

# MOST

Media Oriented Systems Transport

Multimedia and Control  
Networking Technology

**MOST Extended Core Compliance Test  
Specification: MOSTHigh**

**Rev 1.0.3E1**

**04/2016**

Version 1.0-03E1

**MOSTCO CONFIDENTIAL**

See page 3 for the terms of disclosure



## Legal Notice

### **COPYRIGHT**

© Copyright 1999 - 2016 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](mailto:contact@mostcooperation.com)

Web: [www.mostcooperation.com](http://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 - 2016 MOST Cooperation.  
All rights reserved.

MOST is a registered trademark

## Contents

<b>1</b>	<b>INTRODUCTION .....</b>	<b>8</b>
<b>2</b>	<b>GENERAL NOTES .....</b>	<b>9</b>
2.1	Definitions .....	9
2.1.1	Minimum Requirements For Tester .....	9
2.2	GFT .....	9
2.3	Test Procedure .....	13
2.4	Device Types .....	14
2.5	Input By Device Manufacturer .....	15
2.6	CRC32 Implementation Of FB ET .....	16
2.7	Macros .....	17
2.7.1	MACRO_MH_DSI_EstablishConnection .....	17
2.7.2	MACRO_MH_DSI_EstablishConnection_Hold .....	19
2.7.3	MACRO_MH_DSI_TransferData .....	20
2.7.4	MACRO_MH_DSI_MultipleFramesRequest .....	23
2.7.5	MACRO_MH_DSO_RequestConnection .....	24
2.7.6	MACRO_MH_DSO_EstablishConnection .....	26
2.7.7	MACRO_MH_DSO_MultiConnection .....	27
2.7.8	MACRO_MH_DSO_TransferData .....	30
<b>3</b>	<b>EXTENDED MOST CORE COMPLIANCE TESTS FOR MHP .....</b>	<b>32</b>
3.1	DSI / DSO Identification .....	32
3.1.1	TEST_MH_Identification .....	32
3.2	Functional Tests For DSI .....	35
3.2.1	TEST_MH_DSI_RequestConnection_Ok .....	35
3.2.2	TEST_MH_DSI_Frame_Appropriate .....	37
3.2.3	TEST_MH_DSI_Frame_TooBig .....	40
3.2.4	TEST_MH_DSI_MultipleFramesRequest .....	43
3.2.5	TEST_MH_DSI_Basic .....	46
3.2.6	TEST_MH_DSI_MultiConnection .....	49
3.3	Functional Tests For DSO .....	51
3.3.1	TEST_MH_DSO_RequestConnection .....	51
3.3.2	TEST_MH_DSO_RequestConnection_NoResponse .....	54
3.3.3	TEST_MH_DSO_RequestConnection_Ok .....	56
3.3.4	TEST_MH_DSO_Basic .....	58
3.3.5	TEST_MH_DSO_MultiConnection .....	61
3.4	Timing Tests For DSI .....	64
3.4.1	TEST_MH_DSI_t_ready .....	64
3.4.2	TEST_MH_DSI_t_frame .....	67
3.4.3	TEST_MH_DSI_t_receive .....	69
3.4.4	TEST_MH_DSI_t_Hold .....	71
3.4.5	TEST_MH_DSI_t_dwn_NegAck .....	73
3.4.6	TEST_MH_DSI_t_mfr .....	76
3.5	Timing Tests For DSO .....	78
3.5.1	TEST_MH_DSO_t_send .....	78
3.5.2	TEST_MH_DSO_t_end .....	81
3.5.3	TEST_MH_DSO_t_Hold_Resend .....	84
3.5.4	TEST_MH_DSO_t_retrans .....	87
	<b>APPENDIX 1: REQUIREMENTS AND MEASUREMENT UNCERTAINTY FOR TEST EQUIPMENT AND SETUP .....</b>	<b>90</b>
	<b>APPENDIX 2: FBLOCK ET REFERENCE TO MHP TEST CASES (INFORMATIVE) .....</b>	<b>91</b>
	<b>APPENDIX 3: GFT SYMBOLS (INFORMATIVE) .....</b>	<b>92</b>
	<b>DOCUMENT HISTORY .....</b>	<b>96</b>

## 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 Core Compliance Test Specification	1.2
		1.3
		3.0
[2]	MOST High Protocol Specification, including Errata	2.3.1
[3]	MOST FunctionBlock "Enhanced Testability"	2.6.1
		3.0
[4]	Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part3: TTCN-3 Graphical presentation Format (GFT) ETSI ES 201 873-3 V3.2.1 (2007-02)	3.2.1

## Abbreviations

DSI	Data Sink
DSO	Data Source
DUT	Device Under Test
GFT	TTCN-3 Graphical Presentation Format

## Overview Of Extended Core Compliance Tests (Sorted By Test Number)

Test name	Test number	Page
TEST_MH_Identification	4.0-1	32
TEST_MH_DSI_RequestConnection_Ok	4.1-1	35
TEST_MH_DSI_Frame_Appropriate	4.1-2	37
TEST_MH_DSI_Frame_TooBig	4.1-3	40
TEST_MH_DSI_MultipleFramesRequest	4.1-4	43
TEST_MH_DSI_Basic	4.1-5	46
TEST_MH_DSI_MultiConnection	4.1-6	49
TEST_MH_DSO_RequestConnection	4.2-1	51
TEST_MH_DSO_RequestConnection_NoResponse	4.2-2	54
TEST_MH_DSO_RequestConnection_Ok	4.2-3	56
TEST_MH_DSO_Basic	4.2-4	58
TEST_MH_DSO_MultiConnection	4.2-5	61
TEST_MH_DSI_t_ready	4.3-1	64
TEST_MH_DSI_t_frame	4.3-2	67
TEST_MH_DSI_t_receive	4.3-3	69
TEST_MH_DSI_t_Hold	4.3-4	71
TEST_MH_DSI_t_dwn_NegAck	4.3-5	73
TEST_MH_DSI_t_mfr	4.3-6	76
TEST_MH_DSO_t_send	4.4-1	78
TEST_MH_DSO_t_end	4.4-2	81
TEST_MH_DSO_t_Hold_Resend	4.4-4	84
TEST_MH_DSO_t_retrans	4.4-5	87

# 1 Introduction

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

For super ordinate topics see MOST Core Compliance Test Specification [1].



## 2 General Notes

### 2.1 Definitions

#### 2.1.1 Minimum Requirements For Tester

To be able to perform compliance tests for MOST High, the tester has to meet the following requirements, additionally to all requirements for a node supporting MOSTHigh:

- Maximum possible packet length of 1014 byte (MOST Specification Rev. 2.5) resp. 1524 byte (MOST Specification Rev. 3.0).
- Support of 30 simultaneous MHP connections at least.
- Minimum Average Interrupt Rate of Tester: AIR = 1000.
- The tester is only allowed to deviate from MOST Specification and MHP Specification if indicated in the test case specification.
- Timing measurement of the tester must ensure the measurement uncertainty given in Appendix 1.
- The tester shall ensure that typical timing values are within  $\pm 10\%$  deviation.

### 2.2 GFT

TTCN-3 Graphical presentation Format (GFT) is used to support specification of test cases for MHP.

#### Messages

Within the GFTs, two kinds of messages are utilized, MOST control messages and MHP messages. MOST control messages are connected to port instance "pET".

MHP messages are connected to port instance "pMHPframes".

Port instance "self" covers all messages sent by the tester and received from the tester.

Port instance "pTrigger" is used to exchange information (e.g. trigger events, states) between functions within the tester.

Port instance "pNetwork" is used for configuration purpose of the network. It is not used to exchange MOST control messages or MHP messages.

All port instances are implemented in the tester.

#### 0-Frame:

A 0-Frame is a DataFrame with FrameAckHigh=0 which is omitted in the 0-Frame message:

NULLFrame(?,?,?,?).

#### NegativeAcknowledge:

A NegativeAcknowledge is an Acknowledge with FrameAckHigh=0 and FrameAckLow=0 which are omitted in NegativeAcknowledge message: NegativeAcknowledge(?).

#### Note:

*The MHP related functions of FB ET (DSO and DSI) require parameter SenderHandle. This parameter is not relevant to establish a MHP connection. SenderHandle is relevant only for communication of devices via an established MHP channel.*

## References in GFTs

Function	Description of Function	Description of Parameter
Void f_MHPinit(inout MHPdata_rType p_MHPdata, in integer p_NDF, in integer p_scale, in integer p_blocks)	<p>Initiates a MHP data block filled with Senderhandle (0x1234), Dummy Data (specified by FB ET.DSO) and initialised values of NDF, scale and number of blocks.</p> <p>Example: f_MHPinit(Data, startConnectionParameters.NDFack, startConnectionParameters.Scale, 1)</p> <p>Parameter p_MHPdata based on data type MHPdata_rType which is defined as follows: Type record MHPdata_rType { integer numberOfBlocks, integer scale, integer NDF, hexstring data, integer dataSize }</p>	<p>p_MHPdata: Data to be transmitted</p> <p>p_NDF: Value of NDF</p> <p>p_scale: Value of Scale</p> <p>p_blocks: Number of bocks</p>
Hexstring f_MHPframeData(inout MHPdata_rType p_MHPdata, in integer p_blockNumber, in integer p_frameNumber)	<p>Returns the required payload for a DataFrame (extract of data to be transmitted) according to number of the block and number of the frame.</p> <p>Example: f_MHPframeData(Data,0,1)</p>	<p>p_MHPdata: Data to be transmitted</p> <p>p_blockNumber: Number of the block</p> <p>p_frameNumber: Number of the frame</p>
Void f_TriggerConfigurationStatusNotOk ( )	<p>This function triggers Configuration.Status(NotOk) of DUT.</p> <p>Procedure depends whether DUT is NWM or not.</p> <p>DUT is NWM:</p> <ol style="list-style-type: none"> <li>1. Trigger Configuration.Status(NotOk) by sending FBlockIDs.Status to DUT, using 0xFFFF as sender address.</li> <li>2. Wait for Configuration.Status(NotOk), sent by DUT. If case DUT fails to send Configuration.Status(NokOk) within t_DeadLockMid → Abort test: "DUT not ok: Configuration.Status(NotOk) has not been established".</li> <li>3. Respond to system scan of DUT with normal sender address.</li> </ol> <p>DUT is not NWM:</p> <ol style="list-style-type: none"> <li>1. Send Configuration.Status(NotOk) to DUT.</li> <li>2. Pause according to MOST Specification: MOST Spec 2V5: t_wait_before_scan MOST Spec 3V0: t_wait_before_rescan</li> <li>3. Perform system scan</li> </ol>	

Function	Description of Function	Description of Parameter
Void f_WaitForConfigurationStatusOk() ( )	<p>This function waits until DUT enters Configuration.Status(Ok).</p> <p>Procedure depends whether DUT is NWM or not.</p> <p>DUT is NWM:</p> <ol style="list-style-type: none"> <li>1. Wait for Configuration.Status(Ok), sent by DUT. If case DUT fails to send Configuration.Status(Ok) within t_DeadLockMid → Abort test: "DUT not ok: Configuration.Status(Ok) has not been established".</li> <li>2. Wait for t_DeadLockShort</li> </ol> <p>DUT is not NWM:</p> <ol style="list-style-type: none"> <li>1. Send Configuration.Status(Ok) to DUT</li> <li>2. Wait for t_DeadLockShort</li> </ol>	
Integer f_get_t_AIR_Delay(integer DSI_AIR)	This function returns value of t_AIR_Delay, based on parameter AIR of START CONNECTION, received by DUT.	DSI_AIR: Value of AIR (parameter of START CONNECTION)
Boolean f_MHPcompareCRC(in MHPdata_rType p_MHPdata, in integer receivedCRC)	<p>This function calculates the CRC of p_MHPdata and compares the result with receivedCRC. In case both values are identical, the function returns "true".</p> <p>The checksum calculation is specified by FB ET.DSI.</p> <p>Example: f_MHPcompareCRC(Data, DSI_ResultAck_Msg.Data.DSI_ResultAck_Data.rc32</p>	<p>p_MHPdata : Data for CRC calculation</p> <p>receivedCRC: Checksum</p>
Boolean f_analyse_DataPattern()	<p>This function analyses the data of each data frame, received from DUT via MHP of the active connection.</p> <p>In case the data pattern is identical to the dummy data, specified by FB ET.DSO, the function returns "true".</p>	

## Parameter

All parameters containing manufacturer information start with "mi\_".  
The manufacturer information list contains a reference to relevant GFT parameter, if applicable.

## Timing Values

If not other stated, timing values in GFTs are specified in seconds.

## Enums

The following enums are used in GFTs, based on FBlockET.DSO.

### Parameter NextPacketMethod

Enum	Code	Symbolic Name	Description
e_Normal_HOLD_phase_will_close_after_timeout	0x00	Hold	Normal HOLD phase (will close after timeout).
e_Terminate_the_connection_and_open_a_new_one_if_necessary	0x01	Terminate	Terminate the connection and open a new one if necessary.
e_Try_to_send_next_packet_without_a_hold_optional	0x02	SendNext	Try to send next packet without a hold (optional).

### Parameter AckMode

Enum	Code	Symbolic Name	Description
e_use_single_frame_acknowledge	0x00	SingleFrameAcknowledge	use single frame acknowledge
e_use_block_acknowledge	0x01	BlockAcknowledge	use block acknowledge

## 2.3 Test Procedure

- Use experimental setup of MOST Core Compliance Test Specification [1].
- If it is proven that the DUT is not able to reach precondition (such as modulated signal, lock, scan performed, System State OK), the test case will be marked as "DUT not ok (DUT not able to reach precondition)".
- Because of reproducibility reasons after each test case the DUT has to be disconnected from power until buffer capacitors are empty. The DUT shall be shutdown regularly before disconnecting from power.
- In case one test consists of several test loops, DUT has to be disconnected from power until buffer capacitors are empty before every test loop. The DUT shall be shutdown regularly before disconnecting from power.
- Unknown MHP connections that are not considered by the test case are ignored by the tester.
- In case DUT = DSI: The tester has to consider  $t_{AIR\_Delay}$  during testing.
- In case DUT = DSO: The tester has to provide AIR value (testerAIR = 2000) with START CONNECTION.
- If not other stated, tester has to use maximum value of NDF (parameter requestConnectionNDF of REQUEST CONNECTION command). The value depends on supported MOST Specification (refer to MOST High Protocol Specification, para 10.1 MHP Frame Size).

### Structure of ConID

The function DSO of FB ET requires parameter MHP\_ConID that contains TargetFBlockID, TargetInstID, TargetFktID and TargetOPType. In GFTs, the parameter DSOMHPConID is used with the following structure and values:

TargetFBlockID = 0x0F  
TargetInstID = <dependent on ring position>  
TargetFktID = The target FktID of MHP\_ConID always starts with 0x400. In case of multiple connections, the value will be increased by one with each additional connection.  
TargetOPType = 0x6 (StartResultAck)

## States Of DUT

Each test case specifies a start condition that contains the state of DUT. This state has to be triggered as follows:

State	Procedure to trigger state
<i>DUT in NormalOperation</i>	Use procedure for "DUT: NormalOperation" as specified in MOST Core Compliance Test Specification [1].

## 2.4 Device Types

This specification is applicable for all devices supporting MHP.

## 2.5 Input By Device Manufacturer

General properties that are not MHP specific but potentially relevant for testing (e.g. DUT is TimingMaster) have to be derived from DUT manufacturer information list of Core Compliance Test Specification. The DUT manufacturer information list of Core Compliance Test Specification [1] has to be provided by the DUT manufacturer.

Additionally, the manufacturer of the DUT has to provide some implementation details for MHP test parameterization to the MOST Compliance Test House. They have to be marked in the test plan / test report. For DSI / DSO devices, the following implementation details are required:

Item / Property	Note / Remark	Reference to MHP Compliance Test Case
Support of MHP <ul style="list-style-type: none"> <li>DSI function</li> <li>DSO function</li> </ul>	Implementation of DSI / DSO functionality.  DSI tests will be performed in case DUT implements DSI functionality. DSO tests will be performed in case DUT implements DSO functionality.	all test cases
Necessary value for "Prio" of REQUEST CONNECTION (DUT = DSI)	Reference in GFTs: mi_RequestConnectionPrio	MACRO_MH_DSI_EstablishConnection  4.1-1 TEST_MH_DSI_RequestConnection_Ok  4.1-3 TEST_MH_DSI_Frame_TooBig  4.3-1 TEST_MH_DSI_t_ready
RevID, used by DUT	Only DUT with RevID = "0x02" will be tested.  Reference in GFTs: mi_MHPRevID	all test cases
Maximum number of simultaneous (external) MHP connections (DSI / DSO).	If the DUT establishes by default some MHP connections internally which will reduce the maximum number of remaining MHP connections for testing, this has to be considered.  Reference in GFTs: mi_DSICount mi_DSOCOUNT	4.0-1 TEST_MH_Identification  4.1-6 TEST_MH_DSI_MultiConnection  4.2-5 TEST_MH_DSO_MultiConnection
Boundary descriptor / SBC-Register, specified by system integrator (System-SBC).	The MOST High Test Specification will perform all tests with this boundary descriptor / SBC value.  No references in GFTs exist as it is a configuration parameter of tester.  SBC-Register only applicable for MOST Rev. 1.2 devices. In other cases use "boundary descriptor" of MOST Core Compliance Test Specification Rev. 3.0	all test cases
Support of NextPacketMethod = SendNext	Some test cases use NextPacketMethod = SendNext (try to send next packet without a hold) if supported by DUT.	4.2-1 TEST_MH_DSO_RequestConnection  4.2-4 TEST_MH_DSO_Basic

## 2.6 CRC32 Implementation Of FB ET

To check the CRC32 implementation of FB ET, the following test vectors can be used:

No.	Data	Datalength	crc32
a	{0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37, 0x38, 0x39}	9	0xcbf43926
b	{0xef, 0x29, 0x1b, 0x81, 0x56, 0x38, 0xc7, 0xd8}	8	0x48cd00a8
c	{0x00, 0x00, 0x00}	3	0xff41d912
d	{}	0	0

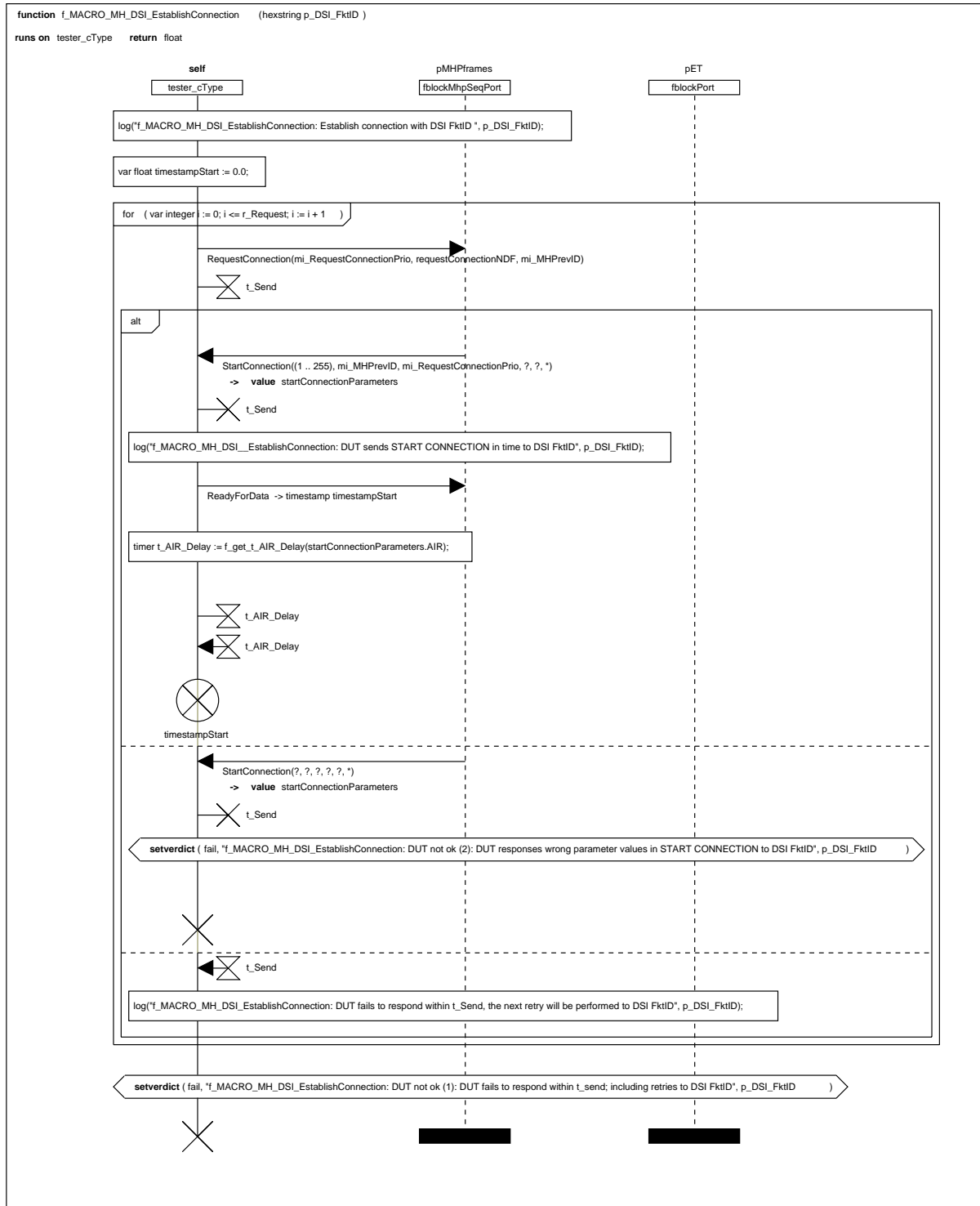


## 2.7 Macros

Macros are called by several test cases.

