocate.ResultAck
Preconditions	- DUT in NormalOperation
Test focus	DUT has to process DeAllocate.StartResultAck correctly
Device type	All source devices, supporting allocate mechanism and transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing at least one source that supports allocate mechanism and transmission class "synchronous"
Note	- Test only applicable if DUT does not contain CM. - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok: The DUT has passed the test DUT not ok (1): The DUT fails to respond within t_DeadLockShort

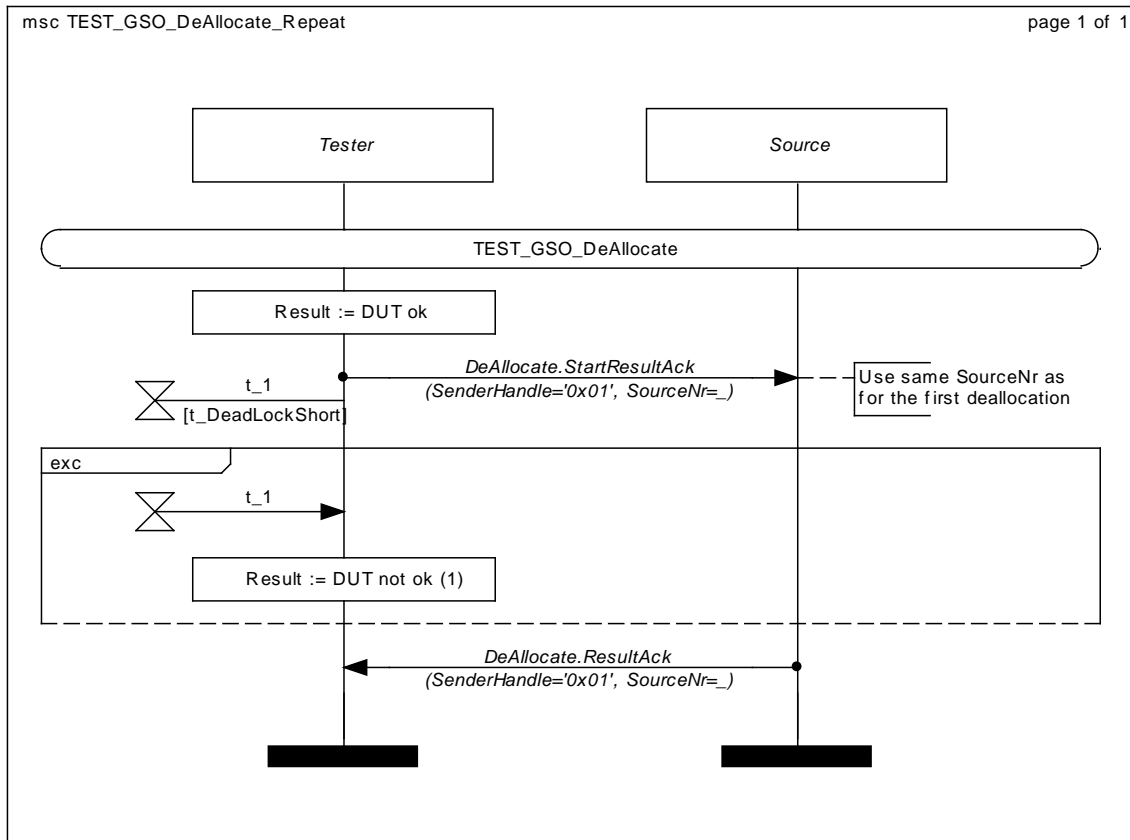
```
msc TEST_GSO_DeAllocate
```

```
sequenceDiagram
    participant Tester
    participant Source
    activate Tester
    Note over Tester: Result := DUT ok
    Tester->>Source: DeAllocate.StartResultAck  
(SenderHandle='0x01', SourceNr=_)
    activate Source
    Source->>Tester: DeAllocate.ResultAck  
(SenderHandle='0x01', SourceNr=_)
    deactivate Source
    Note over Tester: Result := DUT not ok (1)
    Tester->>Tester: exc
    Note over Tester: [t_DeadLockShort]
    Note over Tester: t_1
    Note over Tester: t_1
    deactivate Tester
```

3.13.2.8 TEST_GSO_DeAllocate_Repeat (3.2-9)

Name of test	TEST_GSO_DeAllocate_Repeat 3.2-9
Reference to MOST Specification	[MDS] 4.3.1.5.2 Deallocation Procedure [MS] 3.2.7.2.1.3 Handling of Double Commands
Value of Interest	DeAllocate.ResultAck
Preconditions	- DUT in NormalOperation
Test focus	DUT has to process repeated DeAllocate.StartResultAck correctly
Device type	All source devices, supporting allocate mechanism and transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing at least one source that supports allocate mechanism and transmission class "synchronous"
Note	- Test only applicable if DUT does not contain CM. - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok: The DUT has passed the test DUT not ok (1): The DUT fails to respond within t_DeadLockShort

CORE_GSI_GSO



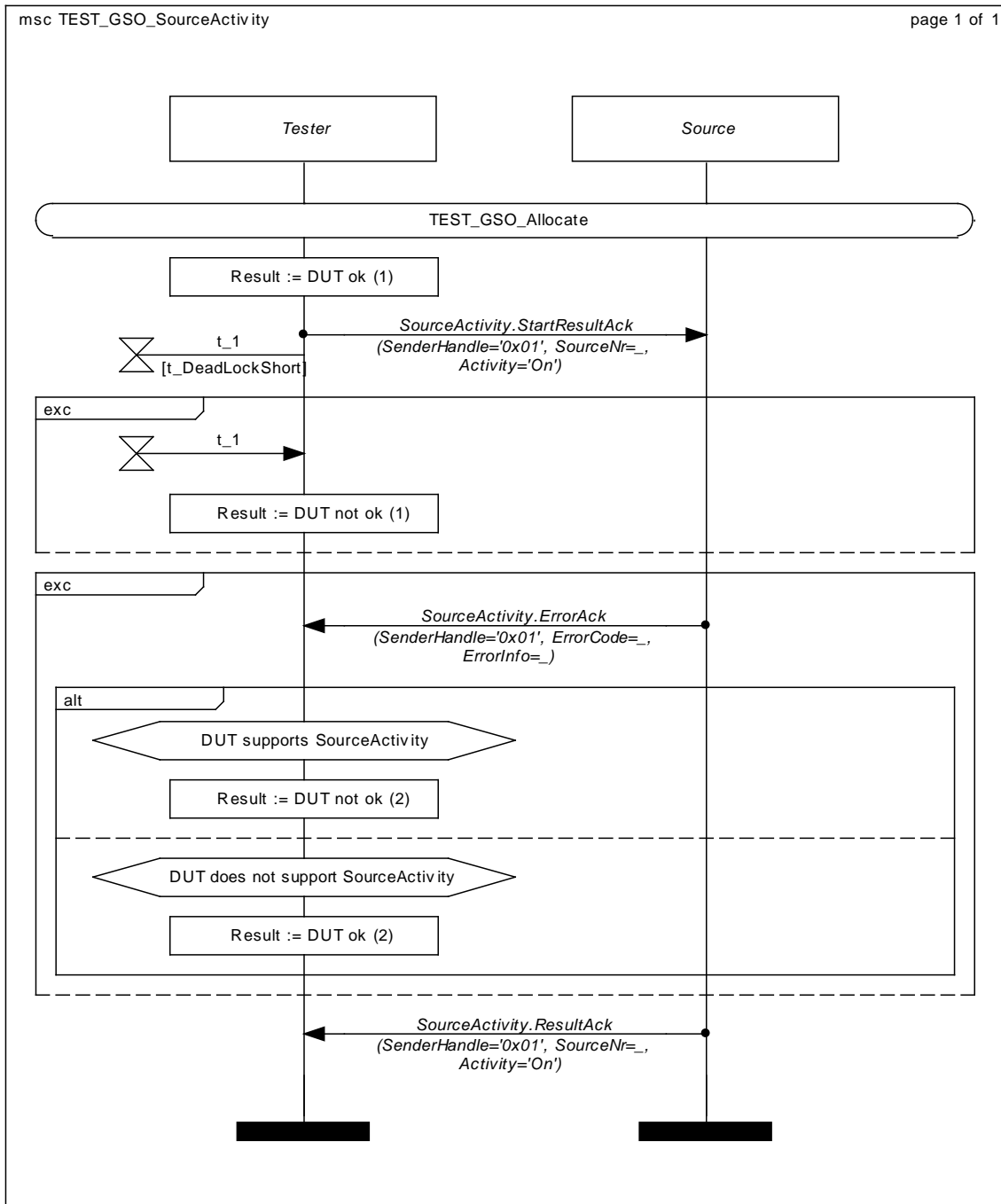
3.13.2.9 TEST_GSO_SourceActivity (3.2-14)

Name of test	TEST_GSO_SourceActivity 3.2-14
Reference to MOST Specification	[MDS] 4.3.1.2 SourceActivity turned on [MS] 3.2.7.2.1.1 Streaming Source
Value of Interest	SourceActivity.ResultAck
Preconditions	- DUT in NormalOperation
Test focus	DUT has to set SourceActivity to "On" if commanded.
Device type	All source devices, supporting allocate mechanism and transmission class "synchronous" [x] MOST150 oPHY [x] MOST150 cPHY [x] MOST50 ePHY
Multi Node Device	All devices containing at least one source that supports allocate mechanism and transmission class "synchronous"
Note	- Test only applicable if DUT does not contain CM - Test to be performed with every SourceNumber, supported by the DUT. - Input by device manufacturer whether DUT supports SourceActivity - Timeout "t_DeadLockShort" is valid for whole MSC.
Results	DUT ok (1): The DUT has passed the test DUT ok (2): The DUT does not support SourceActivity DUT not ok (1): The DUT fails to respond within t_DeadLockShort DUT not ok (2): The DUT responds with ErrorAck although SourceActivity should be supported.