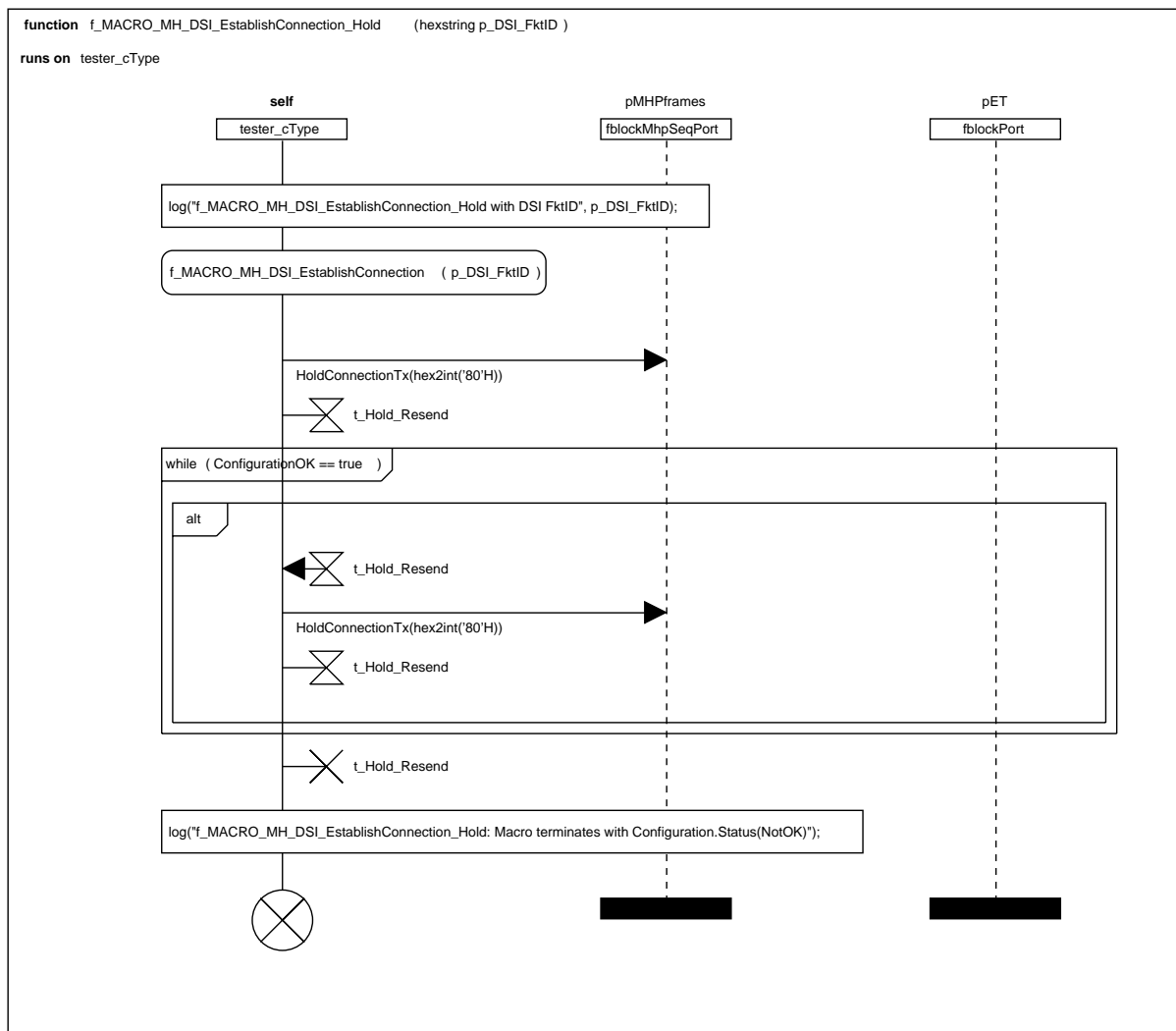
### 2.7.1 MACRO\_MH\_DSI\_EstablishConnection

<b>Name of macro</b>	MACRO_MH_DSI_EstablishConnection
<b>Reference to GFT</b>	f_MACRO_MH_DSI_EstablishConnection (hexstring p_DSI_FktID)
<b>Description</b>	This macro establishes a MHP connection up to READY FOR DATA. It returns timestamp of READY FOR DATA.
<b>Note</b>	
<b>Results</b>	<b>DUT not ok (1):</b> DUT fails to respond within t_send; including retries to DSI FktID <b>DUT not ok (2):</b> DUT responses wrong parameter values in START CONNECTION to DSI FktID



## 2.7.2 MACRO\_MH\_DSI\_EstablishConnection\_Hold

<b>Name of macro</b>	MACRO_MH_DSI_EstablishConnection_Hold
<b>Reference to GFT</b>	f_MACRO_MH_DSI_EstablishConnection_Hold (hexstring p_DSI_FktID)
<b>Description</b>	This macro establishes a MHP connection, using macro f_MACRO_DSI_EstablishConnection (p_DSI_FktID). After that, the connection is set to hold. No return value.
<b>Note</b>	Possibly, the macro returns delayed after ConfigurationOk == False due to timeout t_Hold_Resend. Detection of ConfigurationOk depends on Configuration.Status of NetworkMaster.
<b>Results</b>	

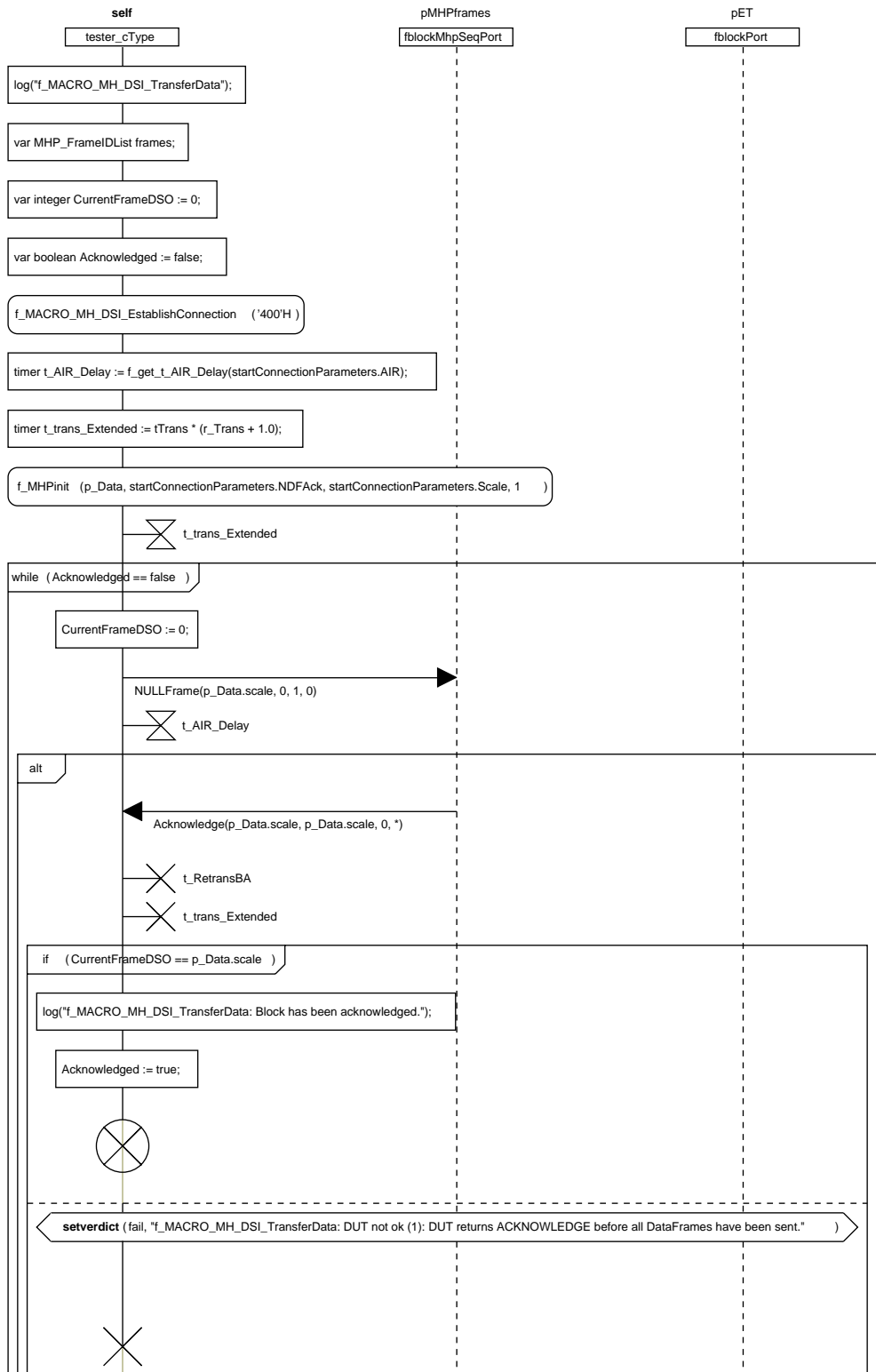


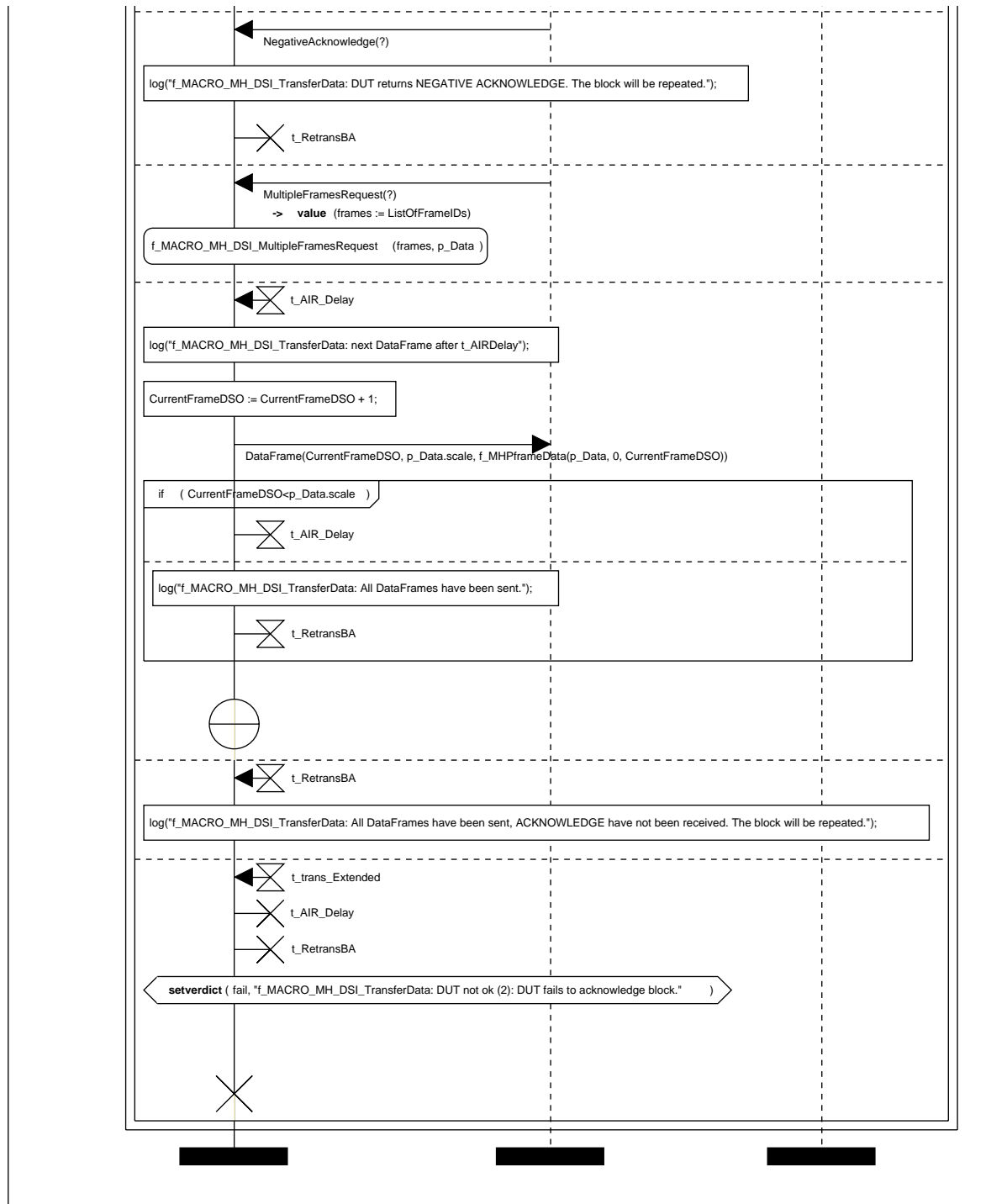
### 2.7.3 MACRO\_MH\_DSI\_TransferData

<b>Name of macro</b>	MACRO_MH_DSI_TransferData
<b>Reference to GFT</b>	f_MACRO_MH_DSI_TransferData (MHPdata_rType p_Data)
<b>Description</b>	This macro transmits data to DUT (DSI), using block acknowledge mechanism. No return value.
<b>Note</b>	
<b>Results</b>	<b>DUT not ok (1):</b> DUT returns ACKNOWLEDGE before all DataFrames have been sent. <b>DUT not ok (2):</b> DUT fails to acknowledge block.

```
function f_MACRO_MH_DSI_TransferData ( MHPdata_rType p_Data )
```

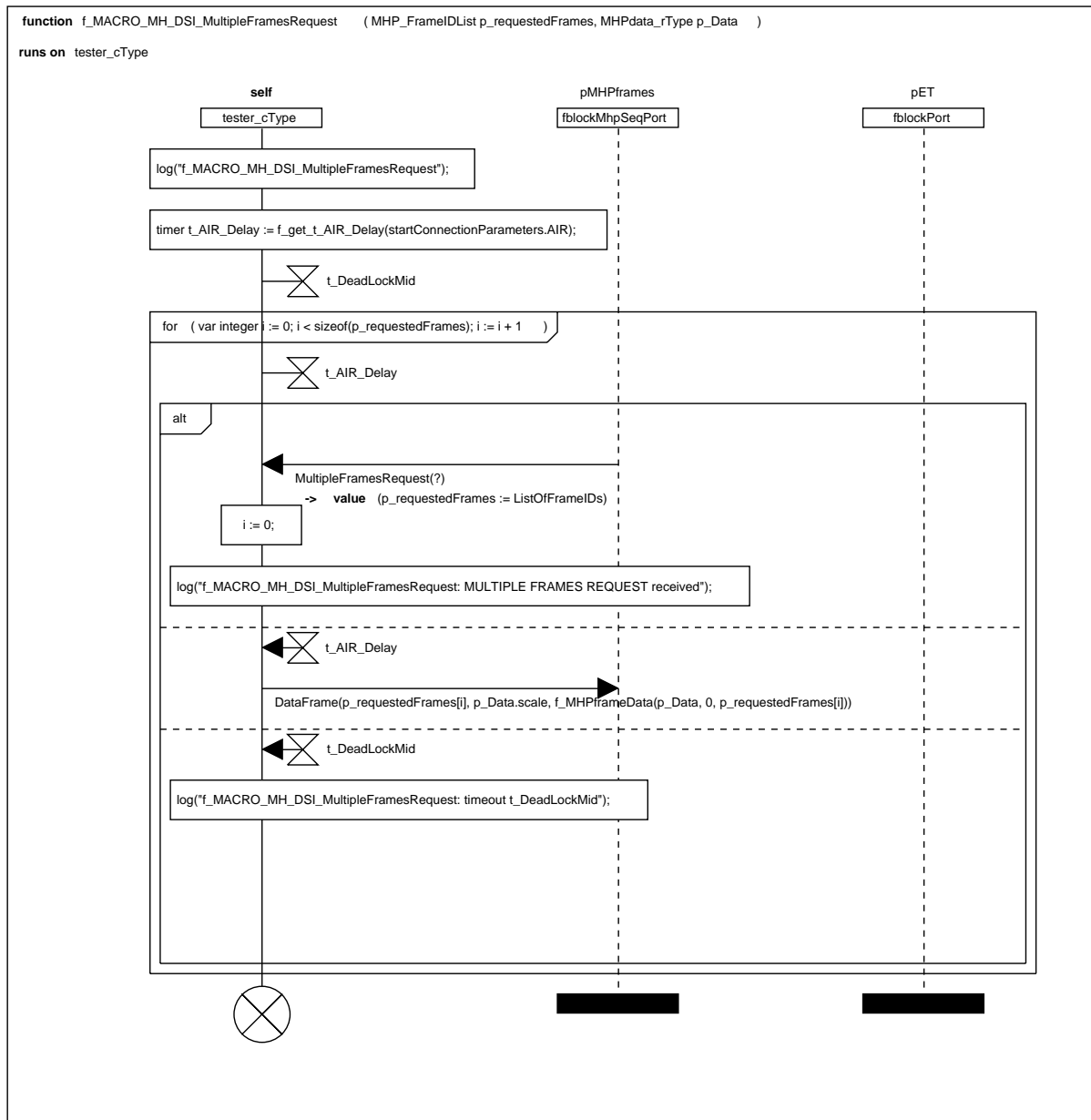
```
runs on tester_cType
```





## 2.7.4 MACRO\_MH\_DSI\_MultipleFramesRequest

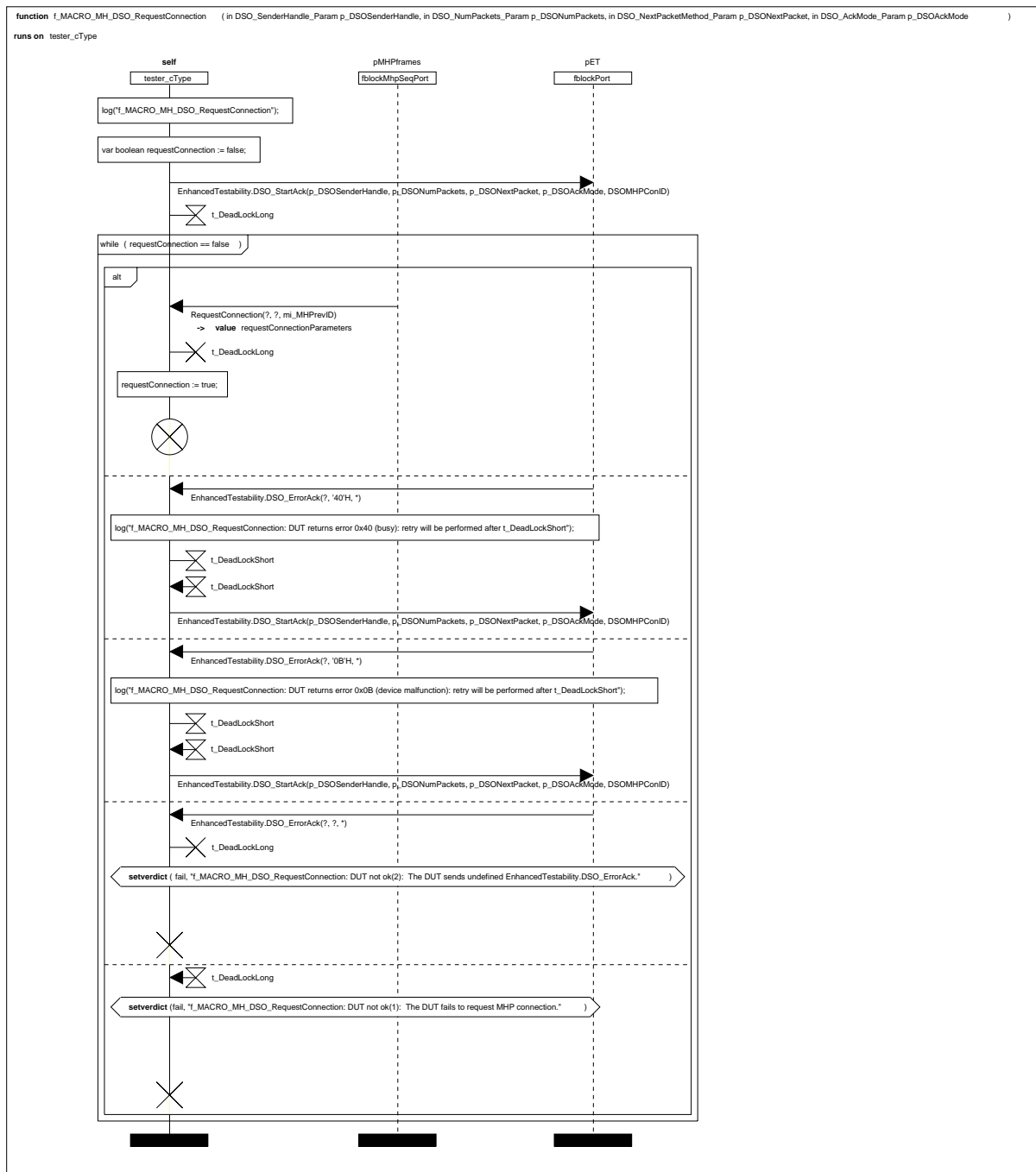
<b>Name of macro</b>	MACRO_MH_DSI_MultipleFramesRequest
<b>Reference to GFT</b>	f_MACRO_MH_DSI_MultipleFramesRequest ( MHP_FrameIDList p_requestedFrames, MHPdata_rType p_Data )
<b>Description</b>	This macro is used to (re)send all frames, the DUT requests by means of MULTIPLE FRAMES REQUEST. No return value.
<b>Note</b>	
<b>Results</b>	



## 2.7.5 MACRO\_MH\_DSO\_RequestConnection

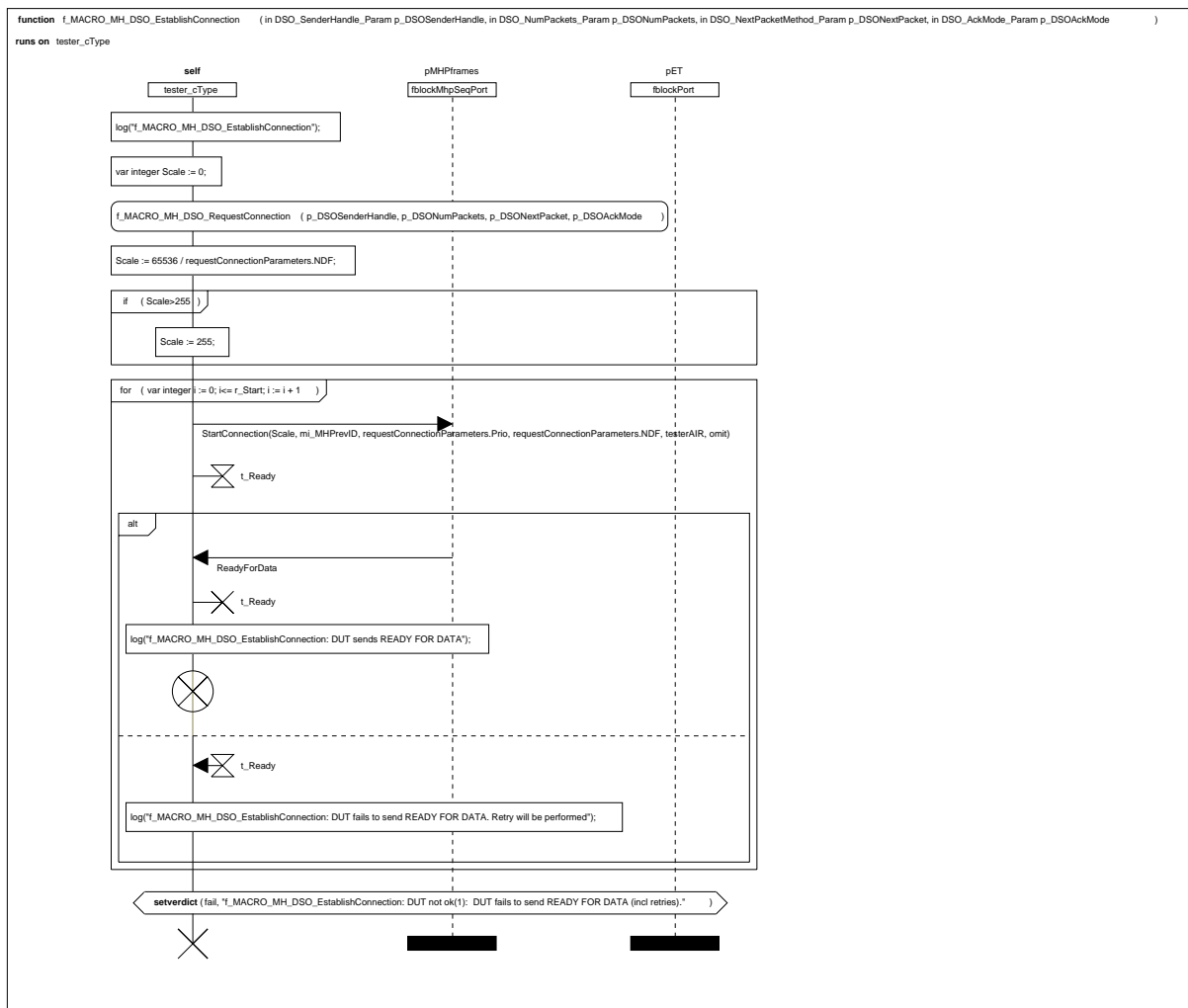
<b>Name of macro</b>	MACRO_MH_DSO_RequestConnection
<b>Reference to GFT</b>	f_MACRO_MH_DSO_RequestConnection (in DSO_SenderHandle_Param p_DSOSenderHandle, in DSO_NumPackets_Param p_DSONumPackets, in DSO_NextPacketMethod_Param p_DSONextPacket, in DSO_AckMode_Param p_DSOAckMode)
<b>Description</b>	This macro triggers REQUEST CONNECTION via FBlockET.DSO including retries in case of FBlockET.ErrorAck 0x0B or 0x40. No return value.
<b>Note</b>	
<b>Results</b>	<b>DUT not ok (1):</b> The DUT fails to request MHP connection. <b>DUT not ok (2):</b> The DUT sends undefined EnhancedTestability.DSO_ErrorAck.





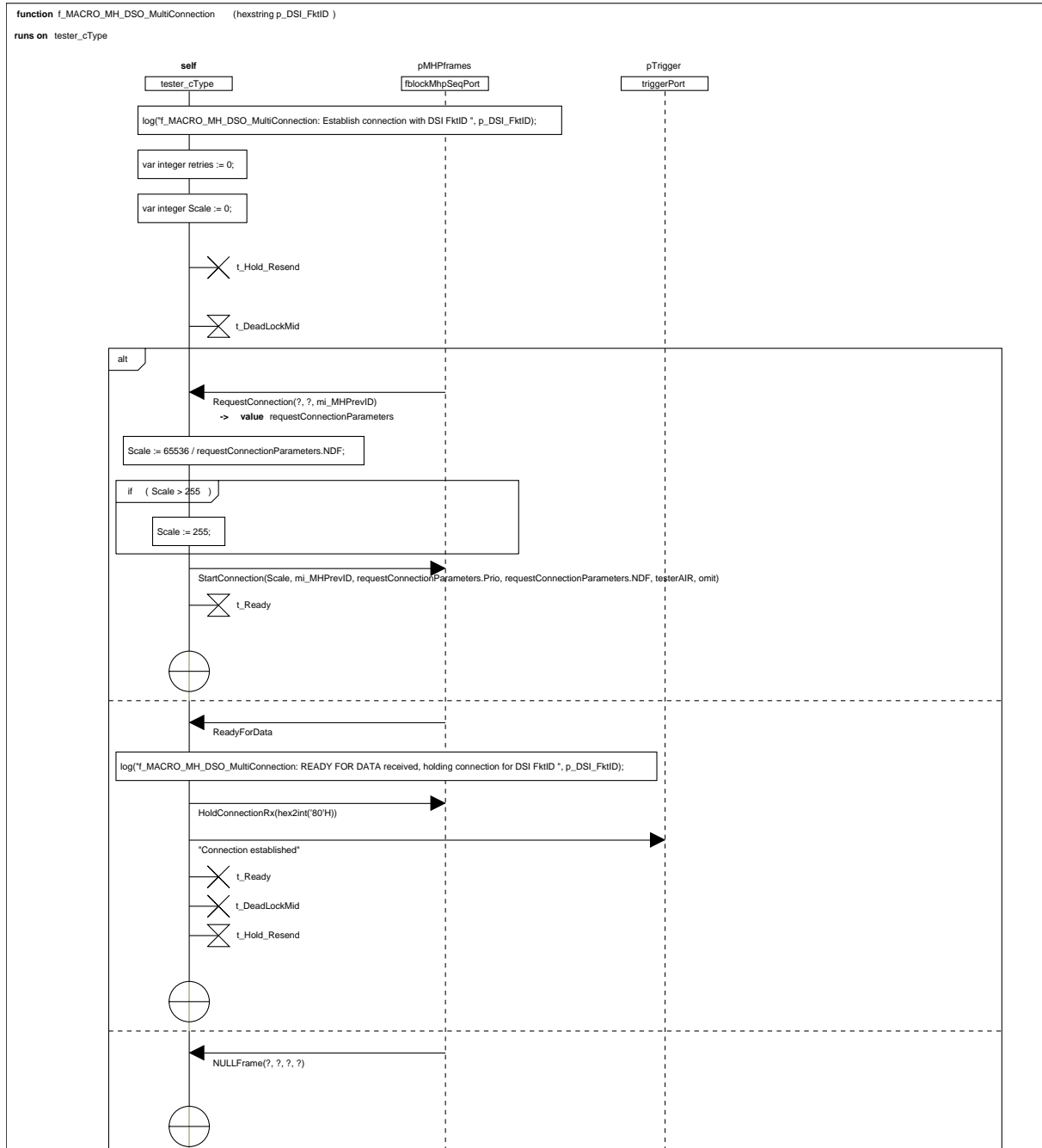
## 2.7.6 MACRO\_MH\_DSO\_EstablishConnection

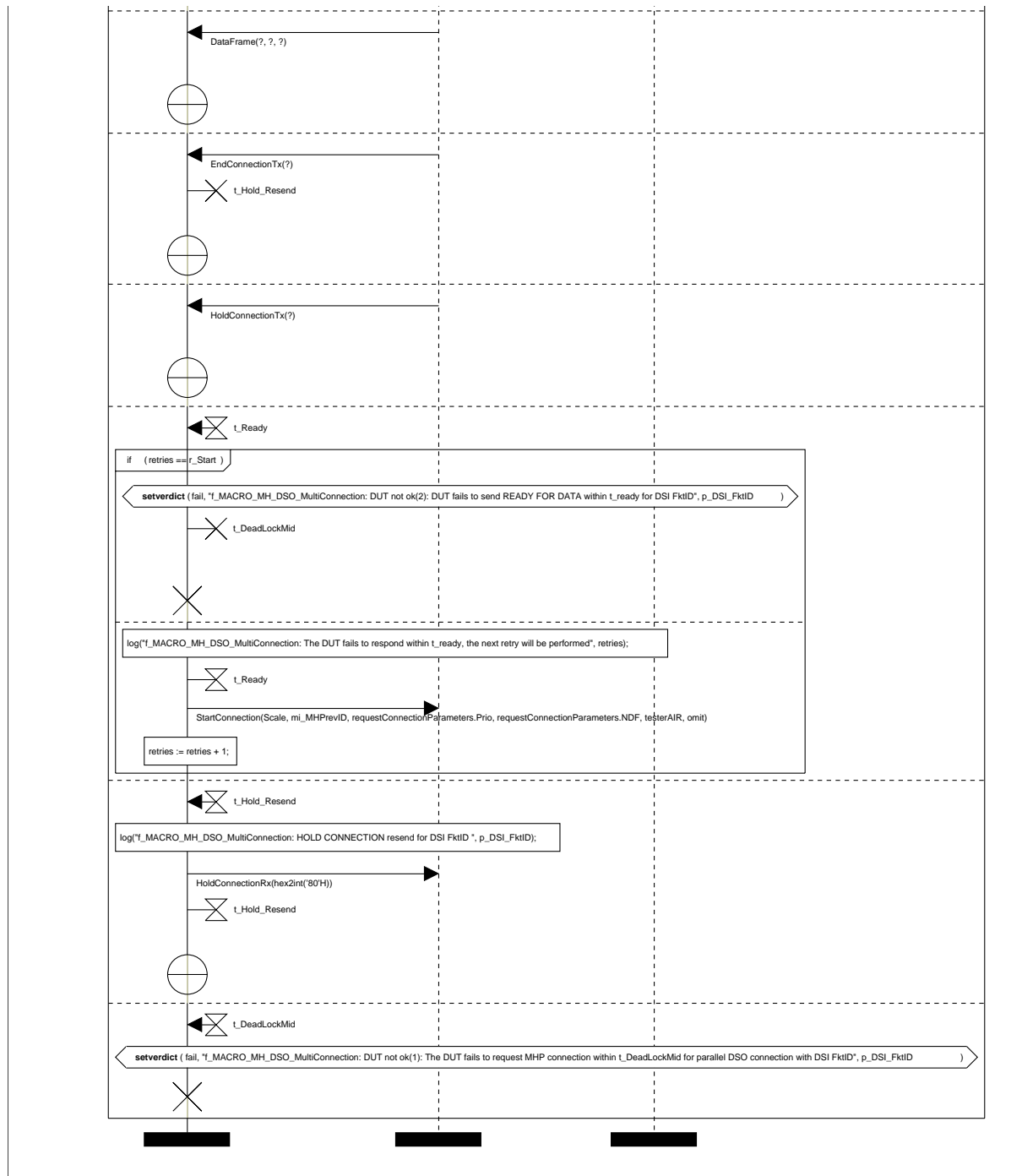
<b>Name of macro</b>	MACRO_MH_DSO_EstablishConnection
<b>Reference to GFT</b>	f_MACRO_MH_DSO_EstablishConnection (in DSO_SenderHandle_Param p_DSOSenderHandle, in DSO_NumPackets_Param p_DSONumPackets, in DSO_NextPacketMethod_Param p_DSONextPacket, in DSO_AckMode_Param p_DSOAckMode)
<b>Description</b>	This macro establishes a MHP connection up to READY FOR DATA. No return value.
<b>Note</b>	
<b>Results</b>	<b>DUT not ok (1):</b> DUT fails to send READY FOR DATA (incl retries).