CORE_GSI_GSO

msc TEST_GSO_SourceActivity

page 1 of 1



4 Appendix 1: Requirements And Measurement Uncertainty For Test Equipment And Setup

Test Case Nr.	Designation	Measured value	Measurement uncertainty
2.1.0-1	Generic FBlock Property test	t_Property	+20ms
2.1.0-2	Generic FBlock Method test	t_ProcessingDefault1	+20ms
2.1.1-4	Wake-up / Shutdown test	t_Config	+10ms
2.1.1-5	Bypass test	t_test (t_Lock + t_WaitNodes)	+10ms
2.1.1-6a	System Lock Flag Test (a)	-	-
2.1.1-6b	System Lock Flag Test (b)	t_Config	+10ms
2.1.2-5	Slave lock detection test	t_Config + t_WaitNodes	+20ms
2.1.3-1	Slave wake-up	t_Restart	+5ms
2.1.3-4	Waking slave timeout	t_Restart t_Config	+5ms +5ms
2.2.1-3	Short unlock test	t_Unlock t_Lock	+1ms, (Verification of setup)
2.2.1-4	Accumulated short unlock test	t_test_min t_test_max t_SSO_ShutDown t_Restart	+1ms, (Verification of setup) +5ms +5ms
2.2.1-5	Critical unlock test	t_Unlock t_Restart (MOST signal case) t_SSO_ShutDown	+5ms +5ms +5ms
2.2.1-6	Sudden Signal Off test	t_SSO_ShutDown	+5ms
2.2.1-7	Shutdown Flag present test	-	-
2.3.1-3	Timeout execute / Timeout suspend	t_WaitSuspend t_ShutDownWait	+5ms +5ms
2.3.1-6	Timeout suspend	t_RetryShutDown	+25ms
2.3.2-2	ShutDown.Start(Query) test	t_WaitSuspend	+5ms
2.3.2-3	ShutDown.Start(Execute) test	t_PwrSwitchOffDelay	+500ms
2.3.3-1	Over-/ undervoltage test	t_test	Not required
2.3.3-4	Over-temperature signal off test	-	-
2.3.3-5	Temperature ShutDown Reaction Test	t_WaitAfterOvertempShutDown	+500ms
2.4.1-1	Restart stop test	t_Config t_Restart	+10ms +5ms
2.4.1-2	Restart continue test	t_Config t_Restart	+10ms +5ms
2.4.1-9	Reaction of NCE test	t_WaitAfterNCE	+30 ms
2.4.1-11	Re-enter delay test	t_Bypass	+ 5 ms
2.4.2-1	Critical Voltage test	<= 9V / (>=16.5V)	+0.1V

Test Case Nr.	Designation	Measured value	Measurement uncertainty
2.4.2-3	Low Voltage test	$\leq 7V$	$\pm 0.1V$
2.5-2	RBD procedure test MOST50 ePHY	t_Diag_Slave t_Diag_Master t_PwrSwitchOffDelay	Not required Not required Not required
2.5-3	TM RBD procedure test MOST50 ePHY	t_Diag_Slave t_Diag_Master t_PwrSwitchOffDelay	Not required Not required Not required
2.5-7	Wakeup after RBD test MOST50 ePHY	t_test (t_Diag_Restart + t_Restart)	$\pm 5ms$
2.5-8	RBD missing TM test MOST50 ePHY	t_Diag_Slave	$\pm 200ms$
2.5-10	RBD closed ring test MOST150 oPHY	-	-
2.5-11	RBD interruption in front of DUT test MOST150 oPHY	-	-
2.5-12	RBD interruption behind DUT test MOST150 oPHY	-	-
2.6.1-1	NWM address initialization test	-	-
2.6.2-1	FBlock polling test	t_WaitForAnswer	$\pm 5ms$
2.6.2-3a	Device Ignore test (a)	-	-
2.6.2-3b	Device Ignore test (b)	t_DelayCfgRequest1 t_DelayCfgRequest2	$\pm 20ms$ $\pm 200ms$
2.6.2-4a	Device Integration test (a)	-	-
2.6.2-4b	Device Integration test (b)	-	-
2.6.2-4c	Device Integration test (c)	-	-
2.6.2-5	Config(Ok) Delay test	t_WaitBeforeScan t_WaitForAnswer	$\pm 20ms$ $\pm 5ms$
2.6.2-6	Double FBlock test	-	-
2.6.2-7	Config(New) Order test	-	-
2.6.3-1	FBlock Status Change Detection test	-	-
2.6.4-1	Address Initialization test	-	-
2.6.4-3	NWM Address Storage test	-	-
2.6.4-4	Address Re-Initialization test	-	-
2.6.4-8	FBlock Response Time test	t_Answer	$\pm 5ms$
2.6.4-9	Control message suppression test	-	-
2.6.4-10	InstID Wildcard Test	-	-
2.7-1	Node Addressing test	-	-
2.8.3-1a	Notification Matrix Storage test (NWM)	t_Property	$\pm 20ms$
2.8.3-1b	Notification Matrix Storage test (Slave)	t_Property	$\pm 20ms$
2.8.3-2	NotificationCheck test	-	-

Test Case Nr.	Designation	Measured value	Measurement uncertainty
2.8.3-7	Notification Matrix Double Entry test	t_WaitforProperty	+20ms
2.8.3-10	Notification Error test	-	-
2.8.4-1	Segmented Message Rejection test	t_WaitForNextSegment	+20ms
2.8.4-2	Segmented Message Sending test	-	-
2.8.4-3	Message Segmentation Error test	t_Property	+20ms
2.8.4-7	Message Segmentation Buffer test	-	-
2.8.4-8	Parallel Segmented Message Reception test	-	-
2.8.4-9	Broadcast error test	-	-
3.0-1	TEST_GSI_GSO_Identification	t_Property	+20ms
3.1-1	TEST_GSI_SinkInfo	t_Property	+20ms
3.1-3	TEST_GSI_Connect	-	
3.1-4	TEST_GSI_Connect_Repeat	-	
3.1-5	TEST_GSI_DisConnect	t_Property	+20ms
3.1-6	TEST_GSI_DisConnect_Repeat	-	
3.2-1	TEST_GSO_SourceInfo	t_Property	+20ms
3.2-3	TEST_GSO_Allocate	-	
3.2-4	TEST_GSO_Allocate_Repeat	-	
3.2-5	TEST_GSO_Allocate_NoChAvail	-	
3.2-6	TEST_GSO_Allocate_ReqChAvail	-	
3.2-7	TEST_GSO_Allocate_WrongSourceNr	-	
3.2-8	TEST_GSO_DeAllocate	-	
3.2-9	TEST_GSO_DeAllocate_Repeat	-	
3.2-14	TEST_GSO_SourceActivity	-	

5 Appendix 2: FBlock ET Reference To Test Cases (Informative)

Note:

"-" means, this function is not used by MOST Core Compliance but by other compliance tests.

FktID	Name	Test Case	Page
0x201	AutoWakeup	2.1.1-4 (Wake-up / Shutdown test)	35
		2.1.1-6a (System Lock Flag Test (a))	31
		2.1.3-1 (Slave wake-up)	39
		2.1.3-4 (Waking slave timeout)	41
		2.4.1-1 (Restart stop test)	67
		2.4.1-2 (Restart continue test)	69
		2.5-2 (RBD procedure test MOST50 ePHY)	79
		2.5-3 (TM RBD procedure test MOST50 ePHY)	81
		2.5-7 (Wakeup after RBD test MOST50 ePHY)	83
		2.5-8 (RBD missing TM test MOST50 ePHY)	85
		2.5-10 (RBD closed ring test MOST150 oPHY)	87
		2.5-11 (RBD interruption in front of DUT test MOST150 oPHY)	89
		2.5-12 (RBD interruption behind DUT test MOST150 oPHY)	91
0x203	DiagResult	2.5-2 (RBD procedure test MOST50 ePHY)	79
		2.5-3 (TM RBD procedure test MOST50 ePHY)	81
		2.5-8 (RBD missing TM test MOST50 ePHY)	85
		2.5-10 (RBD closed ring test MOST150 oPHY)	87
		2.5-11 (RBD interruption in front of DUT test MOST150 oPHY)	89
		2.5-12 (RBD interruption behind DUT test MOST150 oPHY)	91
0x204	Shutdown	2.3.1-3 (Timeout execute / Timeout suspend)	53
		2.3.3-4 (Over-temperature signal off test)	65
0x205	ShutdownSuspendMode	-	-
0x207	SendMessage	2.8.4-2 (Segmented Message Sending test)	143
0x208	EchoMessage	2.8.4-7 (Message Segmentation Buffer test)	147
0x209	MessageBufSize	2.8.4-7 (Message Segmentation Buffer test)	147
		2.8.4-8 (Parallel Segmented Message Reception test)	149
0x211	Reset	2.4.1-11 (Re-enter delay test)	73
0x212	CentralRegistrySize	2.6.2-4b (Device Integration test (b))	103
0x217	MOSTRemoteReset	-	-
0x218	PhysicalLayerTest	-	-
0x219	PhysicalLayerTestResult	-	-

FktID	Name	Test Case	Page
0x220	ECLTrigger	-	-
0x221	ECLInitiatorState	-	-
0x3C8	void ¹	2.8.4-9 (Broadcast error test)	129
0x3FD	DSIDSOCCount	-	-
0x3FE	DSO	-	-
0x400	DSI	-	-

¹ Function reserved; must not be implemented

6 Appendix 3: Document History

Document History

Changes MOST Core Compliance Test Specification Rev. 3.0E6 to MOST Core Compliance Test Specification Rev. 3.0E7.

Change Ref.	Section	Changes
3V0E7_001	<p>InstID Wildcard Test</p> <p>NotificationCheck test</p> <p>TEST_GSI_GSO_Identification</p>	<p>1.) Test 2.6.4-10 InstID Wildcard Test: The test has to be performed with every single registered FBlock of the DUT, except Netblock and FBlock ET. → The test has to be performed with every single registered FBlock of the DUT, except (Netblock and FBlock ET and) PM.</p> <p>2.) Test 2.8.3-2 NotificationCheck Test: The test has to be performed with every FBlock of the DUT, registered in the CR. → The test has to be performed with every FBlock registered in the CR except PM.</p> <p>3.) Test 3.0-1 TEST_GSI_GSO_Identification: Test has to be performed with every single FBlock, reported by the DUT by means of FBlockID.Status. → Test has to be performed with every single FBlock except PM, reported by the DUT by means of FBlockID.Status.</p> <p><i>Reason:</i> <i>It does not make sense to check the PM with FktIDs.Get even if the FBlock PM is registered in the CR.</i> D240-1</p>
3V0E7_002	Sudden Signal Off test	<p>2.2.1-6 Sudden Signal Off test Introduce description: DUT not ok (6): DUT does not set Shutdown Flag correctly.</p> <p><i>Reason: missing result description</i> D244-2</p>

Change Ref.	Section	Changes
3V0E7_003	Parallel Segmented Message Reception test	<p>2.8.4-8 Parallel Segmented Message Reception test</p> <p>Adapt: The tester sends as much first parts of segmented messages ("NetBlock.InstID.<Count>.Get") → The tester sends as many first parts of segmented messages ("ET.InstID.<FunctionID>.Get") as supported by the DUT (every message is sent by a unique value of FunctionID).</p> <p>Remark: "Count" is a placeholder for function ID. D249-1</p>
3V0E7_004	Timeout execute / Timeout suspend Timeout suspend	<p>- 2.3.1-3 Timeout execute / Timeout suspend - 2.3.1-6 Timeout suspend</p> <p>Proposal Improve Core Compliance – PM tests</p> <p>Add to DUT manufacturer list whether PM supports suspend, If the PM does not support suspend the following test case will be omitted: - 2.3.1-3 Timeout execute / Timeout suspend - 2.3.1-6 Timeout suspend</p> <p><i>Reason: The behavior is actually not mandatory. MOST Specification, Rev. 3.0 E2 07/2010, p. 131: "If the PowerMaster receives ShutDown.Result (Suspend) within time tWaitSuspend, it postpones its attempt to switch off for time tRetryShutDown, before retrying to shut down. The PowerMaster may override suspend requests from its Slaves and complete the shutdown, for example, to prevent the occurrence of critical conditions like low voltage."</i></p> <p>D249-3</p>
3V0E7_005	3.1.5	<p>DUT manufacturer List / 2.7-1 Node Addressing Test</p> <p>Adapt: Free FBlock range FBlocks that are not used by the DUT and which can be used by the tester (the range is determined by the size of the central registry divided by 254) ====> DUT manufacturer List Device Integration Test (b) (2.6.2-4b) Device Integration Test (c) (2.6.2-4c)</p> <p>Free FBlock range FBlocks that are not used by the DUT and which can be used by the tester (the range is determined by the size of the central registry divided by 254) <i>Reason: The free block range is necessary for the device integration tests but not for the Node Addressing Test.</i></p> <p>D253-2</p>