## 2.7.7 MACRO\_MH\_DSO\_MultiConnection

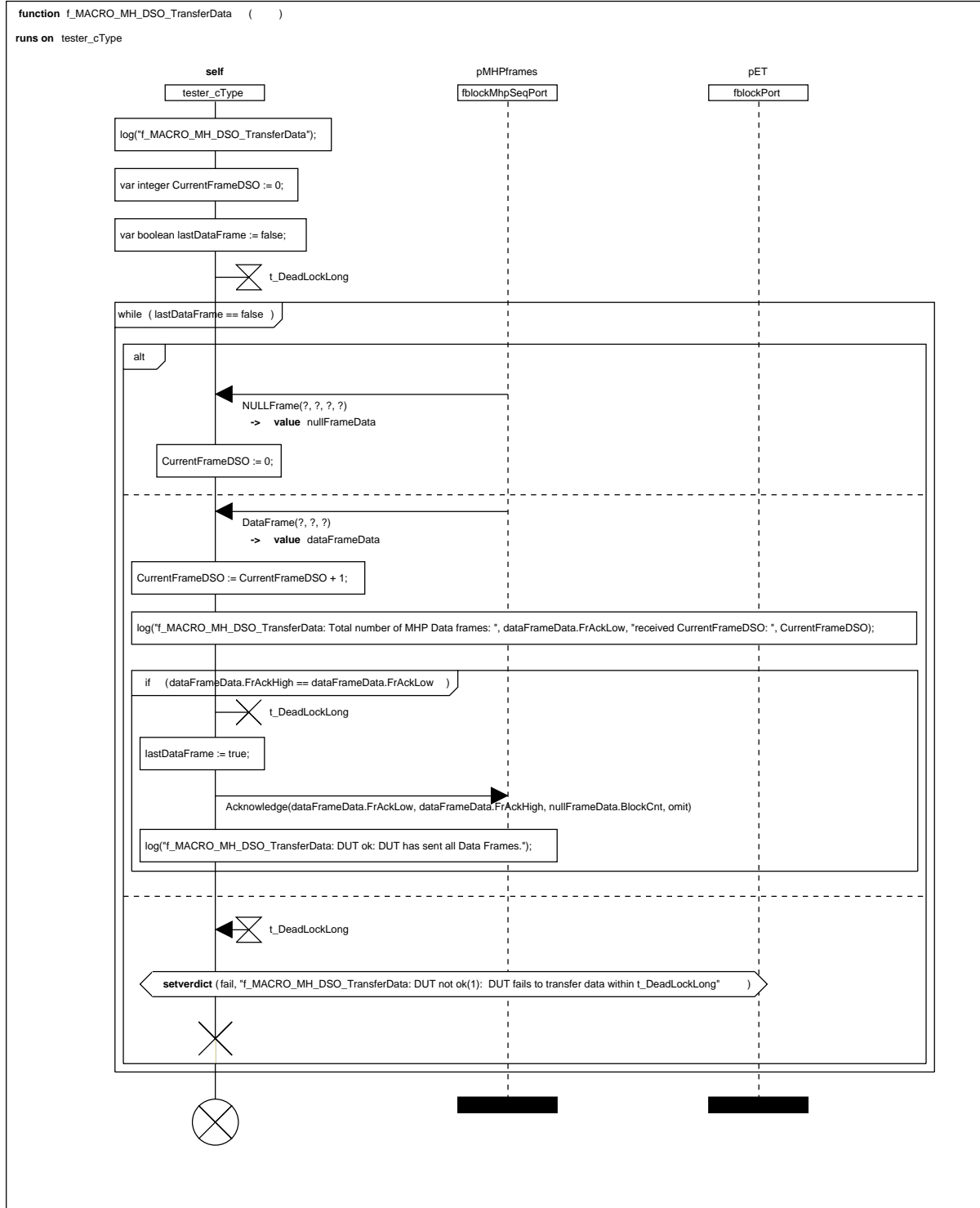
<b>Name of macro</b>	MACRO_MH_DSO_MultiConnection
<b>Reference to GFT</b>	f_MACRO_MH_DSO_MultiConnection (hexstring p_DSI_FktID)
<b>Description</b>	This macro establishes a MHP connection. The connection is set to hold. No return value.
<b>Note</b>	The macro itself only terminates in the error case. It must be terminated by the calling procedure.
<b>Results</b>	<b>DUT not ok (1):</b> The DUT fails to request MHP connection within t_DeadLockMid for parallel DSO connection with DSI FktID. <b>DUT not ok (2):</b> DUT fails to send READY FOR DATA within t_ready for DSI FktID.





## 2.7.8 MACRO\_MH\_DSO\_TransferData

<b>Name of macro</b>	MACRO_MH_DSO_TransferData
<b>Reference to GFT</b>	f_MACRO_MH_DSO_TransferData()
<b>Description</b>	This macro checks the correct number of Data Frames. No return value.
<b>Note</b>	
<b>Results</b>	<b>DUT not ok (1):</b> DUT fails to transfer data within t_DeadLockLong.



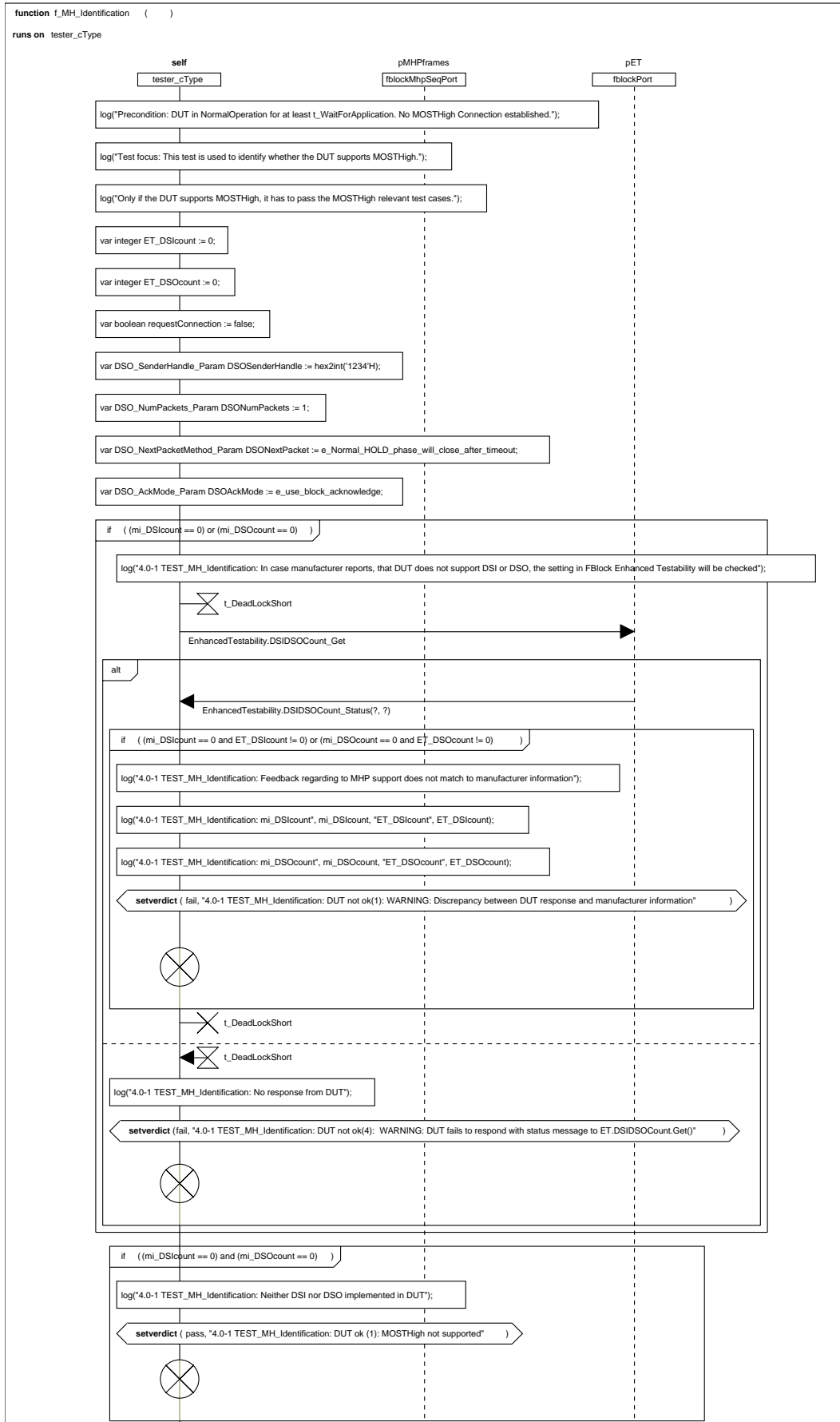
## 3 Extended MOST Core Compliance Tests For MHP

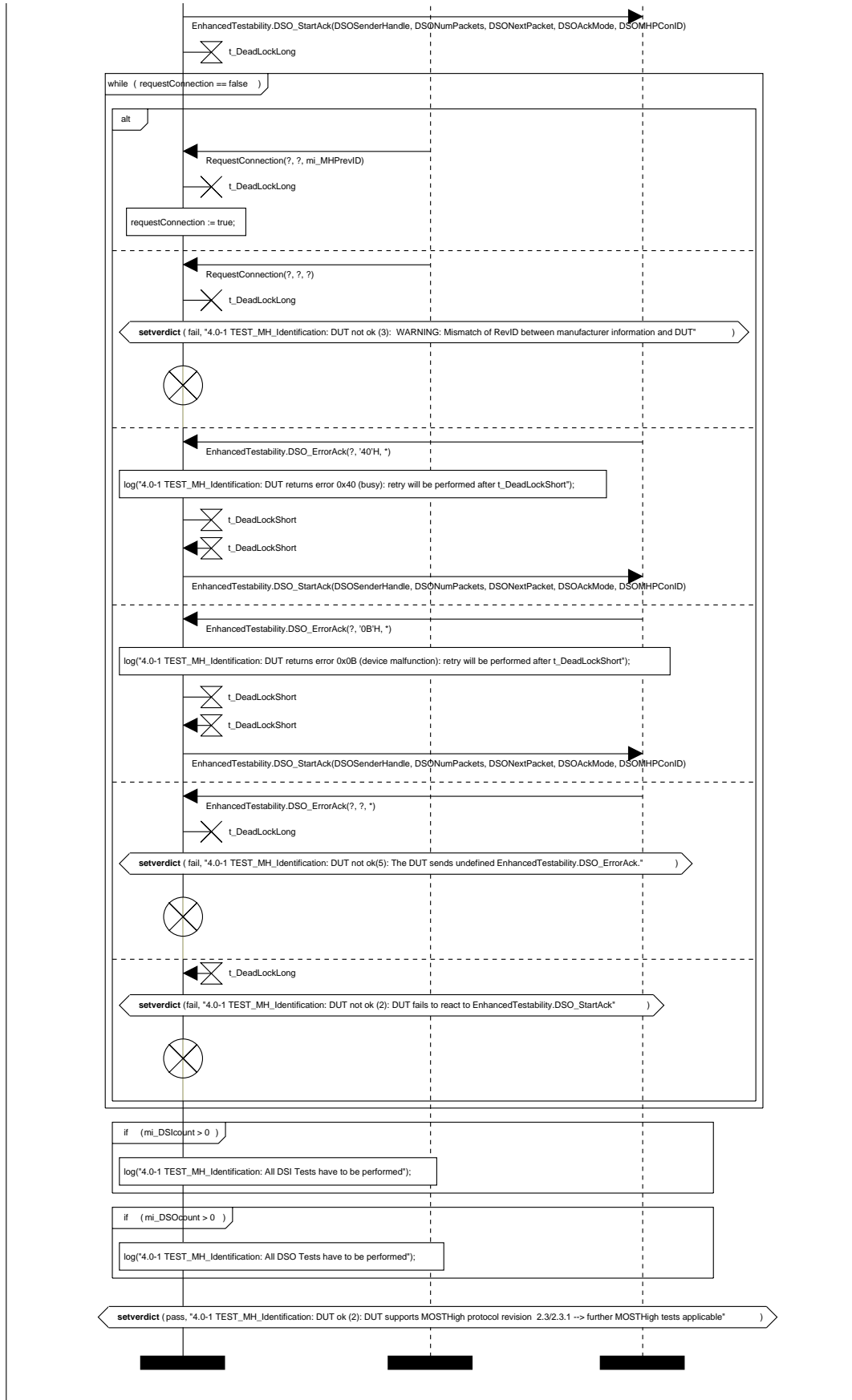
### 3.1 DSI / DSO Identification

#### 3.1.1 TEST\_MH\_Identification

<b>Name of test</b>	TEST_MH_Identification 4.0-1
<b>Reference to GFT</b>	f_MH_Identification ( )
<b>Applicability</b>	All devices
<b>Test focus</b>	This test is used to identify whether the DUT supports MOSTHigh. Only if the DUT supports MOSTHigh, it has to pass the MOSTHigh relevant test cases.
<b>Value of Interest</b>	Supported MOSTHigh protocol revision Implementation of DSI / DSO
<b>Reference to MOST Specification</b>	[3], Fkt DSIDSOCCount; Fkt DSO [2], MH_Gen_EstablishConnection
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	Test results "DUT not ok", containing "WARNING": If the DUT fails with a test result that contains "WARNING", the test could be performed again (after consultation with manufacturer) with updated manufacturer information list.
<b>Results</b>	<b>DUT ok (1):</b> MOSTHigh not supported. <b>DUT ok (2):</b> DUT supports MOSTHigh protocol revision 2.3/2.3.1 --> further MOSTHigh tests applicable. <b>DUT not ok (1):</b> WARNING: Discrepancy between DUT respond and manufacturer information. <b>DUT not ok (2):</b> DUT fails to react to EnhancedTestability.DSO_StartAck. <b>DUT not ok (3):</b> WARNING: Mismatch of RevID between manufacturer information and DUT. <b>DUT not ok (4):</b> WARNING: DUT fails to respond with status message to ET.DSIDSOCCount.Get(). <b>DUT not ok (5):</b> The DUT sends undefined EnhancedTestability.DSO_ErrorAck.







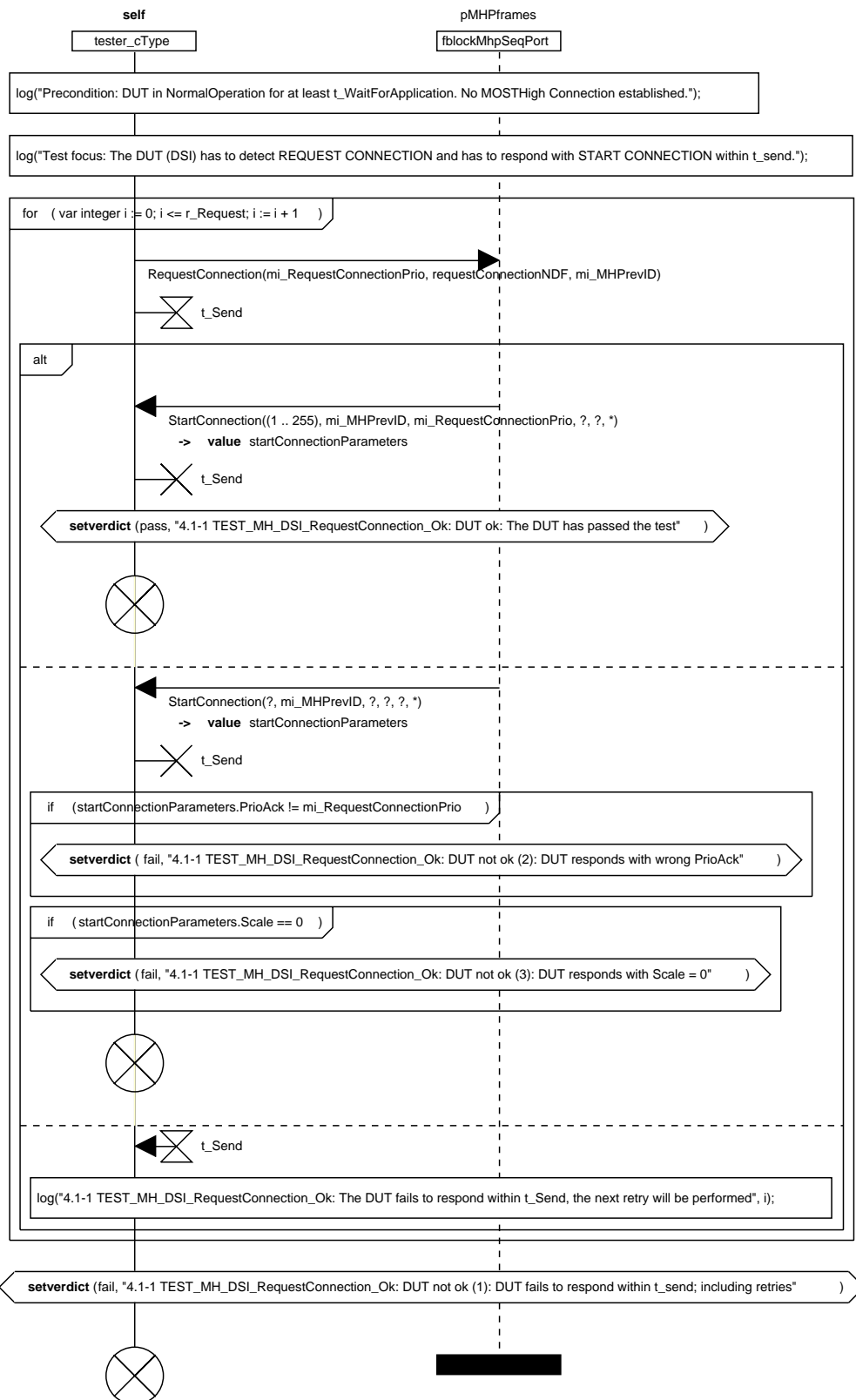
## 3.2 Functional Tests For DSI

### 3.2.1 TEST\_MH\_DSI\_RequestConnection\_Ok

Name of test	TEST_MH_DSI_RequestConnection_Ok 4.1-1
Reference to GFT	f_MH_DSI_RequestConnection_Ok ( )
Applicability	All MHP supporting devices with DSI function
Test focus	The DUT (DSI) has to detect REQUEST CONNECTION and has to respond with START CONNECTION within t_send.
Value of Interest	START CONNECTION, including parameter
Reference to MOST Specification	[2], MH_Gen_EstablishConnection
Experimental set-up	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
Preconditions	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
Note	
Results	<b>DUT ok:</b> The DUT has passed the test <b>DUT not ok (1):</b> DUT fails to respond within t_send; including retries <b>DUT not ok (2):</b> DUT responds with wrong PrioAck <b>DUT not ok (3):</b> DUT responds with Scale = 0

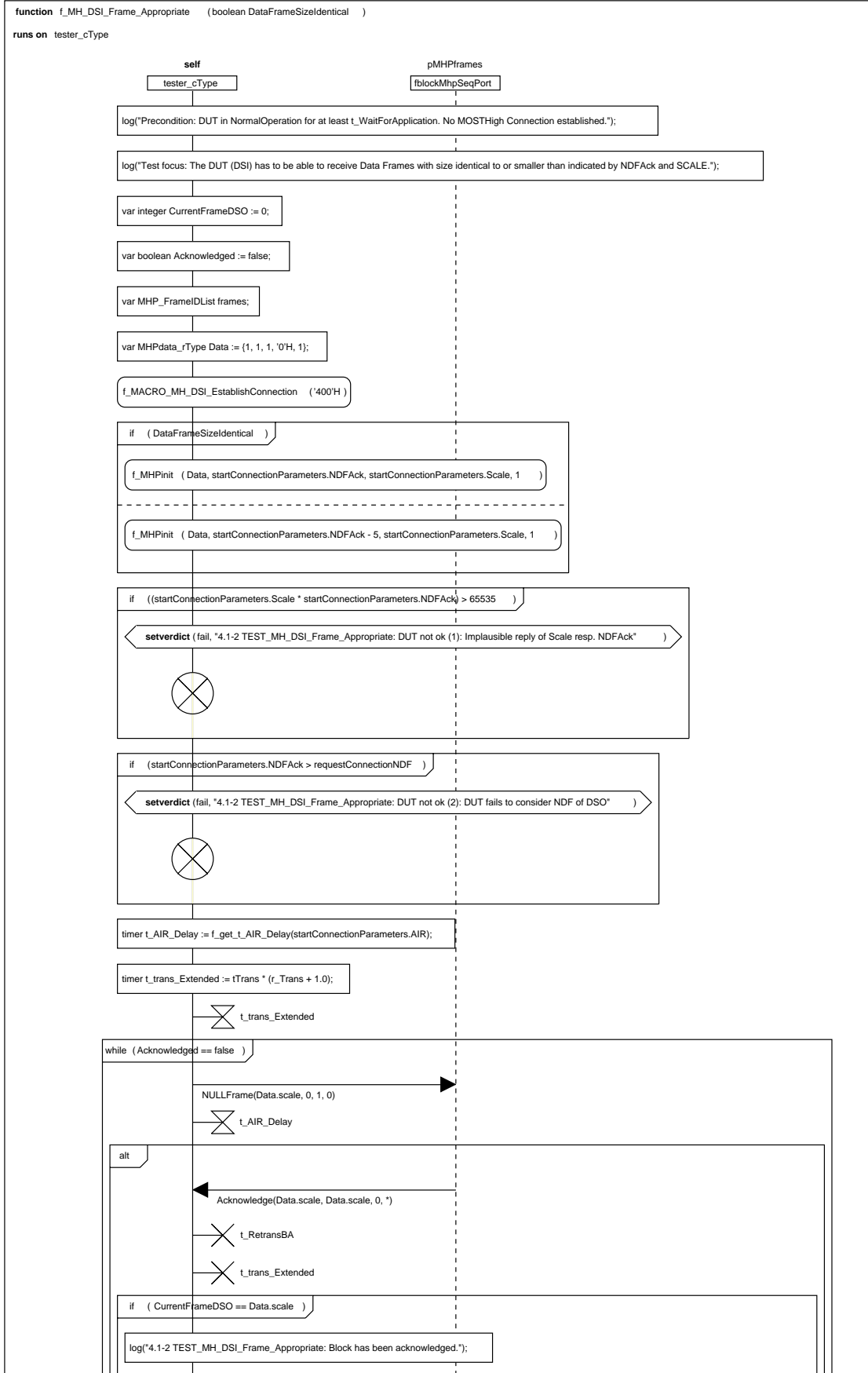
**function** f\_MH\_DSI\_RequestConnection\_Ok ( )

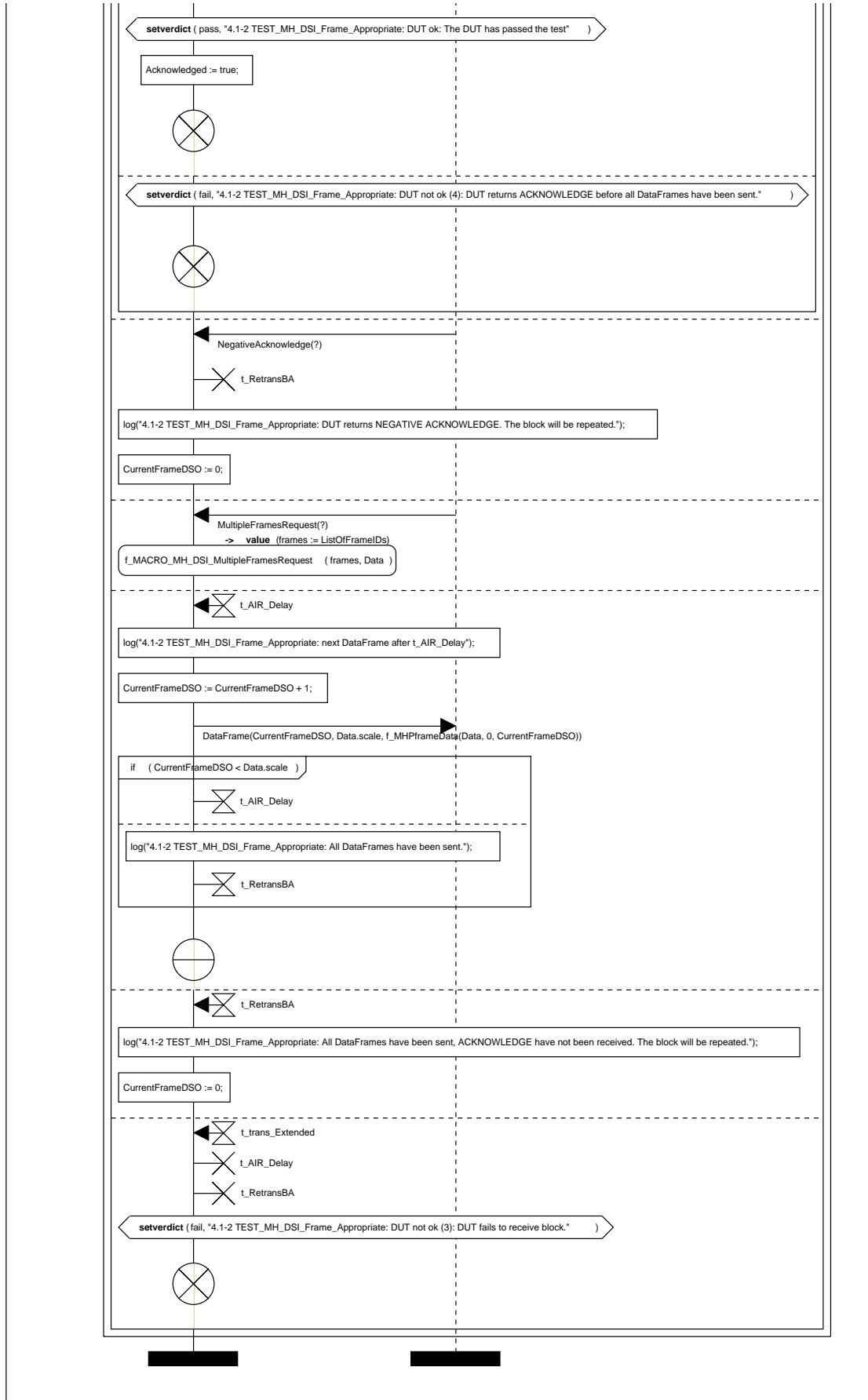
**runs on** tester\_cType



### 3.2.2 TEST\_MH\_DSI\_Frame\_Appropriate

<b>Name of test</b>	TEST_MH_DSI_Frame_Appropriate 4.1-2
<b>Reference to GFT</b>	f_MH_DSI_Frame_Appropriate (boolean DataFrameSizeIdentical)
<b>Applicability</b>	All MHP supporting devices with DSI function
<b>Test focus</b>	The DUT (DSI) has to be able to receive DataFrames with size identical to or smaller than indicated by NDFAck and SCALE.
<b>Value of Interest</b>	Acknowledge
<b>Reference to MOST Specification</b>	[2] MH_Gen_DataTransmissionBA
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	<p>Test has to be performed twice:</p> <ol style="list-style-type: none"> <li>1. DataFrameSizeIdentical = false: → data frame size smaller than NDFAck: Number of data bytes of all frames: NDFAck - 5 Total amount of user data in block = (NDFAck - 5) * SCALE <i>Remark: "-5" because of quadlet oriented data transfer</i></li> <li>2. DataFrameSizeIdentical = true: → data frame size identical to NDFAck: Number of frames per block: SCALE Total amount of user data in block = NDFAck * SCALE</li> </ol> <p>In both cases, test has to be performed with block acknowledge mode (AckMode=BlockAcknowledge).</p>
<b>Results</b>	<p><b>DUT ok:</b> The DUT has passed the test.</p> <p><b>DUT not ok (1):</b> Implausible reply of Scale resp. NDFAck.</p> <p><b>DUT not ok (2):</b> DUT fails to consider NDF of DSO.</p> <p><b>DUT not ok (3):</b> DUT fails to receive block.</p> <p><b>DUT not ok (4):</b> DUT returns ACKNOWLEDGE before all DataFrames have been sent.</p>

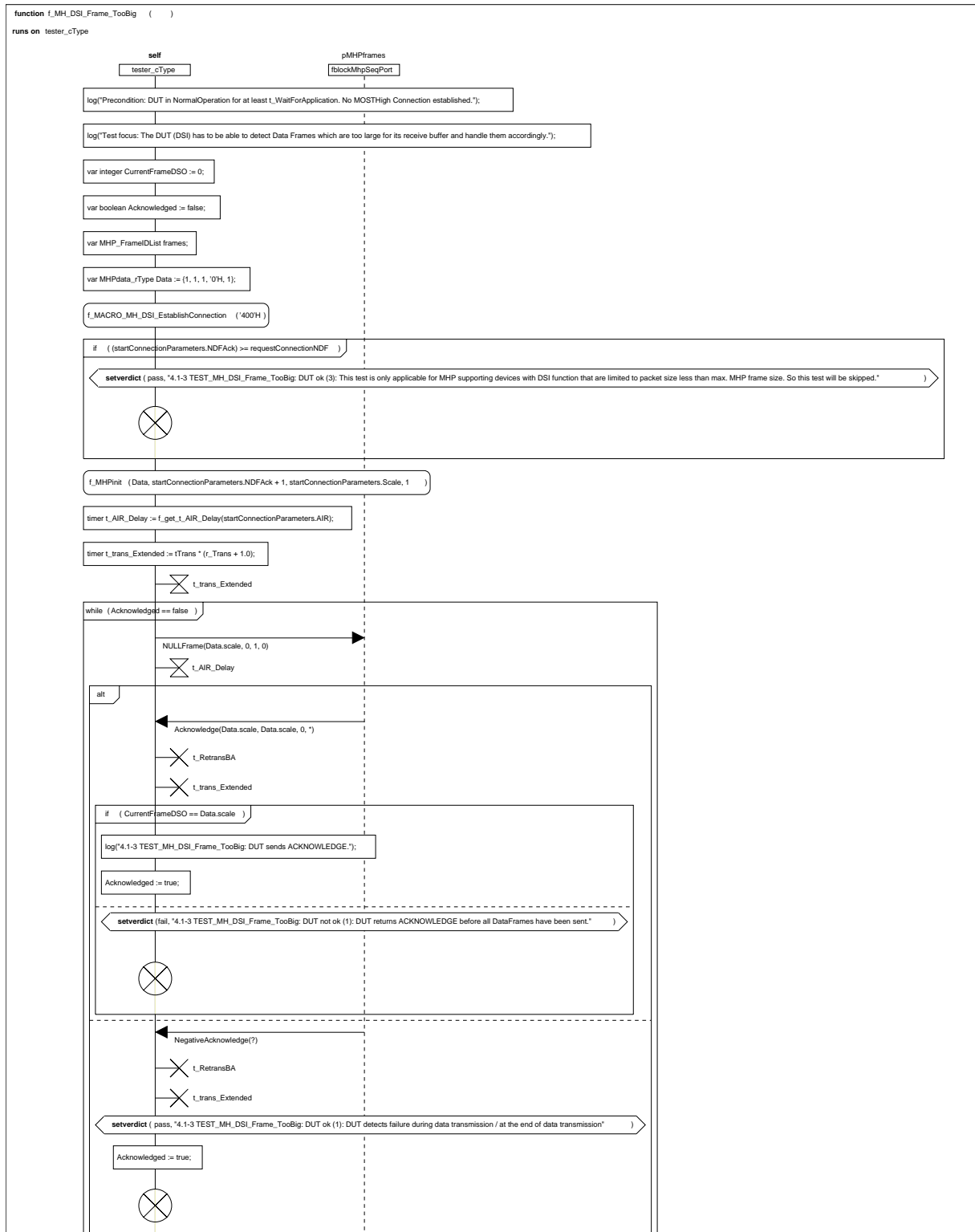


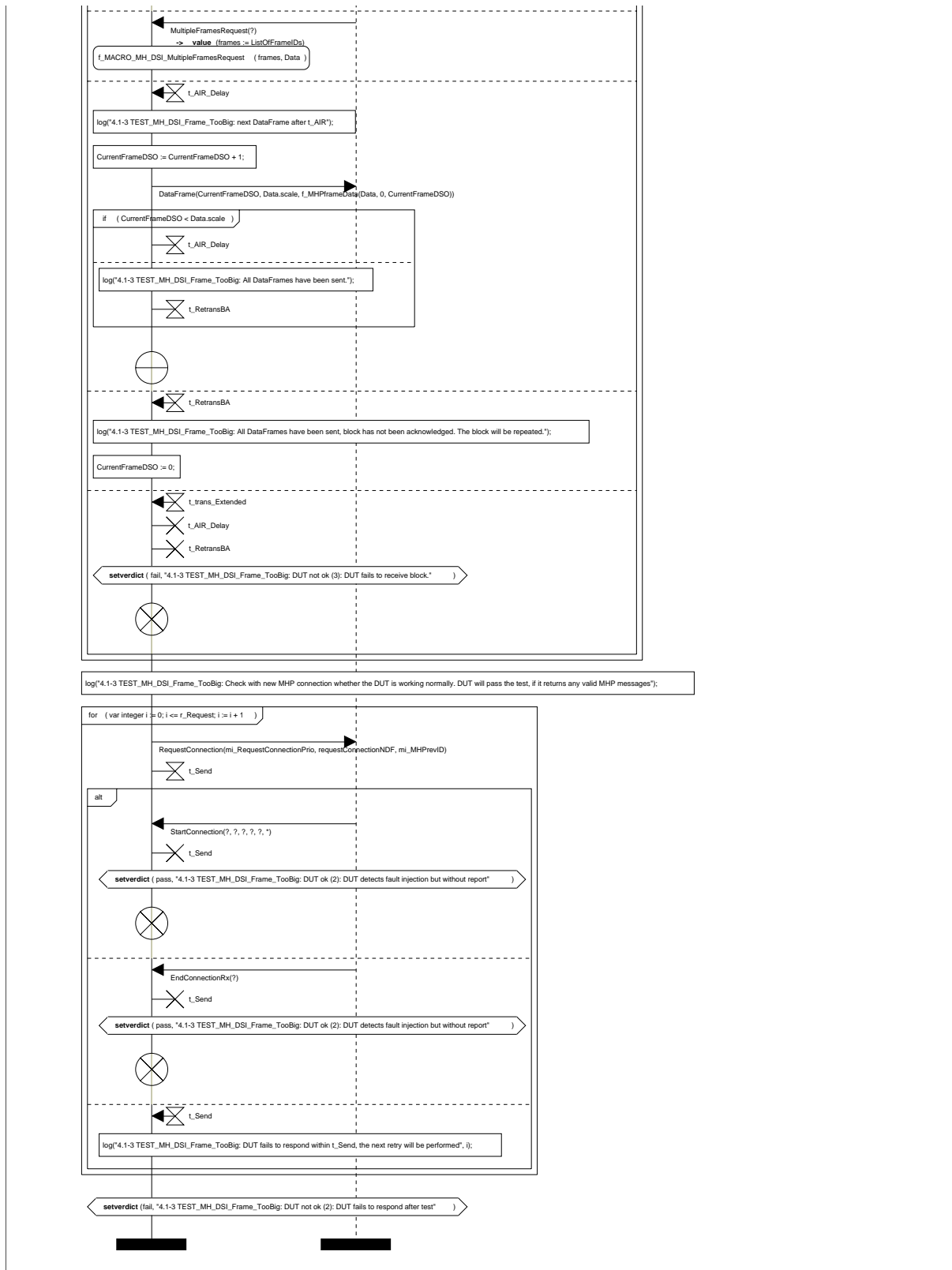


### 3.2.3 TEST\_MH\_DSI\_Frame\_TooBig

<b>Name of test</b>	TEST_MH_DSI_Frame_TooBig 4.1-3
<b>Reference to GFT</b>	f_MH_DSI_Frame_TooBig ( )
<b>Applicability</b>	All MHP supporting devices with DSI function
<b>Test focus</b>	The DUT (DSI) has to be able to detect DataFrames which are too large for its receive buffer and handle them accordingly.
<b>Value of Interest</b>	NegativeAcknowledge
<b>Reference to MOST Specification</b>	[2], para 4.1.2.1 and para 4.1.2.4
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	The DUT could detect / report the error during transmission or at the end of the transmission (with reception of last frame)  In both cases, the DUT will pass the test.
<b>Results</b>	<b>DUT ok (1):</b> DUT detects failure during data transmission / at the end of data transmission. <b>DUT ok (2):</b> DUT detects fault injection but without report. <b>DUT ok (3):</b> This test is only applicable for MHP supporting devices with DSI function that are limited to packet size less than max. MHP frame size. So this test will be skipped. <b>DUT not ok (1):</b> DUT returns ACKNOWLEDGE before all DataFrames have been sent. <b>DUT not ok (2):</b> DUT fails to respond after test. <b>DUT not ok (3):</b> DUT fails to receive block.