Change Ref.	Section	Changes
3V0E7_006	FBlock polling test	<p>Clarify: 2.6.2-1 FBlock Polling Test</p> <p>Delete:</p> <p>“After wakeup, the DUT has to check all function blocks of both testers in the sequence of their ring position.”</p> <p><i>Reason: Test description is clear without this sentence. No such requirement for the DUT according to MOST Specification.</i></p> <p>D259-1</p>
3V0E7_007	Reaction of NCE test	<p>Clarify: 2.4.1-9 Reaction Of NCE Test</p> <p>Delete:</p> <ul style="list-style-type: none"> a) “Test case “NCE with unlock” only applicable if no TM node in front of NWM node inside the DUT.” b) “b) The NCE is generated without unlock (only applicable if DUT <> TM)” c) “b) The NCE is generated without unlock (only applicable if DUT <> TM)” <p>Change:</p> <p>“The test has to be performed twice:” → “The test has to be performed once:”</p> <p><i>Reason: Test case only for NWM. According to MOST Specification “the NetworkMaster FBlock shall reside in the same node as the TimingMaster”.</i></p> <p>D259-2</p>
3V0E7_008	Node Addressing test	<p>Clarify: 2.7-1 Node Addressing Test</p> <p>Delete:</p> <ul style="list-style-type: none"> a) If DUT contains TM, the ring position can only be changed of nodes in front of the TM. b) “If DUT is not TM:” c) “If DUT is TM: Ring position must not be changed.” <p><i>Reason: DUT never is TM as the test case is applicable for “all devices except NWM” and according to MOST Specification “the NetworkMaster FBlock shall reside in the same node as the TimingMaster”.</i></p> <p>D259-3</p>
3V0E7_009	TEST_GSO_Allocate	<p>Clarify: 3.2-3 TEST_GSO_Allocate</p> <p>DUT not ok (2) is missing.</p> <p>Rename “DUT not ok (3):” → “DUT not ok (2):”</p> <p>D259-4</p>
3V0E7_010	Sudden Signal Off test	<p>Clarify: 2.2.1-6 Sudden Signal Off Test</p> <p>Delete:</p> <p>“In case DUT is multi-node device, results of all nodes have to be read out.”</p> <p><i>Reason: The result of “other” nodes has no impact on test result. The existing note “In case of multi-node device: Shutdown result only valid for first node of device” is sufficient.</i></p> <p>D259-5</p>

Change Ref.	Section	Changes
3V0E7_011	Generic FBlock Property test	<p>Clarify: 2.1.0-1 Generic FBlock Property Test</p> <p>Delete:</p> <p>"Each OType test presumes that the previous test has completed the communication process, e.g. if after StartResult the Result is not received, then the process has to be finished by sending Abort. In case Abort fails, the DUT can be reset to finish the test step and continue with the next step."</p> <p><i>Reason: For FBlock Property Test, StartResult is not applicable.</i></p> <p>D259-6</p>

Changes MOST Core Compliance Test Specification Rev. 3.0E5 to MOST Core Compliance Test Specification Rev. 3.0E6.

Change Ref.	Section	Changes
3V0E6_001	Para 3.1.5	Update manufacturer information: Sink /source device: ContentDescription → ContentDescription (data type of the parameter)
3V0E6_002	2.8.4-1	„NetBlock.FBlockIDs.Get“ → "ET.EchoMessage.StartResult()" <i>Reason: NetBlock communication might be handled totally by the NIC therefore an applicative message must be used for this test case</i>
3V0E6_003	2.8.4-3	„NetBlock.FBlockIDs.Get“ → "ET.EchoMessage.StartResult()" <i>Reason: NetBlock communication might be handled totally by the NIC therefore an applicative message must be used for this test case</i>
3V0E6_004	3.1-1	Update Note: The manufacturer information of each sink number will be checked. If DUT supports sink number, the DUT has to respond the correct ContentType, ContentDescription (data type of the parameter) and TransmissionClass.
3V0E6_005	3.2-1	Update Note: The manufacturer information of each source number will be checked. If DUT supports source number, the DUT has to respond the correct ContentType, ContentDescription (data type of the parameter) and TransmissionClass.

Changes MOST Core Compliance Test Specification Rev. 3.0E4 to MOST Core Compliance Test Specification Rev. 3.0E5.

Change Ref.	Section	Changes
3V0E5_001	Para 3.1.5	<p>(1)</p> <p>Add new manufacturer information for test case 2.7-1 (Node Addressing test): Group address of DUT</p> <p>(2)</p> <p>Update manufacturer information "MOST Physical Layer" to consider MOST150 cPHY</p>
3V0E5_002	Para 3.2.1	Update Experimental setup to consider MOST150 cPHY nodes

Change Ref.	Section	Changes
3V0E5_003	2.1.0-1	<p>(1) Adapt: Responses will be only evaluated from actually tested FBlockID and FktID. At the receipt of the response it will be checked whether the InstID is correspondent. (In case of discrepancy: DUT not ok). → Responses will be only evaluated from actually tested FBlockID, InstID and FktID.</p> <p>(2) Update OPType 6 (M: StartResultAck): Add to "Permitted answer from DUT": "- Error(ErrorCode 0x04)"</p> <p>(3) Update OPType 7 (M: AbortAck): Add to "Permitted answer from DUT": "- Error(ErrorCode 0x04)"</p> <p>(4) Update OPType 8 (M: StartAck): Add to "Permitted answer from DUT": "- Error(ErrorCode 0x04)"</p>
3V0E5_004	2.1.0-2	<p>(1) Adapt: Responses will be only evaluated from actually tested FBlockID and FktID. At the receipt of the response it will be checked whether the InstID is correspondent. (In case of discrepancy: DUT not ok). → Responses will be only evaluated from actually tested FBlockID, InstID and FktID.</p> <p>2) Update OPType 6 (StartResultAck): Add to "Permitted answer from DUT": "- All Errors except 0x01,0x02,0x03,0x0A,0x0C "</p> <p>(3) Update OPType 7 (AbortAck): Add to "Permitted answer from DUT": "- All Errors except 0x01,0x02,0x03,0x0A,0x0C "</p> <p>(4) Update OPType 8 (StartAck): Add to "Permitted answer from DUT": "- All Errors except 0x01,0x02,0x03,0x0A,0x0C "</p>
3V0E5_005	2.3.2-2	Update test case so that it works without ET.ShutdownSuspendMode
3V0E5_006	All tests	Update test description (Device type) to consider MOST150 cPHY.

Changes MOST Core Compliance Test Specification Rev. 3.0E3 to MOST Core Compliance Test Specification Rev. 3.0E4. Changes based on ERRATA SHEET Rev. 3.0.4.

Change Ref.	Section	Changes
3V0E4_001	Document References	Update revisions of documents, based on MOST document release list
3V0E4_002	Para 1	Add clarification that this document is informative.
3V0E4_003	Para 3	Delete "The MOST Core Compliance Tests are based on the Core Functions, identified at paragraph "2 Identification of MOST Core Functions". "
3V0E4_004	Para 3.1.5	Add manufacturer information "MOST150 oPHY device: Performing optional RBD tests" For MOST150, RBD tests are optional. DUT manufacturer information will state if RBD tests are performed. [2.5-y (2)]
3V0E4_005	Para 3.1.6	Further Definitions - States Of DUT: DUT: NormalOperation; Effectuate state of DUT: Add in description to both ("DUT is slave device: " and "DUT is NWM master device: " - wait for t_WaitForApplication [GEN 5]
3V0E4_006	Para 3.2.1	Update of Experimental Set-up [GEN 6]
3V0E4_007	2.1.1-5	(1) Adapt Device type: All devices that are wakeable via MOST signal → All devices that are wakeable via MOST signal except TM and PM [2.1.1-5 (1)] (2) Adapt Start Condition: DUT in SleepMode (detectable by monitoring of power consumption). → DUT in SleepMode
3V0E4_008	2.3.2-2	Update flowchart and test description to check reaction on ShutDown.Start(Query) even if the DUT does not support ShutDown.Result(Suspend). [2.3.2-2 (1)]
3V0E4_009	2.3.2-3	Use timeout t_DeadlockShort while waiting for MOST Signal off. [2.3.2-3 (1)]
3V0E4_010	2.5-10	Adapt "Device type" and "Multi Node Device": All devices → All devices performing RBD [2.5-y (2)]
3V0E4_011	2.5-11	Adapt "Device type" and "Multi Node Device": All devices → All devices performing RBD [2.5-y (2)]
3V0E4_012	2.5-12	Adapt "Device type" and "Multi Node Device": All devices → All devices performing RBD [2.5-y (2)]

Change Ref.	Section	Changes
3V0E4_013	2.5-12	<p>(1) Adapt note to value of timer_1 to consider DelayT:</p> <p>a) DUT = TM: $\text{Timer_1} = (\text{t_DiagMaster_T1_max} + \text{t_DiagMaster_T2_max} + \text{t_DiagMaster_T3_max} + \text{t_DiagStart_max})$ \rightarrow $\text{Timer_1} = (\text{DelayTime} + \text{t_DiagMaster_T1_max} + \text{t_DiagMaster_T2_max} + \text{t_DiagMaster_T3_max} + \text{t_DiagStart_max})$</p> <p>b) DUT <> TM: $\text{Timer_1} = (\text{t_DiagSlave_max} + \text{t_DiagStart_max})$ \rightarrow $\text{Timer_1} = (\text{DelayTime} + \text{t_DiagSlave_max} + \text{t_DiagStart_max})$</p> <p>[2.5-12 (2)]</p> <p>(2) Test will be stopped after evaluation of inquiry: “NetBlock. RBDResult.Status received within t_DeadLockMid?”</p> <p>[2.5-12 (3)]</p>
3V0E4_014	2.6.2-3a	<p>Add note at inquiry “Receiving Configuration.Status(Ok) within t_DeadLockShort?”: DUT is allowed to send Configuration.Status(NotOk) or start a rescan before broadcasting Configuration.Status(Ok).</p> <p>[2.6.2-3a (1)]</p>
3V0E4_015	2.6.2-3b	Remove (a/b) from test results as test will be performed only once.
3V0E4_016	2.7-1	<p>Update flowchart to remove “GroupAddress.SetGet” and to consider blocking and unblocking broadcast.</p> <p>[2.7-1(1)]</p>
3V0E4_017	2.8.3-1a	<p>Delete note “Wait for manufacturer dependent time until the property is available.” This (t_WaitForApplication) is now covered by “Further Definitions - States Of DUT: NormalOperation; Effectuate state of DUT”</p> <p>[GEN 5]</p>
3V0E4_018	2.8.3-1b	<p>Delete note “Wait for manufacturer dependent time until the property is available.” This (t_WaitForApplication) is now covered by “Further Definitions - States Of DUT: NormalOperation; Effectuate state of DUT”</p> <p>[GEN 5]</p>
3V0E4_019	2.8.3-2	<p>Delete note “Wait for manufacturer dependent time until the property is available.” This (t_WaitForApplication) is now covered by “Further Definitions - States Of DUT: NormalOperation; Effectuate state of DUT”</p> <p>[GEN 5]</p>
3V0E4_020	2.8.3-7	<p>Delete note “Wait for manufacturer dependent time until the property is available.” This (t_WaitForApplication) is now covered by “Further Definitions - States Of DUT: NormalOperation; Effectuate state of DUT”</p> <p>[GEN 5]</p>
3V0E4_021	2.8.3-10	<p>Delete note “Wait for manufacturer dependent time until the property is available.” This (t_WaitForApplication) is now covered by “Further Definitions - States Of DUT: NormalOperation; Effectuate state of DUT”</p> <p>[GEN 5]</p>
3V0E4_022	Appendix 2	Update FBlock ET references to FB ET Rev. 3.0.3.