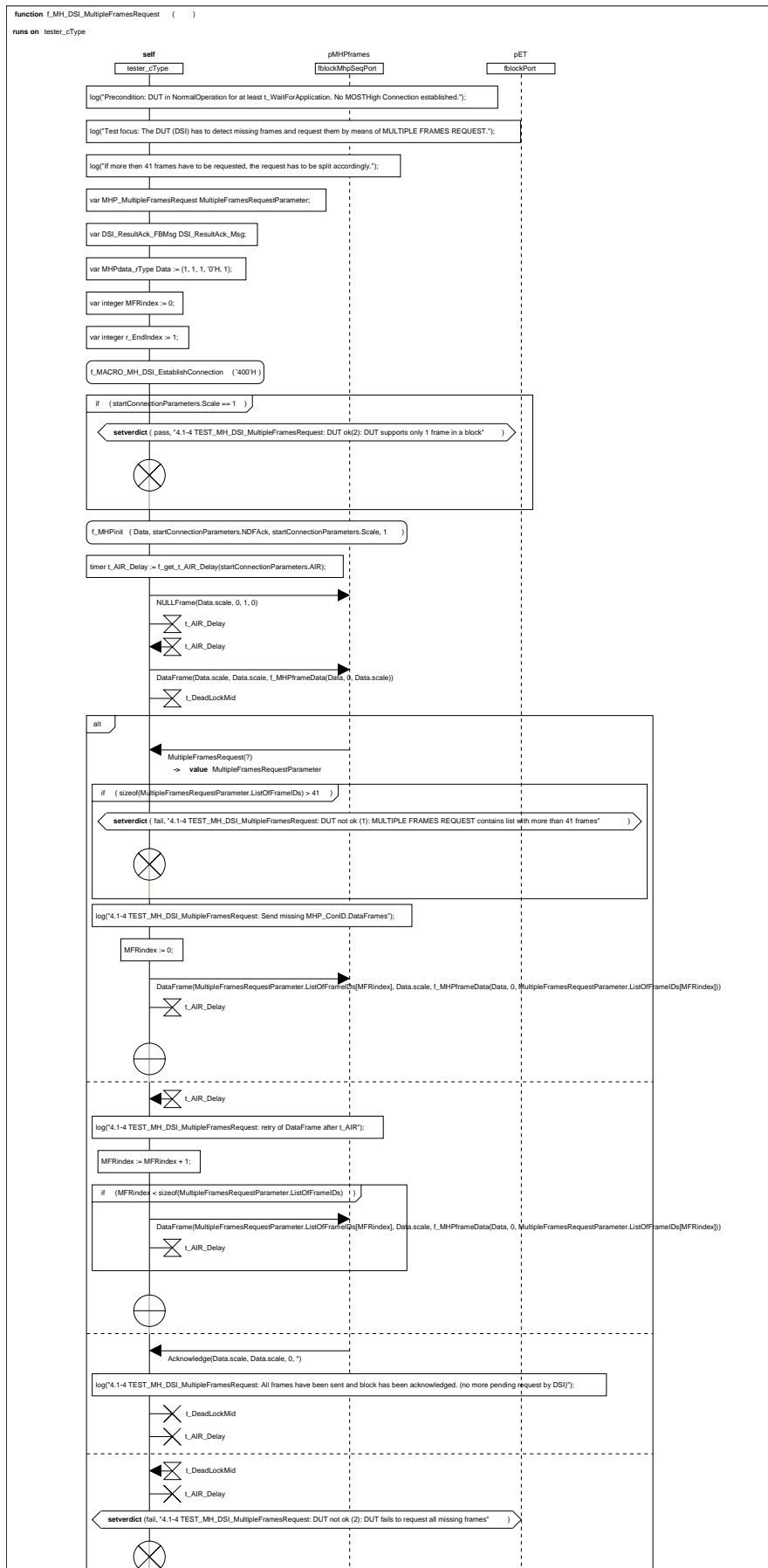


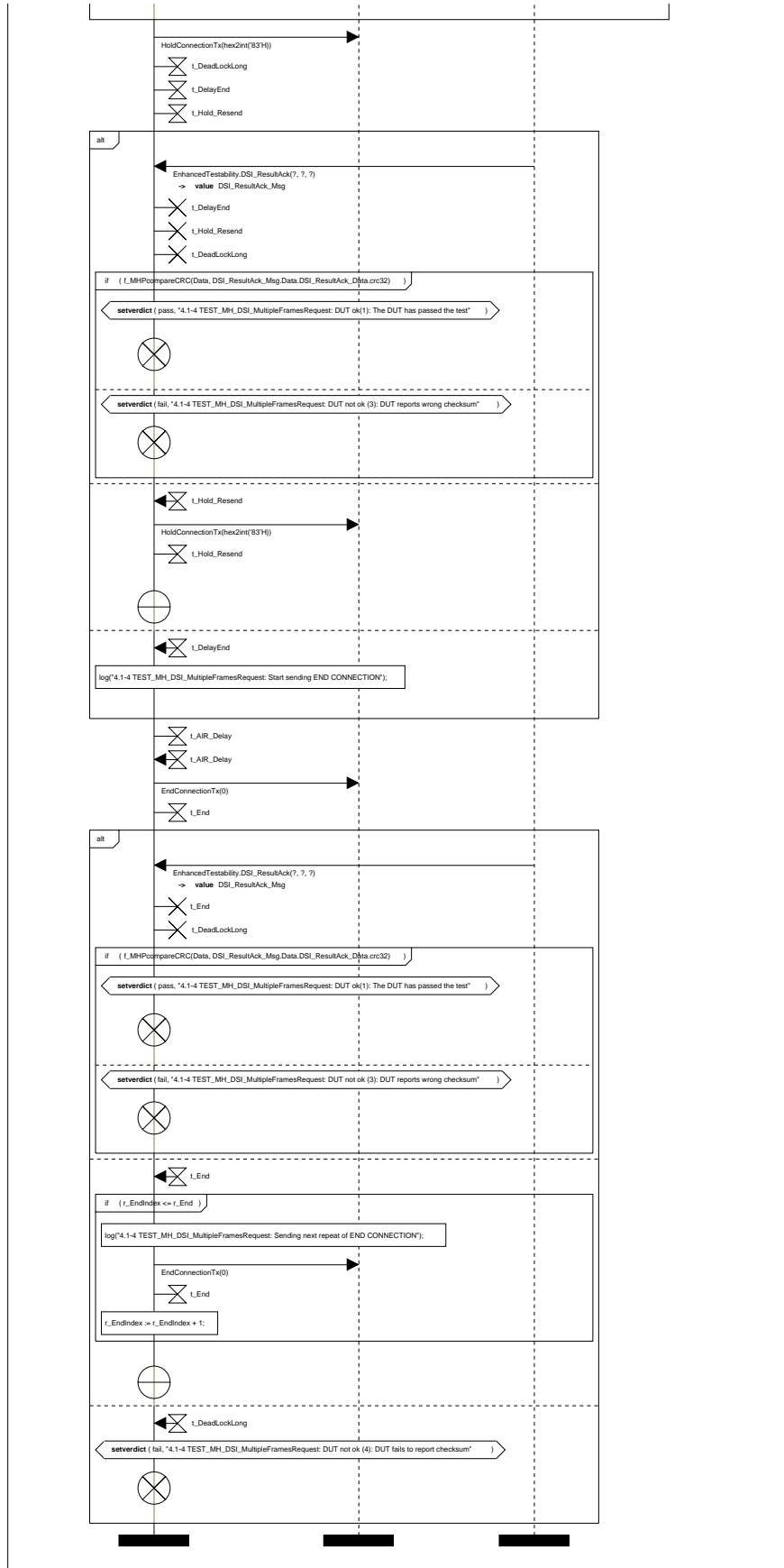




### 3.2.4 TEST\_MH\_DSI\_MultipleFramesRequest

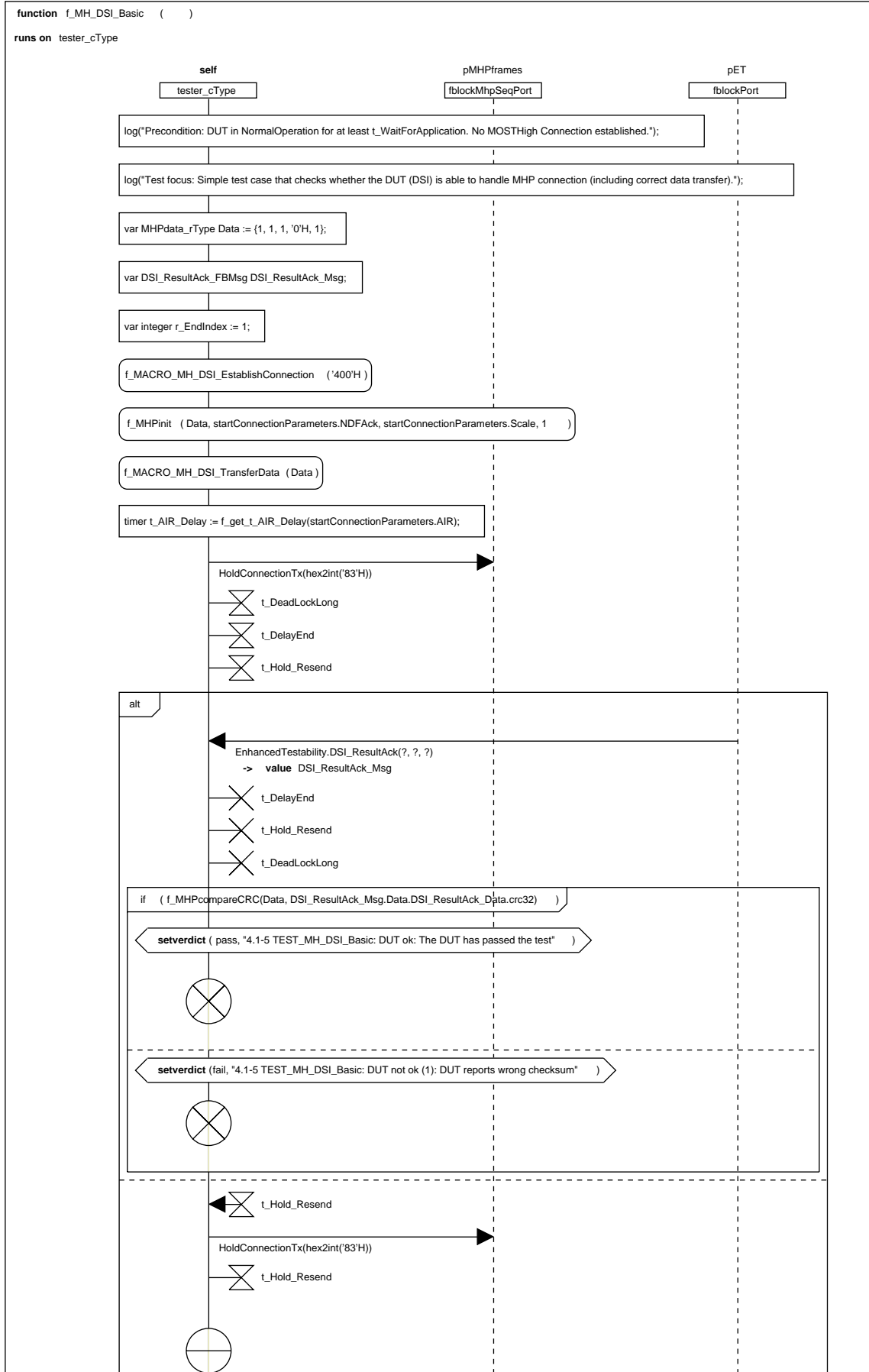
<b>Name of test</b>	TEST_MH_DSI_MultipleFramesRequest 4.1-4
<b>Reference to GFT</b>	f_MH_DSI_MultipleFramesRequest ( )
<b>Applicability</b>	All MHP supporting devices with DSI function
<b>Test focus</b>	The DUT (DSI) has to detect missing frames and request them by means of MULTIPLE FRAMES REQUEST. If more then 41 frames have to be requested, the request has to be split accordingly.
<b>Value of Interest</b>	MULTIPLE FRAMES REQUEST of DUT
<b>Reference to MOST Specification</b>	[2]; MH_Sc_MFRAfterTimeout MH_Sc_MFRAfterLastFrame MH_Sc_MergingMFRs
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	Timings and retries are not focus of this test. The whole test has to be performed within t_DeadLockLong, otherwise the DUT fails the test. The tester has to perform retries according to MOSTHigh Specification.
<b>Results</b>	<b>DUT ok (1):</b> The DUT has passed the test. <b>DUT ok (2):</b> DUT supports only 1 frame in a block. <b>DUT not ok (1):</b> MULTIPLE FRAMES REQUEST contains list with more than 41 frames <b>DUT not ok (2):</b> DUT fails to request all missing frames <b>DUT not ok (3):</b> DUT reports wrong checksum <b>DUT not ok (4):</b> DUT fails to report checksum

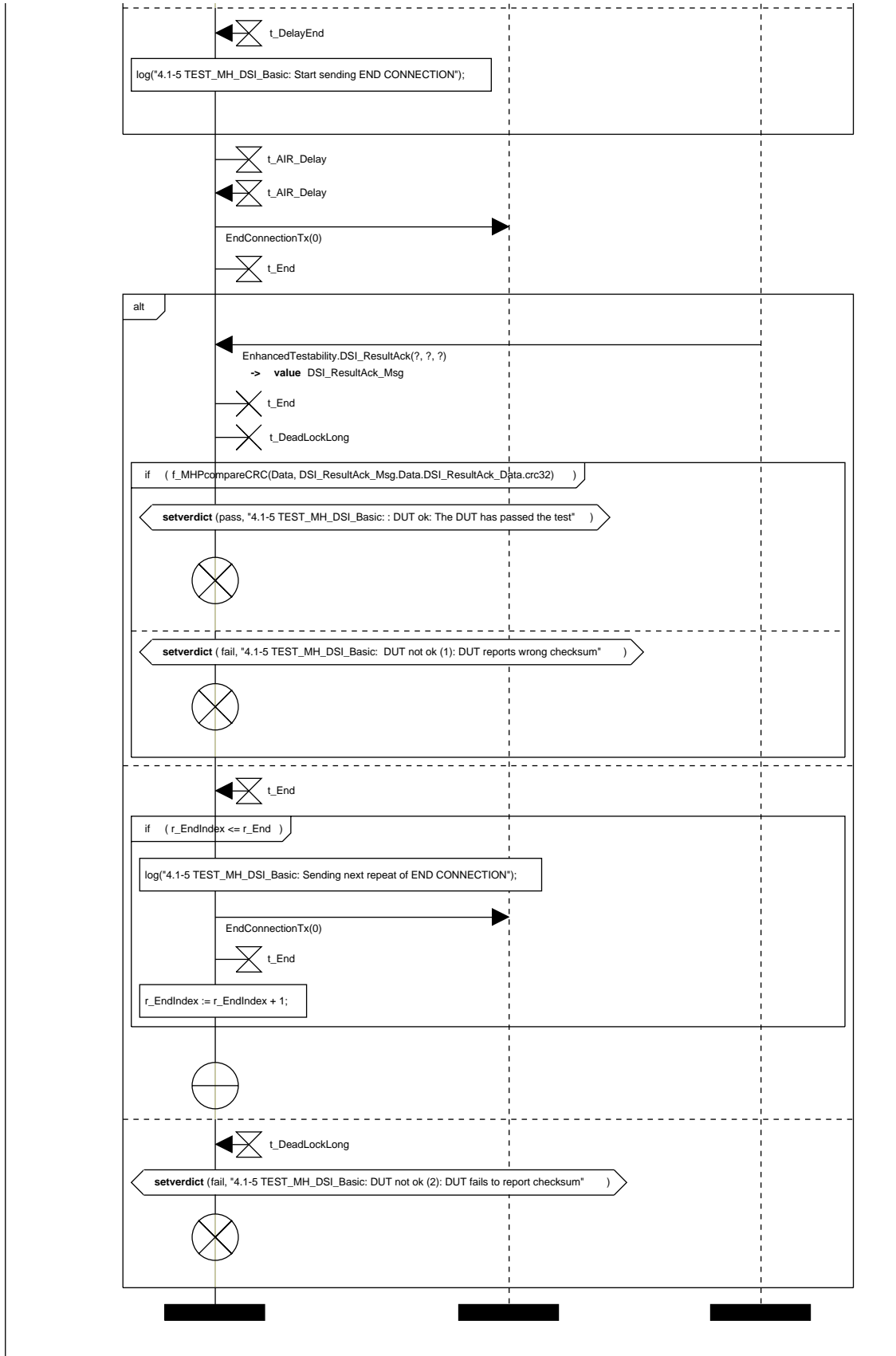




### 3.2.5 TEST\_MH\_DSI\_Basic

<b>Name of test</b>	TEST_MH_DSI_Basic 4.1-5
<b>Reference to GFT</b>	f_MH_DSI_Basic ()
<b>Applicability</b>	All MHP supporting devices with DSI function
<b>Test focus</b>	Simple test case that checks whether the DUT (DSI) is able to handle MHP connection (including correct data transfer).
<b>Value of Interest</b>	DSI.ResultAck
<b>Reference to MOST Specification</b>	[2], MH_Gen_BasicFlow
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	The test has to be performed with block acknowledge mode (AckMode=BlockAcknowledge).  Definition of test pattern to be sent to DUT: The first two bytes of the data pattern contain the SenderHandle. The first data byte shall be 0x00, the seconds shall be 0x01, etc. The value is increased with every data byte. If 0xFF is reached, the next data byte starts with 0x00 again.
<b>Results</b>	<b>DUT ok:</b> The DUT has passed the test. <b>DUT not ok (1):</b> DUT reports wrong checksum. <b>DUT not ok (2):</b> DUT fails to report checksum.

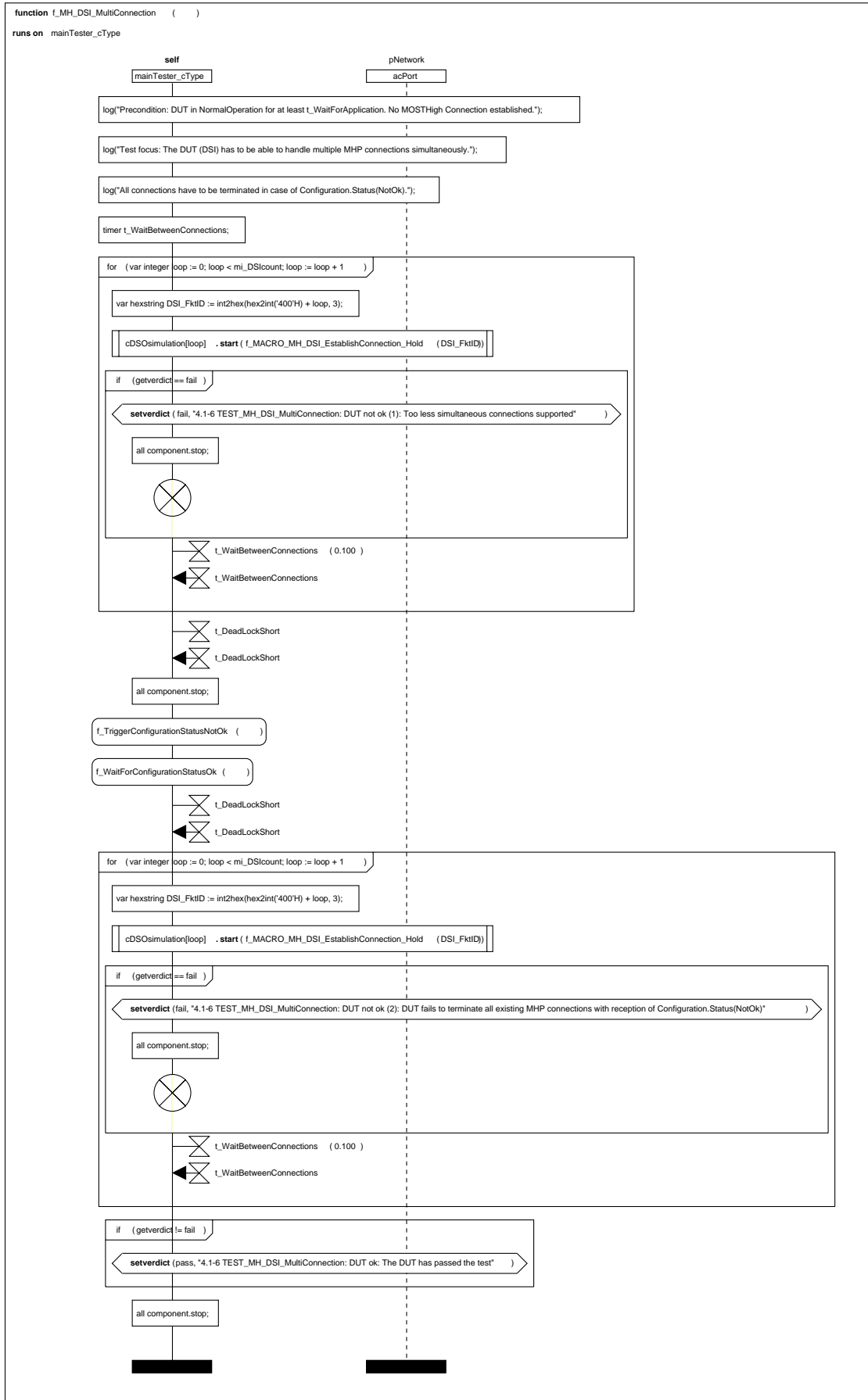






### 3.2.6 TEST\_MH\_DSI\_MultiConnection

<b>Name of test</b>	TEST_MH_DSI_MultiConnection 4.1-6
<b>Reference to GFT</b>	f_MH_DSI_MultiConnection ( )
<b>Applicability</b>	All MHP supporting devices with DSI function
<b>Test focus</b>	The DUT (DSI) has to be able to handle multiple MHP connections simultaneously. All connections have to be terminated in case of Configuration.Status(NotOk).
<b>Value of Interest</b>	Number of parallel established connections. Behavior at Configuration.Status(NotOk) according to existing MHP connections.
<b>Reference to MOST Specification</b>	[2]; para 7.1.8
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	The tester has to support up to 30 simultaneous MHP connections. In case mi_DSICount exceeds this number, mi_DSICount will be handled as if it was defined with the value 30.  The timer t_WaitBetweenConnections (= 100ms) specifies the pause, the tester has to wait between triggering new connections.
<b>Results</b>	<b>DUT ok:</b> The DUT has passed the test. <b>DUT not ok (1):</b> Too less simultaneous connections supported. <b>DUT not ok (2):</b> DUT fails to terminate all existing MHP connections with reception of Configuration.Status(NotOk).



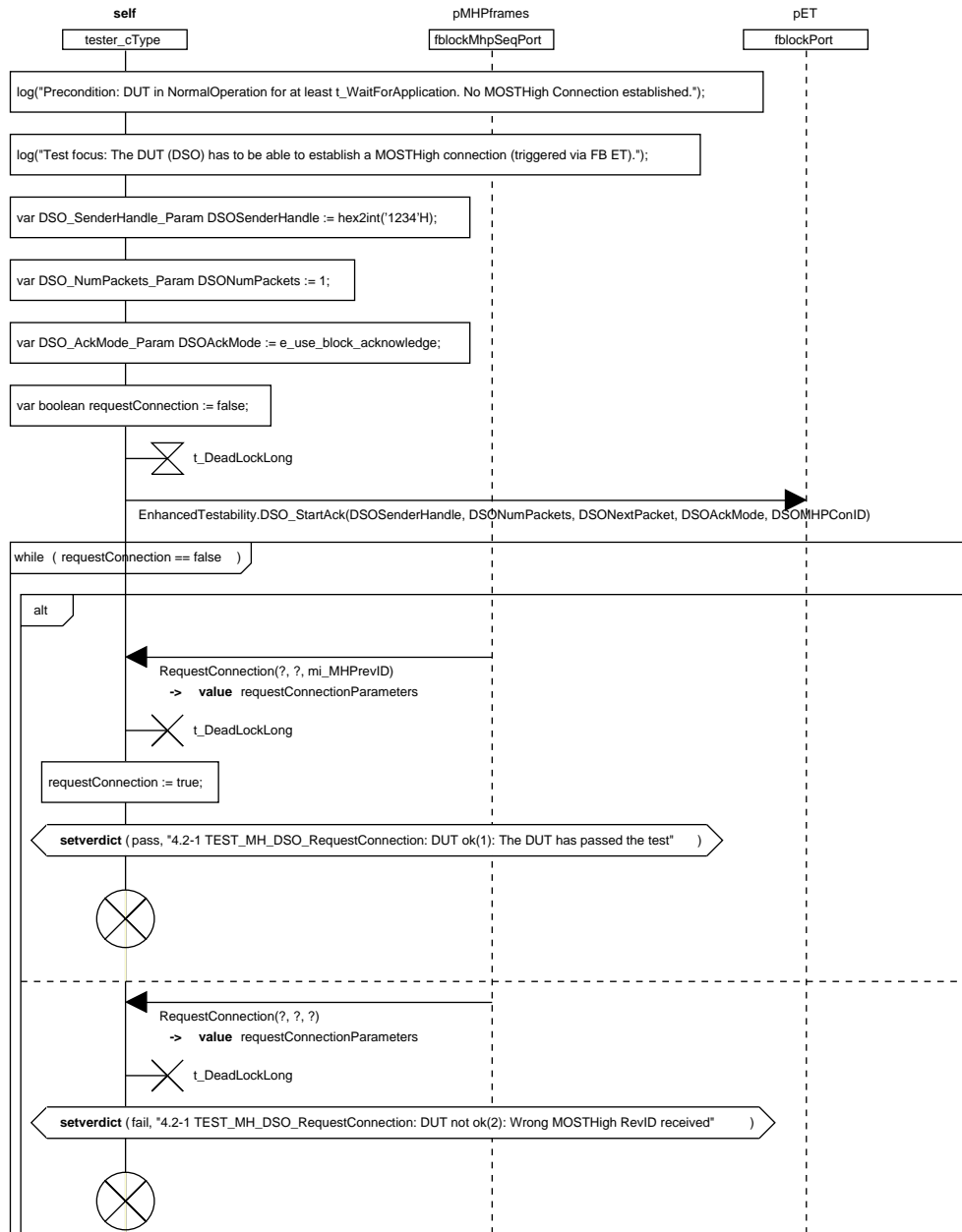
## 3.3 Functional Tests For DSO

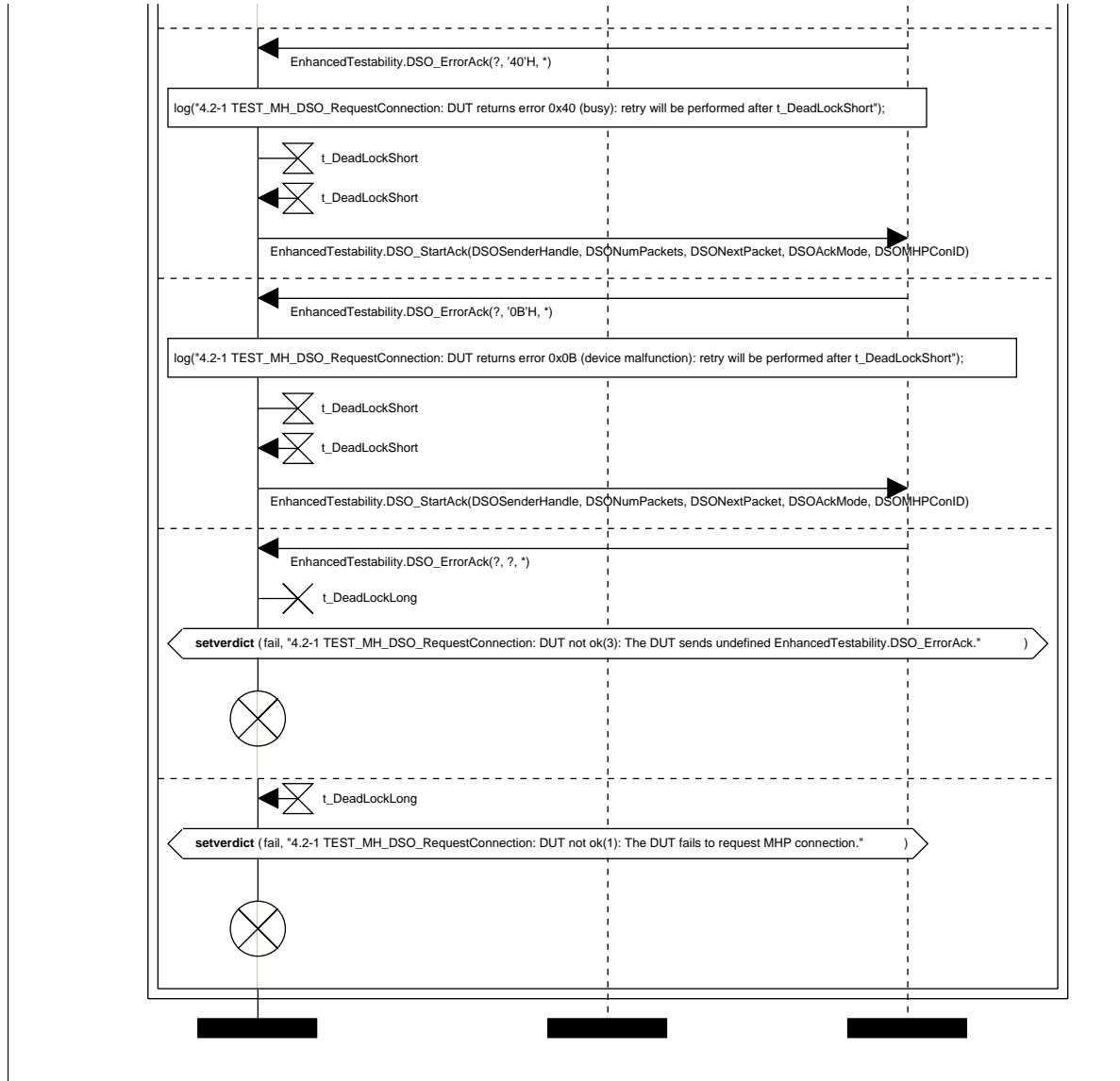
### 3.3.1 TEST\_MH\_DSO\_RequestConnection

<b>Name of test</b>	TEST_MH_DSO_RequestConnection 4.2-1
<b>Reference to GFT</b>	f_MH_DSO_RequestConnection (in DSO_NextPacketMethod_Param DSONextPacket)
<b>Applicability</b>	All MHP supporting devices with DSO function
<b>Test focus</b>	The DUT (DSO) has to be able to establish a MOSTHigh connection (triggered via FB ET).
<b>Value of Interest</b>	REQUEST CONNECTION
<b>Reference to MOST Specification</b>	[2], MH_Gen_EstablishConnection
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	<p>Test has to be performed with block acknowledge mode only (AckMode=BlockAcknowledge).</p> <p>Test has to be performed with the following specified "NextPacketMethod":</p> <ul style="list-style-type: none"> <li>• Hold (DSONextPacket = e_Normal_HOLD_phase_will_close_after_timeout)</li> <li>• Terminate (DSONextPacket = e_Terminate_the_connection_and_open_a_new_one_if_necessary)</li> <li>• SendNext (if supported by DUT) (DSONextPacket = e_Try_to_send_next_packet_without_a_hold_optional)</li> </ul>
<b>Results</b>	<p><b>DUT ok (1):</b> The DUT has passed the test.</p> <p><b>DUT not ok (1):</b> The DUT fails to request MHP connection.</p> <p><b>DUT not ok (2):</b> Wrong MOSTHigh RevID received.</p> <p><b>DUT not ok (3):</b> The DUT sends undefined EnhancedTestability.DSO_ErrorAck.</p>

**function** f\_MH\_DSO\_RequestConnection ( in DSO\_NextPacketMethod\_Param DSONextPacket )

**runs on** tester\_cType



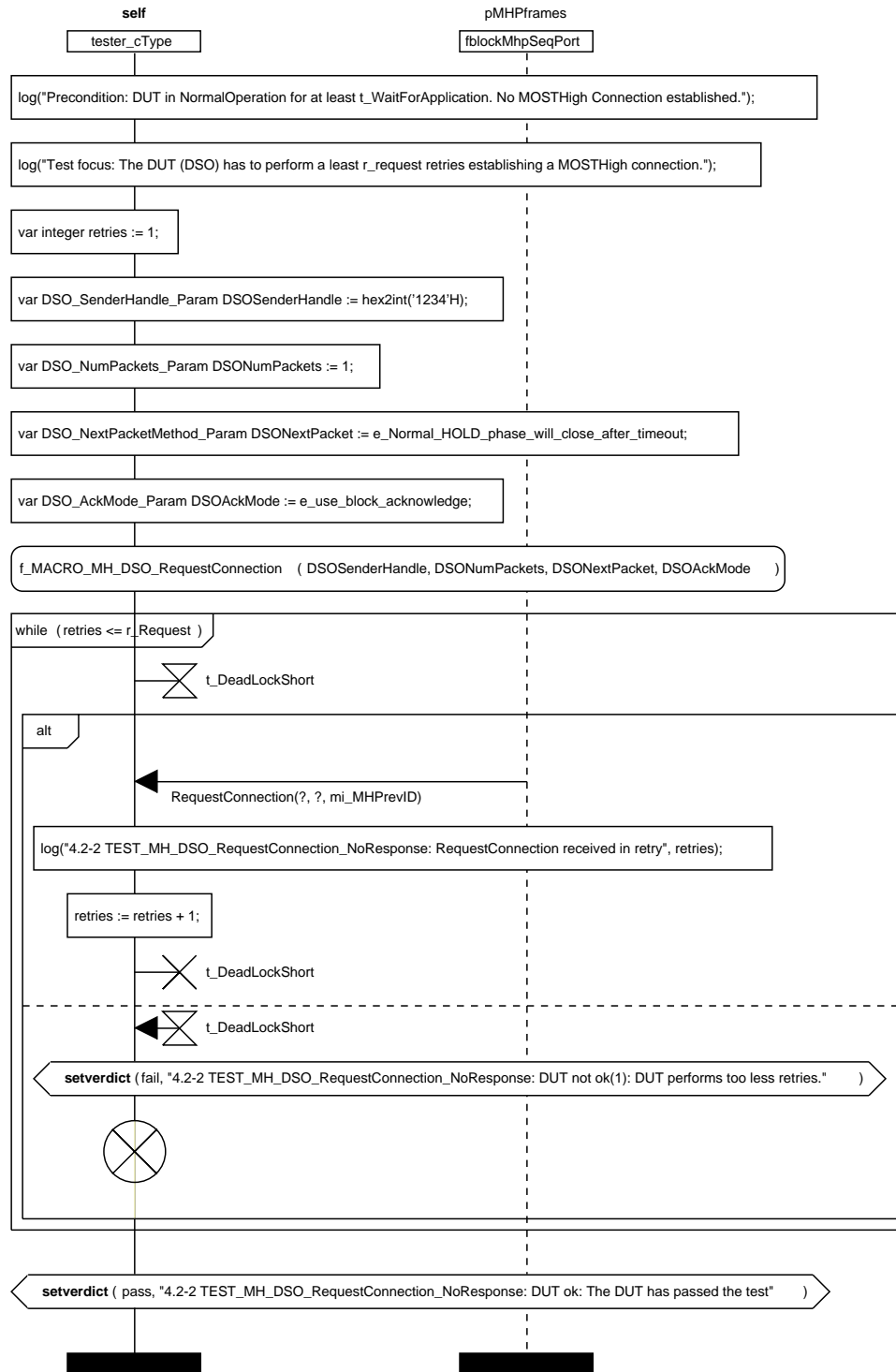


### 3.3.2 TEST\_MH\_DSO\_RequestConnection\_NoResponse

<b>Name of test</b>	TEST_MH_DSO_RequestConnection_NoResponse 4.2-2
<b>Reference to GFT</b>	f_MH_DSO_RequestConnection_NoResponse ( )
<b>Applicability</b>	All MHP supporting devices with DSO function
<b>Test focus</b>	The DUT (DSO) has to perform at least r_request retries establishing a MOSTHigh connection.
<b>Value of Interest</b>	r_Request
<b>Reference to MOST Specification</b>	[2], MH_Sc_RequestFail
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	Test has to be performed with NextPacketMethod=Hold and AckMode=BlockAcknowledge.
<b>Results</b>	<b>DUT ok:</b> The DUT has passed the test. <b>DUT not ok (1):</b> DUT performs too less retries.

**function** f\_MH\_DSO\_RequestConnection\_NoResponse ( )

**runs on** tester\_cType



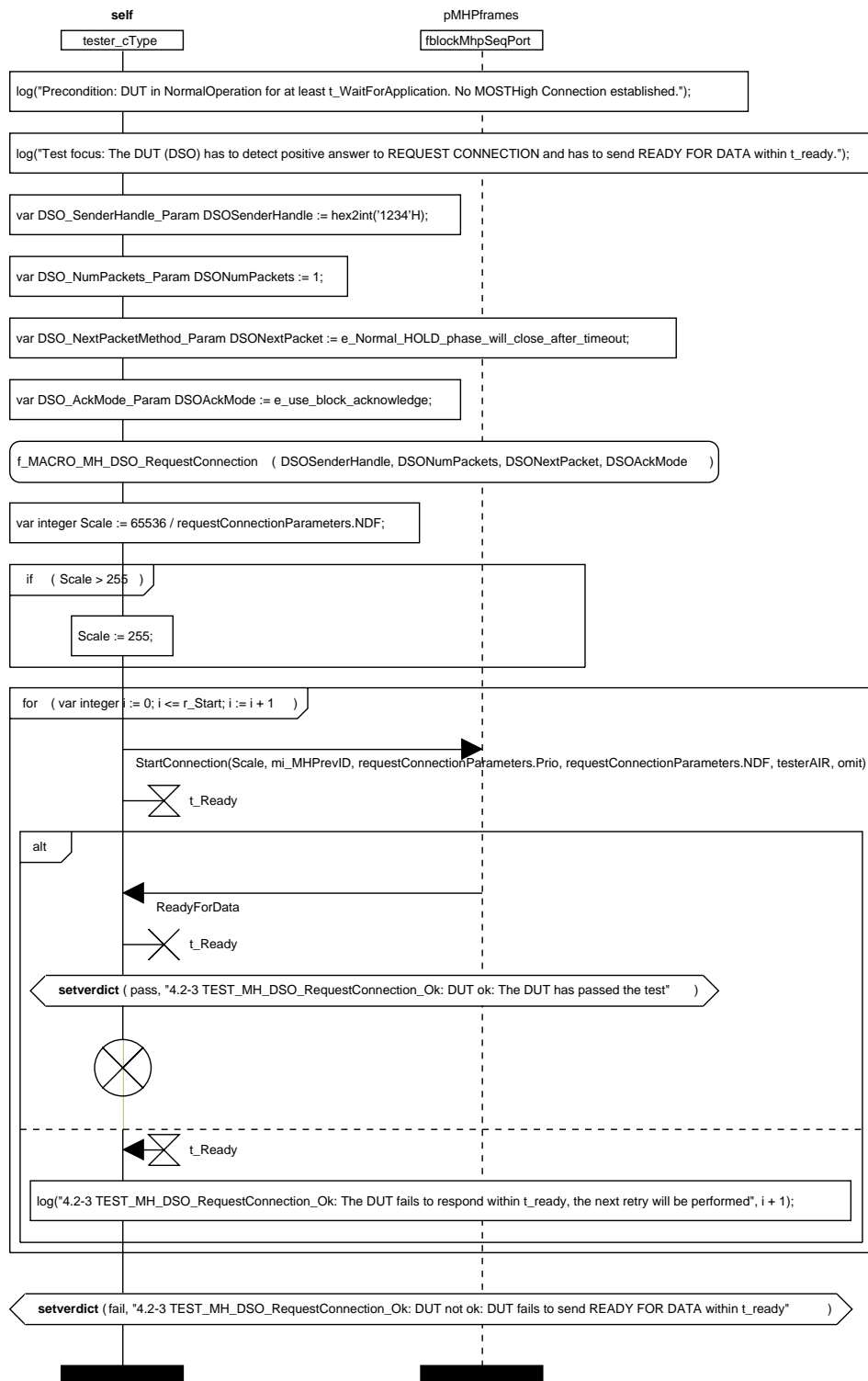
### 3.3.3 TEST\_MH\_DSO\_RequestConnection\_Ok