Changes MOST Core Compliance Test Specification Rev. 3.0E1 to MOST Core Compliance Test Specification Rev. 3.0E3. Changes based on ERRATA SHEET Rev. 3.0.3.

Change Ref.	Section	Changes
3V0E3_001	Para 2.1.4	Add sub-chapter Guideline „Improved Product ID Handling“ [GEN 2]
3V0E3_002	Para 2.1.5	Improve definition of "Multi Node Device" [GEN 3]
3V0E3_003	Para 3.1.8	Add sub-chapter "Preconditions" [GEN 4]
3V0E3_004	2.1.0-2	Add note: During this test case it must be considered that sending of an error is optional in function DTCP_Control. [2.1.0-2 (1)]
3V0E3_005	2.1.3-4	a) Delete value of interest "t_Restart" b) Delete in test description "DUT has to perform retries within t_Restart_max" c) Delete note: "Timeout depends on Retry: False: t_DeadLockMid True: t_Retry_max" d) Adapt Inquiry: Timeout → Timeout t_DeadLockMid <i>Reason: Restart is allowed to be performed after t_Restart_max.</i> [2.1.3-4 (1)]
3V0E3_006	2.2.1-4	(1) Adapt inquiry: (receiving MOST signal) or (t_test > t_Restart_max)? → (receiving MOST signal) or (t_test > t_DeadLockShort)? <i>Reason: Restart is allowed to be performed after t_Restart_max.</i> [2.2.1-4 (1)]
		(2) Adapt inquiry: t_Restart_min < t_test < t_Restart_max? → t_Restart_min < t_test? <i>Reason: Restart is allowed to be performed after t_Restart_max.</i> [2.2.1-4 (1)]
3V0E3_007	2.2.1-6	(1) Adapt Device type: All devices except Multi Node Device (Nodes in MultiNode Device behind TM) → All devices, but in Multi Node Devices only the first node. [2.2.1-6 (1)]

Change Ref.	Section	Changes
3V0E3_008	2.5-11	<p>a) Adapt inquiry Result = "Rel. node pos =0"? → Result = "Rel. node pos with respect to the ring break"?</p> <p>b) Adapt Device type: All devices → All devices, but in MultiNode only the first node.</p> <p>[2.5-11 (1)]</p>
3V0E3_009	2.5-12	<p>Adapt test description: node position =1 → Relative node position with respect to the ring break</p> <p>[2.5-12 (1)]</p>
3V0E3_010	2.6.2-6	<p>Adapt note: Tester must not process FBlockIDs.SetGet from DUT. It has to continue using the same (old) InstID. → Both testers answer regularly to requests to their NetBlocks without changing their FBlock configuration.</p> <p><i>Reason: To avoid change of test focus which does not cover a check for "discard device" during compliance testing.</i></p> <p>[2.6.2-6 (1)]</p>
3V0E3_011	2.6.2-7	<p>Adapt note: Before broadcasting Configuration.Status(Ok), the DUT must not broadcast any other Configuration.Status message. → Before broadcasting Configuration.Status(Ok), the DUT must not broadcast Configuration.Status(NewExt / New / Invalid).</p> <p><i>Reason: Before broadcasting Configuration.Status(Ok), a broadcast of Configuration.Status(NotOk) may occur, it is only forbidden to broadcast Configuration.Status(NewExt / New / Invalid).</i></p> <p>[2.6.2-7 (1)]</p>
3V0E3_012	2.6.4-9	<p>a) Delete: Exception: DUT is allowed to send debug messages (to 0x0FF0).</p> <p>b) Adapt Inquiry: Received any control message from DUT? → Received NetBlock.FBlockIDs.Status(FBlockIDList) from DUT?</p> <p><i>Remark: DUT manufacturer list with list of FBlocks FBlocks ("special applications") seems to be not very practical.</i></p> <p>[2.6.4-9 (1)]</p>
3V0E3_013	2.8.4-1	<p>Timer Values for t_WaitForNextSegment are: min. 4950 typ. 5000 max. 10150 ms.</p> <p><i>Reason: WG DA decision (D112_10): "The timer t_WaitForNextSegment describes the timer within the device until garbage collection is started. Because this device behavior cannot be observed directly, the resulting segmentation error message is evaluated. The timing tolerance needs to be considered twice. Old values (4975 5000 10015 ms) too narrow.</i></p> <p>[2.8.4-1 (1)]</p>

Change Ref.	Section	Changes
3V0E3_014	2.8.4-7	<p>Add note:</p> <p>The tester shall not use TelID 0x4.</p> <p><i>Reason:</i></p> <p><i>In this test case the tester requests the buffer sizes of the DUT by sending the message ET.MessageBufSize.Get(). If the DUT returns "MessageLengthRx > 0", the tester sends the message ET.EchoMessage() to the DUT with length greater than MessageLengthRx to force a segmentation error.</i></p> <p><i>On some devices the test case would fail if the tester used "TelID 0x4" when sending the message ET.EchoMessage() with length > MessageLengthRx.</i></p> <p><i>Background:</i></p> <p><i>Receiving a message that starts with TelID 0x4, the size information in this telegram is used to allocate the RX buffer. But in this case the size is not limited to the length "MessageLengthRx" returned in ET.MessageBufSize.Status(). The DUT is usually able to receive the complete message, without returning segmentation error 0x2.</i></p> <p>[2.8.4-7 (1)]</p>
3V0E3_015	3.1-1	<p>Add note:</p> <p>„In cases where the bandwidth of sink cannot be determined because of the data type (ContentType 0xC0 ... 0xEF System Integrator Specific and 0xF0 ... 0xFE Supplier specific), the test case is skipped for this sink.”</p> <p>[3.1-x (1)]</p>
3V0E3_016	3.1-3	<p>Add note:</p> <p>„In cases where the bandwidth of sink cannot be determined because of the data type (ContentType 0xC0 ... 0xEF System Integrator Specific and 0xF0 ... 0xFE Supplier specific), the test case is skipped for this sink.”</p> <p>[3.1-x (1)]</p>
3V0E3_017	3.1-4	<p>Add note:</p> <p>„In cases where the bandwidth of sink cannot be determined because of the data type (ContentType 0xC0 ... 0xEF System Integrator Specific and 0xF0 ... 0xFE Supplier specific), the test case is skipped for this sink.”</p> <p>[3.1-x (1)]</p>
3V0E3_018	3.1-5	<p>Add note:</p> <p>„In cases where the bandwidth of sink cannot be determined because of the data type (ContentType 0xC0 ... 0xEF System Integrator Specific and 0xF0 ... 0xFE Supplier specific), the test case is skipped for this sink.”</p> <p>[3.1-x (1)]</p>
3V0E3_019	3.1-6	<p>Add note:</p> <p>„In cases where the bandwidth of sink cannot be determined because of the data type (ContentType 0xC0 ... 0xEF System Integrator Specific and 0xF0 ... 0xFE Supplier specific), the test case is skipped for this sink.”</p> <p>[3.1-x (1)]</p>
3V0E3_020	3.2-14	<p>Adapt:</p> <p>SourceActivity.ErrorAck (SenderHandle='0x01', ErrorCode='0x03', ErrorInfo=_) → SourceActivity.ErrorAck (SenderHandle='0x01', ErrorCode='_', ErrorInfo=_)</p> <p><i>Reason: Improvement; there is no distinction of ErrorCode necessary.</i></p> <p>[3.2-14 (1)]</p>
3V0E3_021	Appendix 3	Move Document History to Appendix 3

Changes MOST Core Compliance Test Specification Rev. 3.0 to MOST Core Compliance Test Specification Rev. 3.0E1. Changes based on ERRATA SHEET Rev. 3.0.2.

Change Ref.	Section	Changes
3V0E1_001	para 3.1.5	<p>Introduce DUT manufacturer inputs regarding to RBD</p> <p>(a) Support of RBD Phase 3</p> <p>(b) Manufacturer dependent time interval when ET.DiagResult is available. Information is only necessary in case RBD phase 3 is not supported. The time interval shall be given with respect to the end of RBD phase 2.</p> <p>[2.5-y (1)]</p>
3V0E1_002	para 3.1.6	<p>Revise description of "DUT: Netinterface Off": Delete "- Trigger AutoWakeUp if required and applicable"</p> <p>[GEN 1]</p>
3V0E1_003	2.2.1-5	<p>(1) Adapt inquiry: (receiving MOST signal) or (t_test > t_Restart_max)? --> (receiving MOST signal) or (t_test > t_DeadLockShort)? Reason: Restart is allowed to be performed after t_Restart_max.</p> <p>[2.2.1-5 (1)]</p>
		<p>(2) Adapt inquiry: t_Restart_min < t_test < t_Restart_max? --> t_Restart_min < t_test? Reason: Restart is allowed to be performed after t_Restart_max.</p> <p>[2.2.1-5 (2)]</p>
		<p>(3) Remove Note "Note: If DUT is PM, t_Restart_max must not be considered." Reason: t_Restart_max no longer considered by test case</p> <p>[no reference in errata]</p>
3V0E1_004	2.2.1-7	<p>(1) At the beginning of the test, the ShutdownReason of the DUT shall be set to the defined value "No result available".</p> <p>[2.2.1-7 (1)]</p>
		<p>(2) Adapt: "Interrupt ring between Tester 1 and Tester 2" --> "Interrupt ring between Tester 2 and Tester 1"</p> <p>[2.2.1-7 (2)]</p>
3V0E1_005	2.3.1-3	<p>Add note: Pre-conditions have to be switched off that could prevent DUT from performing ShutDown</p> <p>[2.3.1-3 (1)]</p>
3V0E1_006	2.3.3-1	<p>Add note: The Trigger AutoWakeUp must not be applied.</p> <p>[2.3.3-1 (1)]</p>

Change Ref.	Section	Changes
3V0E1_007	2.4.1-1	a) Adapt: Start (t_Config_max + t_Restart_max) --> Start t_DeadLockMid b) Adapt inquiry: Timeout (t_Config_max + t_Restart_max) --> Timeout t_DeadLockMid [2.4.1-x (1)]
3V0E1_008	2.4.1-2	Test case is not applicable for MOST50 ePhy as ET.Autowakeup is not supported. [2.4.1-2 (1)] a) Adapt: Start (t_Config_max + t_Restart_max) --> Start t_DeadLockMid b) Adapt inquiry: Timeout (t_Config_max + t_Restart_max) --> Timeout t_DeadLockMid [2.4.1-x (1)]
3V0E1_009	2.5-2	Adapt test step: Close the ring, turn on MOST signal and wait until DUT switches on its MOST signal within t_diag_restart_max --> Close the ring, turn on MOST signal and wait until DUT switches on its MOST signal within (t_diag_restart_max + t_restart_max) [2.5-x (1)]
3V0E1_010	2.5-3	Adapt test step: Close the ring, turn on MOST signal and wait until DUT switches on its MOST signal within t_diag_restart_max --> Close the ring, turn on MOST signal and wait until DUT switches on its MOST signal within (t_diag_restart_max + t_restart_max) [2.5-x (1)]
3V0E1_011	2.5-7	(a) t_Diag_Restart_min <= t_test <= t_Diag_Restart_max --> t_Diag_Restart_min + t_Restart_min <= t_test <= t_Diag_Restart_max + t_Restart_max (b) DUT not ok: DUT fail to meet t_Diag_Restart after RBD --> DUT not ok: DUT fails to meet (t_Diag_Restart + t_Restart) after RBD [2.5-7 (1)] Change value of interest: t_Diag_Restart --> (t_Diag_Restart + t_Restart) [no reference in errata]
3V0E1_012	2.5-10	In case DUT does not report RBD result at end of RBD phase 2, wait for t_DiagResultAvailable before requesting DiagResult via FB ET [2.5-y (1)]
3V0E1_013	2.5-11	In case DUT does not report RBD result at end of RBD phase 2, wait for t_DiagResultAvailable before requesting DiagResult via FB ET [2.5-y (1)]

Change Ref.	Section	Changes
3V0E1_014	2.5-12	In case DUT does not report RBD result at end of RBD phase 2, wait for t_DiagResultAvailable before requesting DiagResult via FB ET [2.5-y (1)]
3V0E1_015	2.6.2-3b	The test has to be performed only once. Delete part a) "a) Tester always responds with NAK during the loops (receive buffer locked). The first message received by the tester may be acknowledged (due to potential empty receive buffer of the tester). During the test, the LowLevel and MidLevel retries of the DUT are not considered. The manufacturer has to provide information about number and timing of LowLevel and MidLevel retries to allow tester to filter them out" [2.6.2-3b (1)]
3V0E1_016	2.8.3-1a	Add note at inquiry "Notification Matrix empty?": Each notified function of each FBlock of DUT must be checked. [2.8.3-x (1)]
3V0E1_017	2.8.3-1b	Add note at inquiry "Notification Matrix empty?": Each notified function of each FBlock of DUT must be checked. [2.8.3-x (1)]
3V0E1_018	2.8.3-7	(1) Add note at inquiry "Double entries detected?": No entry of tester 1 is tolerated and does not lead to NotOK (3). Reason: Clarification necessary to avoid ambiguous understanding [2.8.3-7 (1)]
		(2) 1.) Adapt command box: Start t_Property --> Start t_WaitforProperty 2) Adapt Inquiry t_Property expired? --> t_WaitforProperty expired? [2.8.3-7 (2)]
		Change value of interest: t_Property --> t_WaitforProperty [no reference in errata]
		Add note at inquiry "Notification Matrix empty?": Each notified function of each FBlock of DUT must be checked. [2.8.3-x (1)]
3V0E1_019	3.1-1	Adapt: Delete part with StreamDataInfo.Get Replace last exception: "The length field of the Short Stream "SinkNrList" is not equal to value of variable "SinkCount" " --> "DUT manufacturer information is not equal to value of variable "SinkCount" ". [3.1-1 (1)]
		Value of interest / test focus: remove "StreamDataInfo". [no reference in errata]

Change Ref.	Section	Changes
3V0E1_020	3.1-3	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
3V0E1_021	3.1-4	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
3V0E1_022	3.1-5	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
3V0E1_023	3.1-6	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
3V0E1_024	3.2-1	Adapt: Delete part with StreamDataInfo.Get Replace last exception: "The length field of the Short Stream "SourceNrList" is not equal to value of variable "SourceCount" " --> "DUT manufacturer information is not equal to value of variable "SourceCount" ". [3.2-1 (1)]
		Value of interest / test focus: remove "StreamDataInfo.Status". [no reference in errata]
3V0E1_025	3.2-3	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
		Add note to test description: Already allocated sources should be deallocated before sending Allocate.StartResultAck to a new SourceNr. [3.2-3 (1)]
3V0E1_026	3.2-4	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]

Change Ref.	Section	Changes
3V0E1_027	3.2-5	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
		Allocate.ErrorAck(SenderHandle='0x01', ErrorCode='0x20',ErrorInfo='SourceNr, BlockWidth') --> Allocate.ErrorAck(SenderHandle='0x01', ErrorCode='0x20',ErrorInfo='_') [3.2-5 (1)]
		Delete last exception and "DUT not ok(2)" as neither SourceNr nor BlockWidth evaluated anymore. [no reference in errata]
3V0E1_028	3.2-6	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
3V0E1_029	3.2-7	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
3V0E1_030	3.2-8	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
		Delete note: "If DUT responds with xxx.ProcessingAck, the tester has to repeat xxx.StartResultAck after t_CM_DeadLockPrev (max. 10 retries)" [3.2-x (1)]
3V0E1_031	3.2-9	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
		Delete note: "If DUT responds with xxx.ProcessingAck, the tester has to repeat xxx.StartResultAck after t_CM_DeadLockPrev (max. 10 retries)" [3.2-x (1)]
3V0E1_032	3.2-14	t_CM_DeadLockPrev --> t_DeadLockShort [3.x (1)]
		Change value of interest: Delete t_CM_DeadLockPrev [no reference in errata]
		Delete note: "If DUT responds with xxx.ProcessingAck, the tester has to repeat xxx.StartResultAck after t_CM_DeadLockPrev (max. 10 retries)" [3.2-x (1)]

Change Ref.	Section	Changes
3V0E1_033	Appendix 1	Test case 2.5-7: t_test (t_Diag_Restart) --> t_test (t_Diag_Restart + t_Restart) [Appendix 1]
		Remove all references to t_CM_DeadLockPrev [no reference in errata]

Changes MOST Core Compliance Test Specification Rev. 1.3 to MOST Core Compliance Test Specification Rev. 3.0

Change Ref.	Section	Changes
3V0_001	2.2.1 2.1.5 3.1	Reference to "secondary node " deleted
3V0_002	2.2.1-5	- Set Shutdown Flag at beginning of test (→ t_SSO_Shutdown not to be checked) Note added: "Test case "critical unlock without shutdown flag set" is covered by test case 2.2.1-6 Sudden Signal Off test."
3V0_003	2.3.1-3	PM has to set Shutdown Flag if performing ShutDown (DUT not ok (4)) Replaced t_Suspend by t_WaitSuspend ShutDown.Start(Query) → ShutDown.StartAck(Query) ShutDown.Start(Execute) → ShutDown.StartAck(Execute)
3V0_004	2.1.2-4	Test case deleted (t_Boundary removed from specification)
3V0_005	2.1.2-5	Test case deleted (In MOST3, MOST25 specific description of stable lock detection removed) Revised by 3V0_109: Test case re-introduced (change ref. 3V0_109)
3V0_006	2.1.2-6	Test case deleted
3V0_007	2.8.1-1	Test case deleted
3V0_008	2.1.1-2	Test case deleted and replaced by new test case 2.1.1-6 (system lock flag test)
3V0_009	2.2.1-4	Shutdown Flag set at beginning of test Changes of t_test_min / t_test_max necessary according to new value of t_Lock_Min
3V0_010	2.3.1-6	ShutDown.Result(Suspend) → ShutDown.ResultAck(Suspend) ShutDown.Start(Query) → ShutDown.StartAck(Query) ShutDown.Start(Execute) → ShutDown.StartAck(Execute)
3V0_011	2.3.3-5	ShutDown.Result(TemperatureShutDown) → ShutDown.ResultAck(0x03) (revised by 3V0_191)
3V0_012	2.3.2-2	ShutDown.Result(Suspend) → ShutDown.ResultAck(Suspend) ShutDown.Start(Query) → ShutDown.StartAck(Query) t_Suspend_typical no longer exists → t_Suspend_max (revised by 3V0_192)
3V0_013	2.3.2-3	ShutDown.Start(Execute) → ShutDown.StartAck(Execute) (revised by 3V0_193)
3V0_014	2.3.3-1	Value of interest: "Supply voltage" → "Undervoltage Management"
3V0_015	2.3.3-4	NetBlock.Shutdown.Result(0x03) → NetBlock.Shutdown.ResultAck(0x03) g_dead → g_critical Devcie type: All devices except PowerMaster
3V0_016	2.4.1-9	Configuration.Status(New) → Configuration.Status(NewExt)

Change Ref.	Section	Changes
3V0_017	2.5-2	Test case deleted. New test cases defined for RBD Revised by 3V0_124: Test case reactivated for MOST50 ePHY devices (change ref. 3V0_124)
3V0_018	2.5-3	Test case deleted. New test cases defined for RBD Revised by 3V0_125: Test case reactivated for MOST50 ePHY devices (change ref. 3V0_125)
3V0_019	2.5-7	Test case deleted. New test cases defined for RBD Revised by 3V0_126: Test case reactivated for MOST50 ePHY devices (change ref. 3V0_126)
3V0_020	2.5-8	Test case deleted. New test cases defined for RBD Revised by 3V0_127: Test case reactivated for MOST50 ePHY devices (change ref. 3V0_127)
3V0_021	2.6.2-7	Configuration.Status(New) → Configuration.Status(NewExt)
3V0_022	2.6.4-3	Configuration.Status(New) → Configuration.Status(NewExt)
3V0_023	2.8.3-1a	Configuration.Status(New) → Configuration.Status(NewExt)
3V0_024	2.8.3-1b	Configuration.Status(New) → Configuration.Status(NewExt)
3V0_025	3.2-10	Test case deleted (SourceConnect was MOST25 specific)
3V0_026	3.2-11	Test case deleted (SourceConnect was MOST25 specific)
3V0_027	3.2-12	Test case deleted (SourceDisconnect was MOST25 specific)
3V0_028	3.2-13	Test case deleted (SourceDisconnect was MOST25 specific)
3V0_029	3.2-3	MOST25 specific part of test case deleted
3V0_030	3.2-4	MOST25 specific part of test case deleted
3V0_031	3.2-5	MOST25 specific part of test case deleted
3V0_032	3.2-6	MOST25 specific part of test case deleted
3V0_033	3.2-7	MOST25 specific part of test case deleted
3V0_034	3.2-8	MOST25 specific part of test case deleted
3V0_035	3.2-9	MOST25 specific part of test case deleted
3V0_036	2.8.3-2	Deleted Note from flowchart: "Note: Test has to be performed with every single FBlock of the DUT."
3V0_037	3.0-1	Added note to case "DUT not ok (2)": If DUT reports sink / source relevant FktIDs, the manufacturer has to indicate that the relevant FBlock contains sink / source (and vice versa) with the following exception: In case functions are only implemented in principle and source resp. sink count is zero the result is set correspondingly: DUT ok (1): FBlock neither contains sink nor source according to manufacturer info or DUT ok (2): FBlock contains source according to manufacturer info or DUT ok (4): FBlock contains sink according to manufacturer info In that case, result of all relevant sink / source tests will be set to "DUT ok".
3V0_038	2.5-10	New test case "RBD closed ring test"
3V0_039	2.5-11	New test case "RBD interruption before DUT test"
3V0_040	2.5-12	New test case "RBD interruption behind DUT test"

Change Ref.	Section	Changes
3V0_041	2.5-13	New test case "RBD all slave / multimaster test" (revised by 3V0_154)
3V0_042	2.1.1-6a	New test case "System Lock Flag test (a)"
3V0_043	2.1.1-6b	New test case "System Lock Flag test (b)"
3V0_044	2.2.1-6	New test case "Sudden signal off test"
3V0_045	2.2.1-7	New test case "Shutdown flag preset test"
3V0_046	3.0-1	Changed note "The test has to be performed to identify whether the DUT contains sink or source" → "The test has to be performed to identify whether the DUT contains sink or source for streaming data (synchronous and isochronous)." Reason: Isochronous data also covered by sink / source tests. (revised by 3V0_085)
3V0_047	2.6.2-3b	Test case updated: First 10 loops has to be use t_DelayCfgRequest1 Last 3 loops has to be use t_DelayCfgRequest2 Note added: "Test does not focus on border between 20. check and 21. check. Test only checks whether DUT changes from short check interval to long check interval."
3V0_048	2.6.2-4b	Test case updated: In case the CR is full while new FBlocks are indicated by tester2, the DUT has to broadcast Configuration.Status(NewExt, <empty list>). The new indicated FBlocks must not be stored into the CR. (Note at end of flow chart updated accordingly)
3V0_049	2.1.0-1	Added note: "All commands from DUT (OPType 0x0 .. 0x8) will be ignored for all FBlocks". "For Netblock and NMW FBlock, OPTypes 0x09 ... 0xF not tested."
3V0_050	2.1.0-2	Added note: "All commands from DUT (OPType 0x0 .. 0x8) will be ignored for all FBlocks" "For Netblock and NMW FBlock, OPTypes 0x09 ... 0xF not tested."
3V0_051	2.1.1-1	Flowchart updated: Added "Wait for 10 sec." between "DUT in SleepMode" and "Generate wake-up event" (revised by 3V0_153)
3V0_052	2.1.3-1	Replaced t_manufacturer_0x00 with t_DeadLockMid
3V0_053	2.1.3-4	Replaced t_manufacturer_0x00 with t_DeadLockMid
3V0_054	2.1.1-4	Replaced t_manufacturer_0x00 with t_DeadLockMid
3V0_055	2.4.1-2	Note added in flowchart: "In order to assure that wakeup duration has expired, the tester is allowed to wait additionally for t_DeadLockMid."
3V0_056	2.4.1-9	t_WaitAfterNCE_max replaced by t_DeadLockMid
3V0_057	2.4.2-3	Test result updated: DUT not ok (1): DUT disturbs communication before U_Low is reached or before MOST signal is switched off. (revised by 3V0_176)

Change Ref.	Section	Changes
3V0_058	2.6.2-3b	Flowchart updated: a) After wakeup, wait for t_DeadLockMid; Tester1: open bypass. b) $(t_WaitForAnswer_min + t_DelayCfgRequest1_min) \leq t_test \leq (t_WaitForAnswer_max + t_DelayCfgRequest1_max)$? " → $t_DelayCfgRequest1_min \leq t_test \leq (t_WaitForAnswer_max + t_DelayCfgRequest1_max)$?"
3V0_059	2.6.2-4b	Note added: "Tester has to wait for manufacturer dependent time before requesting CR."
3V0_060	2.6.4-10	Exception for Netblock and FBET
3V0_061	2.6.4-10	Flowchart updated: "Received FkIds.Status or FkIds.Error from DUT?" → "Received FkIds.Status from DUT?"
3V0_062	2.8.3-2	a) Exception for Netblock b) Flowchart updated: At the end of the test, the DUT is allowed to respond empty list or any NotificationCheck error
3V0_063	2.8.4-2	Added note: In case the MessageBufSize is dynamic (indicated by value 0) the length to be sent by the tester is 44 bytes (4 segments)
3V0_064	3.1-3 3.1-4 3.1-5 3.1-6 3.1-8	Added note: "In case where the bandwidth of sink cannot be determined because of the data type, the test case is skipped for this sink."
3V0_065	3.1-3 3.1-4	Added note to disconnect sink before connect to avoid resource overflow
3V0_066	3.2-6	MSC updated: Wait for t_DeadLockShort after freeing bandwidth
3V0_067	Appendix 3	Deleted
3V0_068	Document References	FBET 2V5 → FBET 2V6 and higher
3V0_069	Introduction	"This Core Compliance Test Specification is based on MOST Specification 3V0 and requires FBlock ET 2.5" → "This Core Compliance Test Specification is based on MOST Specification 3V0 and requires FBlock ET 2.6 or higher"
3V0_070	Para 2.1.2	FBET V2.5 → FBET 2V6 Reference to OASIS Netservices deleted
3V0_071	2.8.3-1b	Flowchart updated (left branch): 1. send Configuration.Status(NotOk) to DUT 2. Request Notification Matrix from DUT → 1. send Configuration.Status(NotOk) to DUT 2. wait for t_DeadLockShort 3. send Configuration.Status(OK) to DUT 4. Request Notification Matrix from DUT"
3V0_072	2.6.4-9	Removed reference to "deallocate.all" from test case Flowchart updated: "Received any message from DUT?" → "Received any control message from DUT?"

Change Ref.	Section	Changes
3V0_073	2.2.1-5	Added note in flowchart: "Length on unlock has to exceed t_Unlock_max to ensure DUT detects critical unlock." Flowchart updated (Sequence inserted): "Stop generating MOST signal" "Delete Shutdown Flag"
3V0_074	2.3.3-4	Added note to flowchart: "DUT is allowed to use any SenderHandle"
3V0_075	Para 1	Deleted "This Core Compliance Test Specification is based on MOST Specification 3V0 and requires FBlock ET 2.6 or higher. Appendix 4 shows relevant test cases which are applicable for MOST Specification 2V4, 2V3 and 2V2."
3V0_076	Appendix 1	Measurement uncertainty of T_SSO_Shutdown: +-5ms
3V0_077	Doc. Version	Changed Document Version: 1V4 → 3V0
3V0_078	2.3.1-3	Added notes to flowchart: "DUT is allowed to use any SenderHandle" "Use SenderHandle, provided by DUT"
3V0_079	2.3.1-6	Testdescription updated: "The test is continued from 2.3.1-2" → "The test is continued from 2.3.1-3" Added notes to flowchart: "DUT is allowed to use any SenderHandle" "Use SenderHandle, provided by DUT during test case 2.3.1-3"
3V0_080	2.3.2-2	Tester has to use SenderHandle 0x01
3V0_081	2.3.2-3	Tester has to use SenderHandle 0x01
3V0_082	2.3.3-5	Tester has to use SenderHandle 0x01
3V0_083	2.8.3-2	Note updated: "The test has to be performed with every single registered FBlock of the DUT except Netblock" → "The test has to be performed with every single registered FBlock of the DUT"
3V0_084	para 3.1	Section "Manufacturer timings" deleted
3V0_085	3.0-1	Isochronous data not covered by sink / source tests
3V0_086	2.1.1-6b	Flowchart updated: "Stable lock?" → "Tester1 detects stable lock?"
3V0_087	2.2.1-6	Flowchart: Query "Shutdown Flag set within t_DeadLockMid" defined more clearly. Tester2 has to detect Shutdown Flag. Update of experimental setup according to MasterMode / SlaveMode of Tester1.
3V0_088	2.2.1-7	Flowchart: Wait for t_SSO_Shutdown after setting of Shutdown Flag
3V0_089	2.6.2-3b	Use experimental set-up #2: - Tester 1 in slave mode; bypass closed - Tester 2 in slave mode or in master mode (depends on DUT) behaves like normal MOST device Tester2 has to prevent potential shutdown during test. It has to send ShutDown.ResultAck(..., Suspend) to the PM in case receiving ShutDown.StartAck(..., Query). Flowchart updated: Tester1 opens bypass after t_WaitForApplication

Change Ref.	Section	Changes
3V0_090	Para 3.1	New manufacturer input (for test case 2.6.2-3b): Delay between "NetOn" and availability of application (= t_WaitForApplication)
3V0_091	2.6.4-10	Flowchart: "Response war Error" → "Received FktIDs.Error" "Note: In case DUT = NWM: DUT has to respond with its own InstID (In case this InstID is 0x00; DUT is allowed to use 0x00)."
3V0_092	All Sink / Source tests	SyncDataInfo → StreamDataInfo DataType → ContentType DataDescription → ContentDescription SourceCount → SourceNrList SinkCount → SinkNrList Connections → Channels
3V0_093	3.2-8	Test result "DUT not ok(3): DUT used wrong Label during ResourceDeAllocate" deleted
3V0_094	2.6.2-4b	Test case updated; not extended memory management according to CR (independent memory for every node) will be detected and test will be finished with "DUT ok (2)" (revised by 3V0_151)
3V0_095	2.5-10 2.5-11 2.5-12 2.5-13	Added note: "Tester must not switch on MOST signal before DUT switches on MOST signal (= DUT has to start with RBD procedure before tester starts with RBD procedure). To avoid deadlocks, test can be stopped in case of DUT fails to switch on MOST signal within t_Diag_Start_max + t_Diag_Signal_max".
3V0_096	2.5-12	Use experimental set-up #2
3V0_096	Para 3.1	Procedure "Perform Wake-up": Replaced "5. Wait for SBC > 5" by "5. Wait for System Lock Flag set"
3V0_097	Document References Para 2.1.2	Referring to MOST FunctionBlock "Enhanced Testability": Change from "2V6 and higher" to "3V0"
3V0_098	Appendix 2	Reference to FB ET function 0x214 (ManufacturerTimings) deleted.
3V0_099	2.6.4-9	Flowchart: Exception added: "DUT is allowed to send debug messages (to 0x0FF0)"
3V0_100	2.6.4-9	„Tester (NWM) must not send FBlockIDs.Get to DUT" → "Tester (NWM) must not send any message to DUT"
3V0_101	2.4.1-1	Flowchart: Updated Note within loop: „(3) MOST signal on" → „(3) MOST signal off"
3V0_102	All Notification Tests	Wording improved:: Changed "... with every registered FBlock" → "... with every FBlock, registered in the CR..."
3V0_103	2.2.1-6	Flowchart updated: Timeout for Shutdown-Flag: T_DeadLockMid → t_ShutDown_max
3V0_104	Para 3.1	States of DUT: Normal Operation „MOST Spec 2V5" → „MOST Spec 3V0"
3V0_105	2.1.1-6a	Typo in Note: "then" → "when"
3V0_106	2.3.2-2	Typo in Experimental set-up
3V0_107	2.6.2-4b	Flowchart: Typo in Note at „DUT not ok(1): „ist" → „its"
3V0_108	2.2.1-4 2.2.1-5	Unlock Tests changed according to MCTG forum proposal "Unlock tests (2.2.1-4; 2.2.1-5) without shutdown flag preset"

Change Ref.	Section	Changes
3V0_109	2.1.2-5	Slave Lock Detection Test re-introduced again according to MCTG proposal "Reactivation of Test Case 2.1.2-5 (Slave lock detection test) in MOST Core Compliance Test Spec 3V0"
3V0_110	2.8.4-7	Adapt note in flowchart: "The size of the message must not exceed the max. supported buffer size of the DUT." → "In case of fix buffer size, the size of the message must not exceed the max. supported buffer size of the DUT."
3V0_111	2.8.4-7	Adapt note in flowchart: "In case of dynamic buffer size, Segmentation Error (0x02) is also allowed. In this case the test must be aborted to avoid subsequent errors." → "In case of dynamic buffer size, Segmentation Error (0x02) or (0x04) are also allowed. In this case the test must be aborted to avoid subsequent errors."
3V0_112	2.8.4-2	Adapt note in flowchart: "Trigger the segmented message via FBlock ET (EchoMessage). Depending on the buffer size given by FB ET the message consists of 2, 3 or 4 segments: "size = 13 .. 22" : 2 segments "size = 23 .. 33" : 3 segments "size > 33" : 4 segments In case the MessageBufSize is dynamic (indicated by value 0) the length to be sent by the tester is 44 bytes (4 segments)" → "MOST50 ePHY: Trigger the segmented message via FBlock ET (EchoMessage). Depending on the buffer size given by FB ET the message consists of 2, 3 or 4 segments: "size = 13 .. 22" : 2 segments "size = 23 .. 33" : 3 segments "size > 33" : 4 segments In case the MessageBufSize is dynamic (indicated by value 0) the length to be sent by the tester is 44 bytes (4 segments) MOST150 oPHY: Depending on the buffer size given by FB ET the message consists of 2, 3 or 4 segments: In case the MessageBufSize is dynamic (indicated by value 0) the tester has to send 4 segments."
3V0_113	all	Experimental setup #1 deleted. Only one experimental setup exists, based on setup #2. All test cases, based on setup #1 have been updated with a note "Tester 2 in spy mode".
3V0_114	3.2-6	Test focus updated
3V0_115	all	Replaced wording "before DUT" → "in front of DUT"
3V0_116	2.8.4-1	"value of interest" updated: t_Property replaced by t_WaitForNextSegment; Flow chart updated: DUT has to respond "Segmentation Error(05)" within t_WaitForNextSegment(min/max)
3V0_117	Appendix 4	T_WaitForNextSegment covered by measurement uncertainty table (+-20ms)
3V0_118	2.8.4-2	Note in Flow chart updated: "Telegram type has to be correct (1,2,2,2,3)" → "Sequence of TelIDs has to be correct (for example 1,2,2,2,3). Replaced "5 seconds" with t_WaitForNextSegment_max Use ET.SendMessage instead of ET.EchoMessage to trigger DUT sending segmented message
3V0_119	2.2.1-4	Last loop of flow chart: Check for "MOST signal off" deleted to enable test case for MOST50 ePHY; Note added: "MOST50 ePHY: Tester2: Check for „MOST signal off" (instead of Shutdown Flag)"
3V0_120	2.3.2-2	Node added: "If test will end with "DUT ok (1)" or "DUT ok (2)", continue with test case 2.3.2-3 (ShutDown.Start(Execute) test)."
3V0_121	2.3.2-3	Flow chart: "Send ShutDown.StartAck(0x01, Query)" and "Wait for t_Suspend_max" inserted at beginning to prepare DUT for shutdown.
3V0_122	all	Added Note to all flow charts that checks CapabilityToWake of DUT: Use NetBlock.DeviceInfo.Get(CapabilityToWake) of DUT to check CapabilityToWake.

Change Ref.	Section	Changes
3V0_123	all	Add indication for every test to be applicable for: [x] MOST150 oPHY [x] MOST50 ePHY (based on MOST3V0 Addendum A – MOST50 Adaption)
3V0_124	2.5-2	Test case reactivated for MOST50 ePHY devices
3V0_125	2.5-3	Test case reactivated for MOST50 ePHY devices
3V0_126	2.5-7	Test case reactivated for MOST50 ePHY devices
3V0_127	2.5-8	Test case reactivated for MOST50 ePHY devices
3V0_128	2.3.2-3	"Wait for t_Suspend_max" replaced by "wait for t_WaitSuspend_min"
3V0_129	2.3.3-5 2.3.2-2 2.3.2-3	SenderHandle, used by tester: 0x01 → 0x1234
3V0_130	2.2.1-4	Test case updated: - At the end of the test, "Most signal off" will be tested - Separate test cases for MOST150 and MOST50 Partially revised by 3V0_219: Test case for MOST50 ePHY deleted
3V0_131	2.2.1-5	Separate test cases for MOST150 and MOST50 Revised by 3V0_219: Test case for MOST50 ePHY deleted
3V0_132	2.8.3-2	Add note at inquiry "DUT responds with list, containing all functions that support notification?": "Note: If the FBlock does not contain any functions that support notification, the DUT is allowed to respond with empty list or any NotificationCheck error."
3V0_133	2.3.1-3	Flowchart updated: Time to switch off MOST signal: t_ShutDownWait → t_ShutDownWait + t_SSO_Shutdown (only applicable for MOST150 oPHY devices)
3V0_134	3.2-5	"Allocate all availabel channels. Every channel has to be allocated separately" → "Allocate all availabel bandwidth. Every connection has to be allocated separately."
3V0_135	2.7-1	Flowchart updated: "Send message to..." → " Send GroupAddress.Get to ..."
3V0_136	Document References	General FBlock 3V0 → 3V0.1
3V0_137	3.1-1	Update of last inquiry; adapted to wording of General FBlock V3.0.1 (The length field of the Short Stream "SinkNrList" corresponds to the amount of sink numbers.)
3V0_138	2.5-11	Flowchart updated (last inquiry); based on NetBlock: Result = "Position detected; Pos=0"? → RBDStatus = "No activity"?
3V0_139	Document References	MOST Specification 3V0 → 3V0E1
3V0_140	2.3.3-5	Flowchart updated: Combined both last inquiries within one loop
3V0_141	General Notes; States of DUT	DUT: Normal Operation; Detect state of DUT: "b) DUT is slave device" → "b) DUT is NWM device"
3V0_142	General Notes; States of DUT	DUT: SleepMode; Detect state of DUT: Mechanism to detect sleep mode revised; Consideration of MOST50 ePHY
3V0_143	3.1-5	Deleted part Mute.Get as there is no requirement in MOST Specification that "muting" and "disconnect" correspond.
3V0_144	all	NetInterfacePowerOff → NetInterface Off
3V0_145	2.4.1-11	Test case not applicable for MOST50 ePHY

Change Ref.	Section	Changes
3V0_146	2.4.2-3	Flowchart updated: Deleted note "Tolerances of manufacturer have to be considered. If U_Low has hysteresis, the higher value has to be used"
3V0_147	2.4.2-1	Note update: "The test message has to be sent at every 0.5V step" → "The test message has to be sent at every step"
3V0_148	Para 2.1.3	NetServices → Network Services
3V0_149	2.8.4-9	Test description: InstID(0x0) → InstID(0x00)
3V0_150	2.1.3-1 2.2.1-6 2.2.1-7 2.3.2-3	T_ShutDown → t_DeadLockShort (to avoid additional measurement of t_shutdown)
3V0_151	2.6.2-4b	Test procedure updated; now independent from CR handling of DUT
3V0_152	2.6.2-4c	New test case "Device integration test (c) 2.6.2-4c"; continues from "Device integration test (b) 2.6.2-4b"
3V0_153	2.1.1-1	Test case deleted; covered by Lim. Phy. Compliance
3V0_154	2.5-13	Test case deleted; The precondition for a valid ring break diagnosis result is the existence of exactly one designated TimingMaster in the system.
3V0_155	2.5-10 2.5-11 2.5-12	Update of test cases to check RBDResult from DUT at the end of the test
3V0_156	Para 2.1.4	Update of definition of "family"
3V0_157	2.6.1-1	Update of Note how to get address from DUT
3V0_158	Para 3.1.6	Update of "DUT Manufacturer Information List": Now, voltage levels are included; tolerances no longer required
3V0_159	Para 3.1.8	Procedure "Check CapabilityToWake of DUT" deleted
3V0_160	2.1.1-6a	"CapabilityToWake" no longer checked during test. "PermissionToWake" no longer set during test. Add to "Device type": "Device which is capable to wake via MOST signal." Specification of parameter for AutoWakeup
3V0_161	2.1.1-4	"CapabilityToWake" no longer checked during test. "PermissionToWake" no longer set during test. Add to "Device type": "Device which is capable to wake via MOST signal." Changed Start Conditions: "DUT in NormalOperation; Ring Closed" Specification of parameter for AutoWakeup
3V0_162	2.1.3-1	"CapabilityToWake" no longer checked during test. "PermissionToWake" no longer set during test. Add to "Device type": "Device which is capable to wake via MOST signal." Specification of parameter for AutoWakeup
3V0_163	2.1.3-4	"CapabilityToWake" no longer checked during test. "PermissionToWake" no longer set during test. Add to "Device type": "Device which is capable to wake via MOST signal." Specification of parameter for AutoWakeup
3V0_164	2.4.1-1	"CapabilityToWake" no longer checked during test. "PermissionToWake" no longer set during test. Add to "Device type": "Device which is capable to wake via MOST signal." Specification of parameter for AutoWakeup
3V0_165	2.4.1-2	"CapabilityToWake" no longer checked during test. "PermissionToWake" no longer set during test. Add to "Device type": "Device which is capable to wake via MOST signal." Specification of parameter for AutoWakeup

Change Ref.	Section	Changes
3V0_166	2.3.3-1	Voltage level no longer derived vom FB ET but from manufacturer list. U_Normal → U_DUT_Operating U_Super → U_DUT_Operating_max
3V0_167	2.4.2-1	Start Conditions: Voltage level of power supply = U_DUT_Operating U_Normal → U_DUT_Operating U_Super → U_DUT_Operating_max
3V0_168	2.4.2-1	Test case applicable for MOST50 ePHY in case TM contains PM
3V0_169	all	NetInterface PowerOff → NetInterface Off
3V0_170	2.1.1-6b	Start Conditions: "DUT in NormalOperation" → "DUT: NetInterface Off"
3V0_171	3.1.1 wake-up	Now part of definition of procedures
3V0_172	2.4.1-1	Rename of variable "Retry" to "Attempt" for better understanding Dut not ok (1): ... too less .. → ... too little...
3V0_173	2.4.1-2	Rename of variable "Retry" to "Attempt" for better understanding. DUT not ok (1): The DUT fails to perform at least one wake-up retry. Test description: "permanent wakeup" → "using parameter Duration = 15 to simulate extended local wakeup event"
3V0_174	2.1.2-5	Test case not applicable for MOST50 ePHY
3V0_175	Para 3	New manufacturer information: DUT is capable to wake via MOST signal.
3V0_176	2.4.2-3	Specifiction of loops to reduce / increase voltage level DUT not ok (1): DUT disturbs communication before MOST signal is switched off.
3V0_177	2.3.3-4	Shutdown.ResultAck → Shutdown.Result Timeout of last query: t_DeadLockMid → t_DeadLockLong Wording of temperature levels synchronized with FB ET.
3V0_178	2.3.3-5	Shutdown.ResultAck → Shutdown.Result t_WaitAfterOvertempShutdown_min → max(t_WaitAfterOvertempShutdown_min, t_Restart_min)
3V0_179	RBD tests	Added MOST50 ePHY / MOST150 oPHY to test name. More detailed mechanism to trigger RBD (t_SSO_Shutdown considered for MOST150 oPHY)
3V0_180	2.5-2 2.5-3 2.5-7 2.5-8	RBD timings no longer derived from FB ET; now they are specified by MOST spec. Note deleted: "The reported node position of the RBD result has to be ignored."
3V0_181	2.5-10	DelayTime added to note in the middle of the flow chart ("t_Diag_Start_max + t_Diag_Signal_max" → "DelayTime + t_Diag_Start_max + t_Diag_Signal_max")
3V0_182	2.5-12	DelayTime added to note in the middle of the flow chart ("t_Diag_Start_max + t_Diag_Signal_max" → "DelayTime + t_Diag_Start_max + t_Diag_Signal_max")
3V0_183	2.5-10	Note added: "Potentially, RBDResult.Status will be received already while waiting for end of timer_1."
3V0_184	2.5.11	Note added: "Potentially, RBDResult.Status will be received already while waiting for end of timer_1."
3V0_185	All RBD tests	Harmonization of RBD result evaluation
3V0_186	2.5-2 2.5-3	Perform wake-up → turn on MOST signal
3V0_187	Para 3	Procedure "Perform wake-up": (e.g. optical slave wakeup including t_Restart) → (e.g. slave wakeup including t_Restart)
3V0_188	Appendix 2	Function "DiagTimeout": deleted references to test cases Function "VoltageLevels": deleted references to test cases Function "ECLActivate" added Function "ECLParticipantState" added Function "ECLInitiatorState" added

Change Ref.	Section	Changes
3V0_189	2.3.1-3	ShutDown.StartAck → ShutDown.Start t_Suspend → t_WaitSuspend
3V0_190	2.3.1-6	ShutDown.StartAck → ShutDown.Start t_Suspend → t_WaitSuspend DUT not ok (2) → DUT ok (2) (The PowerMaster may override suspend requests from its slaves)
3V0_191	2.3.3-5	ShutDown.ResultAck → ShutDown.Result
3V0_192	2.3.2-2	ShutDown.StartAck → ShutDown.Start t_Suspend → t_WaitSuspend
3V0_193	2.3.2-3	ShutDown.StartAck → ShutDown.Start
3V0_194	2.6.2-5	t_WaitBeforeScan_min → t_WaitBeforeScan_max (revised by 3V0_248)
3V0_195	Para 3	DUT:SleepMode: delete "Set PermissionToWake = "False""
3V0_196	2.4.1-1	Delete t_Restart and t_Config from value of interest
3V0_197	2.4.1-2	Delete t_Restart and t_Config from value of interest
3V0_198	2.4.1-1 2.4.1-2	Note about wake-up attempt: t_Config_min → t_Config t_Restart_min → t_Restart
3V0_199	Changes of DUT	Delete definition of change levels as already specified by MOST Compliance Specification; New rules defined how to treat changes of DUT
3V0_200	Manufacturer Information	Added "Transmission Class" to "Sink devices"
3V0_201	2.6.2-3a	Test setup: Tester2 in slave mode Request CR from DUT → Tester2: Request CR from DUT Delete "Change address of Tester1 to valid one"
3V0_202	Document Reference	MOST Compliance Requirements Specification 2V0 → 2V1 MOST Specification 3V0E1 → 3V0E2
3V0_203	Para 2.1.2	Figures → values; with new example
3V0_204	Para 3.1.1	New example for timer definitions
3V0_205	Manufacturer information list	DUT supports optical wakeup according to MOST Specification (wakeup via MOST signal) → DUT designed to be woken by MOST signal (defined as MOST signal on)
3V0_206	2.4.2-3	Note added to "generate (external) wake-up event": "If DUT needs an electrical wake-up signal, this has to be generated."
3V0_207	2.2.1-5	Note in Test Description: "If DUT does not support wake-up neither via MOST nor via electrical wake-up" → "If DUT does not support wake-up neither via MOST signal nor via electrical wake-up"
3V0_208	2.5-2 2.5-3	"Switch off power; DUT in Normal Operation" → "Switch power off; wait 1s; switch power on; effectuate DUT NormalOperation"
3V0_209	2.5-7	Test description: "A new wake-up event is generated (MOST signal on)" → The tester starts generating MOST signal
3V0_210	2.6.1-1	Note added to "perform wake-up": "If DUT needs an electrical wake-up signal, this has to be generated."
3V0_211	2.6.2-1	Note deleted: "If DUT is NWM but not TM, it has to check tester 1 before tester 2 (devices has to be checked in ring position order)." Note added to "perform wake-up": "If DUT needs an electrical wake-up signal, this has to be generated."
3V0_212	General Notes; Procedures	Added Note: MOST signal = modulated signal

Change Ref.	Section	Changes
3V0_213	3.1-1	TransmissionClass will be checked, too
3V0_214	3.2-1	TransmissionClass will be checked, too
3V0_215	3.1-3 3.1-4 3.1-5 3.1-6 3.1-8	Deleted Note: "In case where the bandwidth of sink cannot be determined because of the content type, the test case is skipped for this sink."
3V0_216	3.1-1 3.1-3 3.1-4 3.1-5 3.1-6 3.1-8	Update device type: All sink devices, supporting the following transmission classes: <ul style="list-style-type: none"> - Synchronous - Packetized Isochronous (might be done now, to be decided) - Discretelsochronous (will not be done now, only later as decided) Not applicable for: <ul style="list-style-type: none"> - QoS-IP - Asynchronous
3V0_217	3.2-1 3.2-3 3.2-4 3.2-5 3.2-6 3.2-7 3.2-8 3.2-9 3.2-14	Update device type: All source devices, supporting the following transmission classes: <ul style="list-style-type: none"> - Synchronous - Packetized Isochronous (might be done now, to be decided) - Discretelsochronous (will not be done now, only later as decided) Not applicable for: <ul style="list-style-type: none"> - QoS-IP - Asynchronous
3V0_218	2.2.1-3	Update FlowChart: Move check for "receiving MOST signal" into loop
3V0_219	2.2.1-4 2.2.1-5	Test cases for MOST50 ePHY deleted
3V0_220	2.2.1-7	Test setup: Tester 1 in master mode
3V0_221	2.8.4-2	First note in FlowChart: a) "sent by the tester → sent by the DUT" b) In case the MessageBufSize is dynamic (indicated by value 0) the DUT has to send at least (LAMSmax + 1) bytes and at maximum 4 segments
3V0_222	2.5-8	Update of test description: Expected RBD result: "inconclusive (0xFC)" (revised by 3V0_244)
3V0_223	3.1-1 3.1-3 3.1-4 3.1-5 3.1-6 3.1-8 3.2-1 3.2-3 3.2-4 3.2-5 3.2-6 3.2-7 3.2-8 3.2-9 3.2-14	Update device type: Delete section "not applicable for"
3V0_224	3.1-1	Test is applicable for all Transmission Classes --> Device Type: All sink devices
3V0_225	3.2-1	Test is applicable for all Transmission Classes --> Device Type: All source devices

Change Ref.	Section	Changes
3V0_226	3.1-3 3.1-4 3.1-5 3.1-6 3.1-8	Device Type: All sink devices, supporting transmission class "synchronous" Multi Node Devices: All devices containing at least one sink that supports transmission class "synchronous"
3V0_227	3.2-3 3.2-4 3.2-5 3.2-6 3.2-7 3.2-8 3.2-9 3.2-14	Device Type: All source devices, supporting allocate mechanism and transmission class "synchronous" Multi Node Devices: All devices containing at least one source that supports allocate mechanism and transmission class "synchronous"
3V0_228	3.1-8	Delete test case; mute no longer tested.
3V0_229	all	Inc(x) \rightarrow x := x + 1
3V0_230	Document Reference; Para 2.1.2	FB ET 3V0 \rightarrow FB ET 3V0.1
3V0_231	Changes of DUT	Re-Test is mandatory in case of implementation of new specification version \rightarrow Re-Test is mandatory in case of implementation of new specification version which affect a compliance relevant functional behavior of a MOST device in the system context.
3V0_232	2.2.1-4 2.2.1-5	FlowChart: Delete action "Delete Shutdown Flag"; add Note "By switching off MOST signal, Shutdown Flag will be deleted automatically."
3V0_233	3.2-5	MSC: New procedure for tester to allocate bandwidth
3V0_234	3.2-6	MSC: New procedure for tester to deallocate bandwidth
3V0_235	3.0-1	Delete exception for case "function StreamDataInfo not implemented in principal"
3V0_236	2.2.1-7	Test only performed once (with SSO)
3V0_237	2.4.1-1 2.4.1-2	FlowChart: Delete "Wait 1s"; Replaced by loop to wait for MOST signal from DUT.
3V0_238	2.1.1-6a	FlowChart: Delete "Wait 1s"
3V0_239	2.1.1-4	FlowChart: Delete "Wait 1s"
3V0_240	2.1.3-1	FlowChart: Delete "Wait 1s"
3V0_241	2.1.3-4	FlowChart: Delete "Wait 1s"
3V0_242	Changes of DUT	Clarify wording, when re-test is mandatory
3V0_243	Product variants	"Level 1" and "Level 2" changes: reference to MOST Compliance Requirements Specification
3V0_244	2.5-8	a) Any RBD result results in DUT ok, except "Diagnosis ok" b) Experimental setup: Tester 1 in master mode. c) Add note to flowchart: "Tester 1 has to perform RBD procedure that is specified for TimingSlave."
3V0_245	Para 3.1.5	Update of DUT Manufacturer Information List
3V0_246	2.4.2-1	Update flowchart: New procedure for loops to reduce and increase voltage level. Delete note "Hysteresis of manufacturer met?" Delete result "DUT not ok (4): No Hysteresis as specified by manufacturer (U_DUT_Operating <-> U_Critical)."

Change Ref.	Section	Changes
3V0_247	2.8.4-8	Adapt note: "Use message "NetBlock.FBlockIDs.Get" for the test. Every message must have an unique ID. The ID is incremented with every loop." → "Use message "NetBlock.InstID.<Count>.Get" for the test. Reason: Every message must have a unique ID. The ID changes by incrementing the value count with every loop."
3V0_248	2.6.2-5	Test update: DUT has to perform system scan before t_WaitBeforeRescan_max expires
3V0_249	Para 2.1.3	New definition of re-test
3V0_250	Appendix 3	Deleted
3V0_251	2.3.2-2	FlowChart: First query: "...SuspendMode" → "...ShutdownSuspendMode" DUT has to send ShutDown.Result(Suspend) within t_WaitSuspend_min
3V0_252	2.4.2-1	FlowChart: Delete boxes "Level1 := actual voltage level" and "Level2 := actual voltage level"
3V0_253	2.6.2-5	FlowChart; 2. query: "DUT performs system scan?" → "DUT starts system scan?"
3V0_254	Para 3.1.5	DUT Manufacturer Information List: "LowLevel retries" → "LowLevel and MidLevel retries"
3V0_256	2.1.1-4	FlowChart: Update procedure to check whether DUT switches off MOST signal between t_Config_min and t_Config_max
3V0_257	2.4.1-1	Test description: "...wakeup condition will be generated permanently..." → "...wakeup condition will be simulated permanently..."