<b>Name of test</b>	TEST_MH_DSO_RequestConnection_Ok 4.2-3
<b>Reference to GFT</b>	f_MH_DSO_RequestConnection_Ok ( )
<b>Applicability</b>	All MHP supporting devices with DSO function
<b>Test focus</b>	The DUT (DSO) has to detect positive answer to REQUEST CONNECTION and has to send READY FOR DATA within t_ready.
<b>Value of Interest</b>	READY FOR DATA
<b>Reference to MOST Specification</b>	[2], MH_Gen_EstablishConnection
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	Test has to be performed with NextPacketMethod=Hold and AckMode=BlockAcknowledge.
<b>Results</b>	<b>DUT ok:</b> The DUT has passed the test. <b>DUT not ok:</b> DUT fails to send READY FOR DATA within t_ready.



```
function f_MH_DSO_RequestConnection_Ok ( )
```

```
runs on tester_cType
```

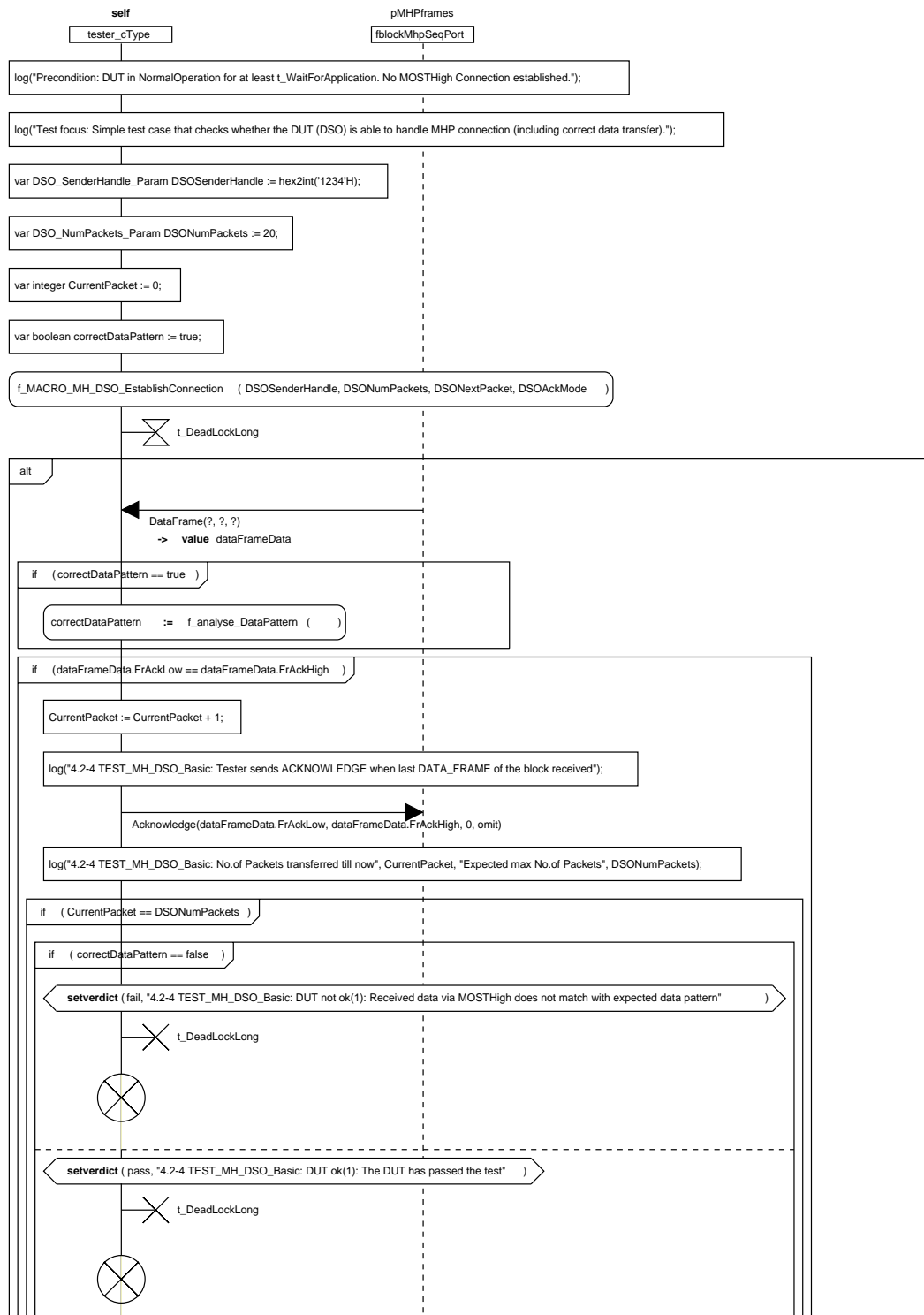


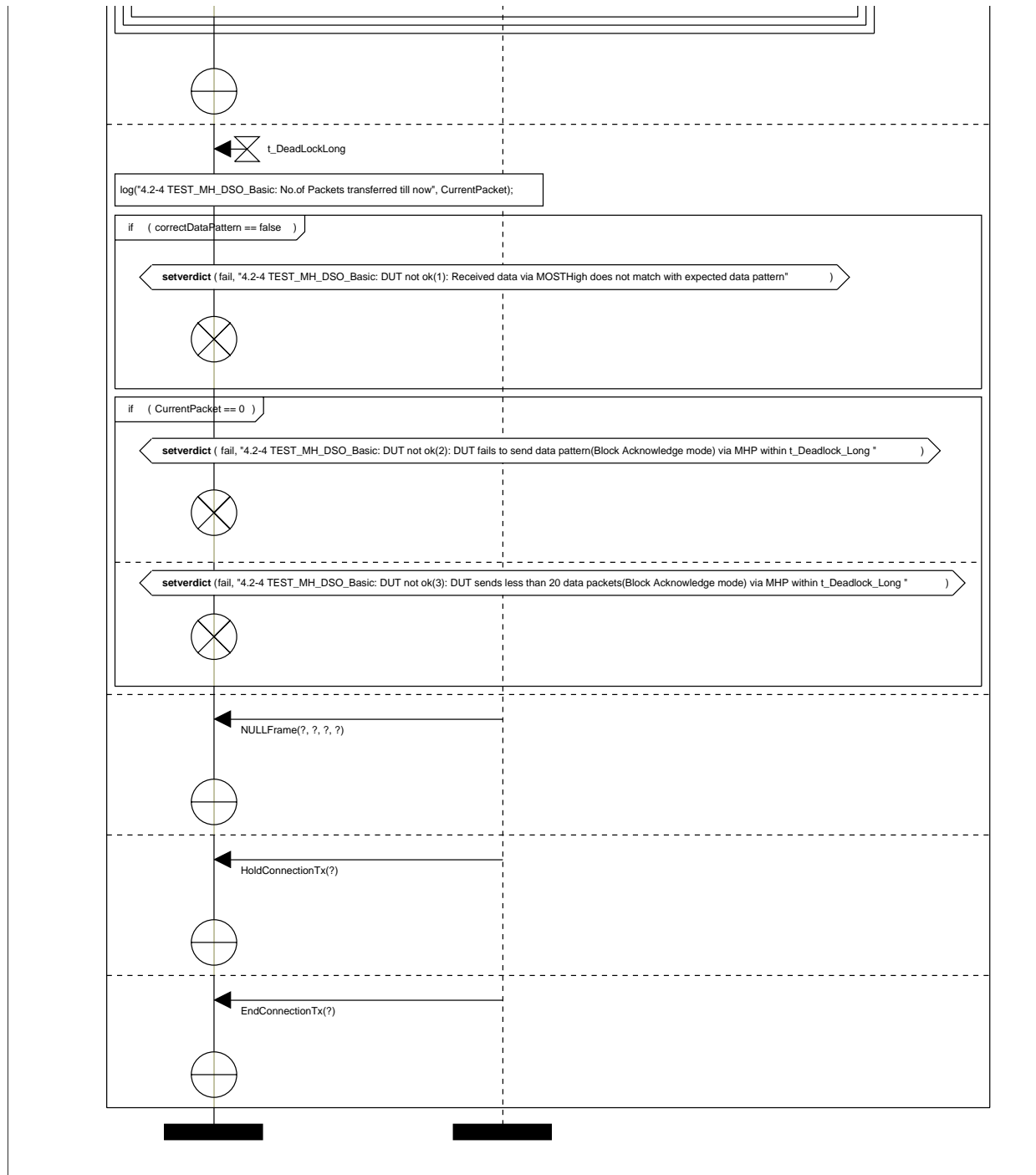
### 3.3.4 TEST\_MH\_DSO\_Basic

<b>Name of test</b>	TEST_MH_DSO_Basic 4.2-4
<b>Reference to GFT</b>	f_MH_DSO_Basic (in DSO_NextPacketMethod_Param DSONextPacket, in DSO_AckMode_Param DSOAckMode)
<b>Applicability</b>	All MHP supporting devices with DSO function
<b>Test focus</b>	Simple test case that checks whether the DUT (DSO) is able to handle MHP connection (including correct data transfer).
<b>Value of Interest</b>	Data pattern, transmitted via MOSTHigh
<b>Reference to MOST Specification</b>	[2], MH_Gen_BasicFlow
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	<p>Test has to be performed with block acknowledge mode only (AckMode=BlockAcknowledge).</p> <p>Test has to be performed with the following specified "NextPacketMethod":</p> <ul style="list-style-type: none"> <li>• Hold (DSONextPacket = e_Normal_HOLD_phase_will_close_after_timeout)</li> <li>• Terminate (DSONextPacket = e_Terminate_the_connection_and_open_a_new_one_if_necessary)</li> <li>• SendNext (if supported by DUT) (DSONextPacket = e_Try_to_send_next_packet_without_a_hold_optional)</li> </ul>
<b>Results</b>	<p><b>DUT ok (1):</b> The DUT has passed the test.</p> <p><b>DUT not ok (1):</b> Received data via MOSTHigh does not match with expected data pattern.</p> <p><b>DUT not ok (2):</b> DUT fails to send data pattern (Block Acknowledge mode) via MHP within t_Deadlock_Long.</p> <p><b>DUT not ok (3):</b> DUT sends less than 20 data packets (Block Acknowledge mode) via MHP within t_Deadlock_Long.</p>

**function** f\_MH\_DSO\_Basic (in DSO\_NextPacketMethod\_Param DSONextPacket, in DSO\_AckMode\_Param DSOAckMode )

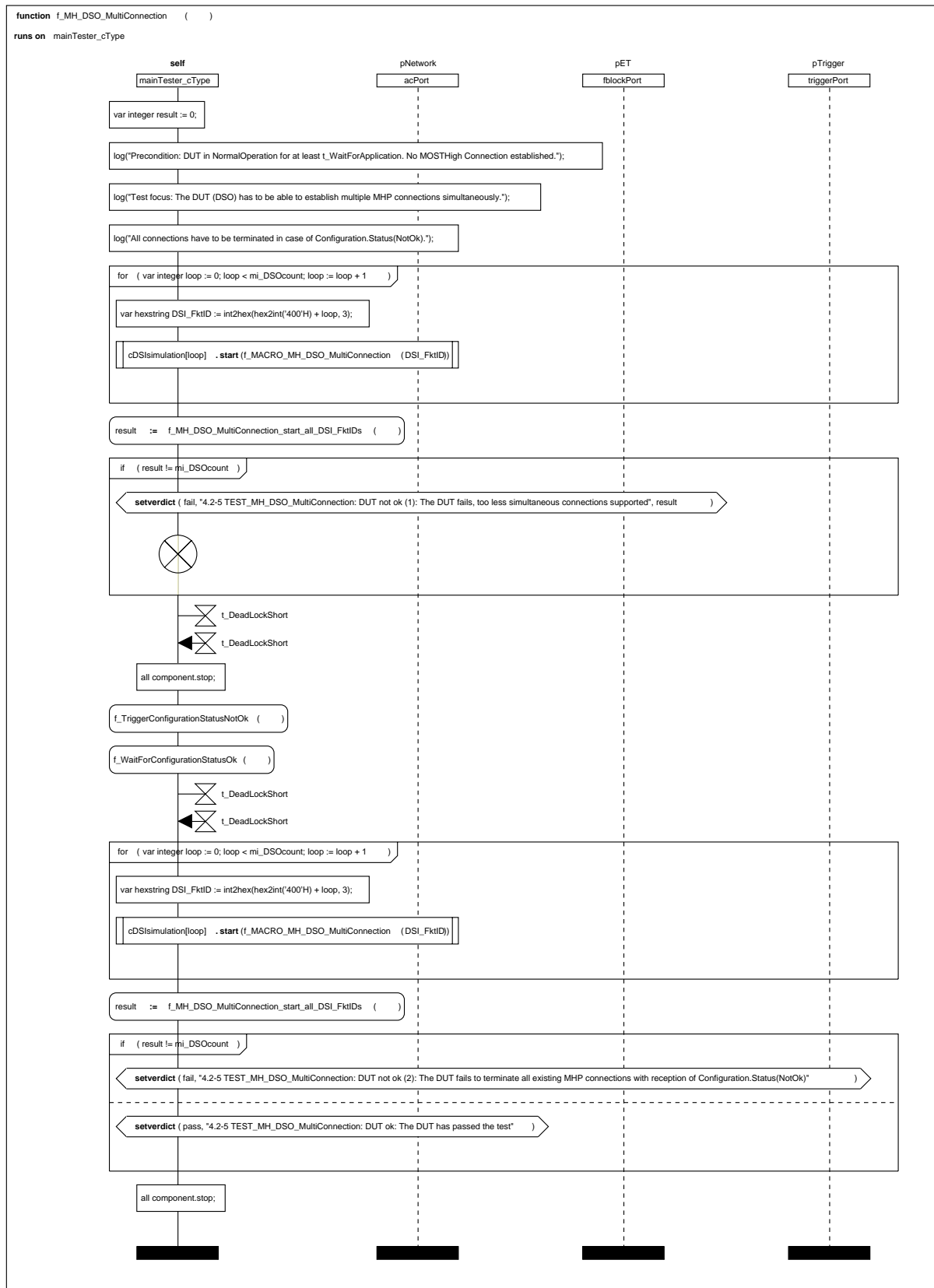
**runs on** tester\_cType

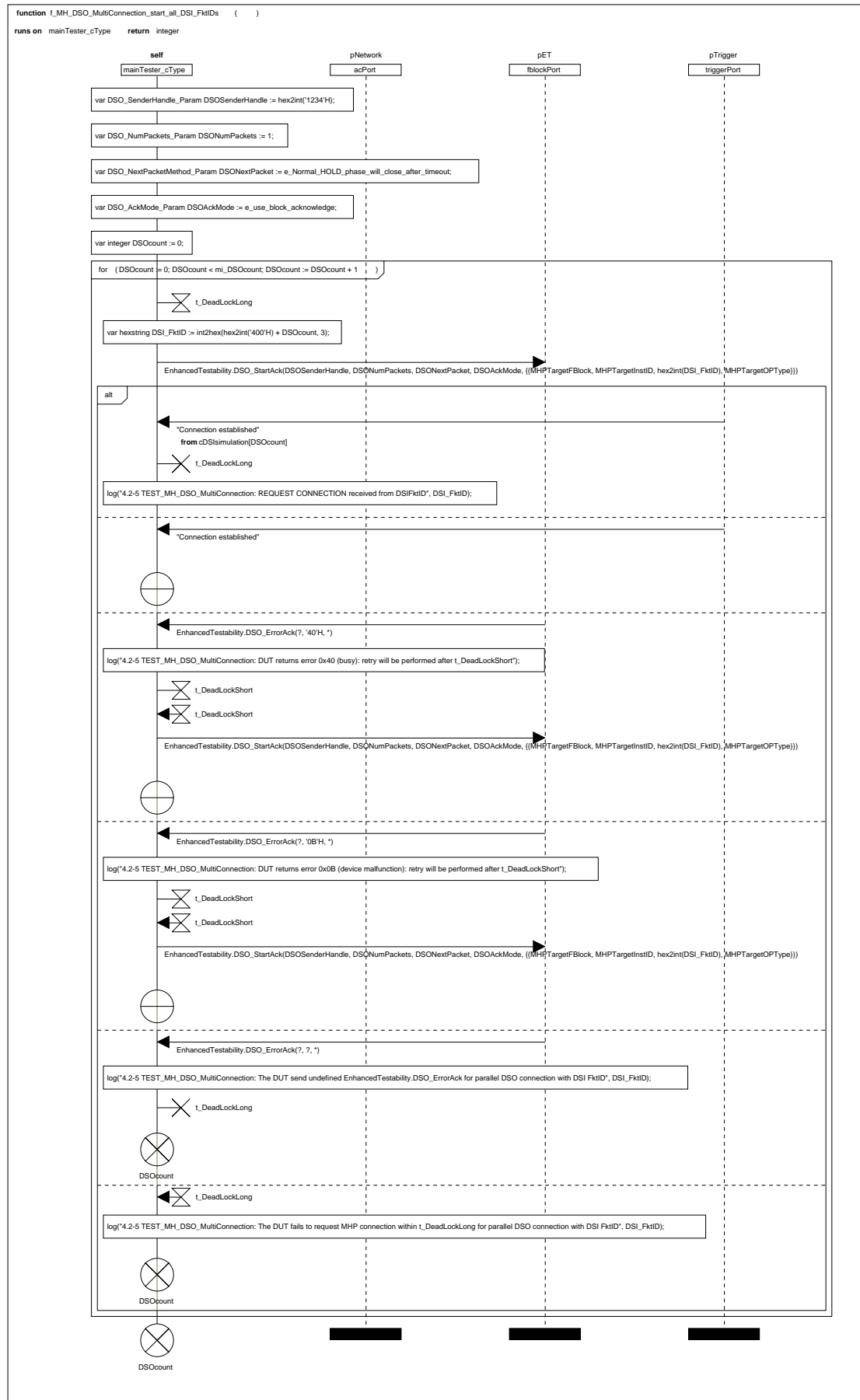




### 3.3.5 TEST\_MH\_DSO\_MultiConnection

<b>Name of test</b>	TEST_MH_DSO_MultiConnection 4.2-5
<b>Reference to GFT</b>	f_MH_DSO_MultiConnection ( ) f_MH_DSO_MultiConnection_start_all_DSI_FktIDs ( )
<b>Applicability</b>	All MHP supporting devices with DSO function
<b>Test focus</b>	The DUT (DSO) has to be able to establish multiple MHP connections simultaneously. All connections have to be terminated in case of Configuration.Status(NotOk).
<b>Value of Interest</b>	Number of parallel established connections. Termination of connections at Configuration.Status(NotOk).
<b>Reference to MOST Specification</b>	[2]; para 7.1.8 [2]; para 5.5
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	Test has to be performed with NextPacketMethod=Hold and AckMode=BlockAcknowledge.  The tester has to support up to 30 simultaneous MHP connections. In case mi_DSOCcount exceeds this number, mi_DSOCcount will be handled as if it was defined with the value 30.
<b>Results</b>	<b>DUT ok:</b> The DUT has passed the test. <b>DUT not ok (1):</b> The DUT fails, too less simultaneous connections supported. <b>DUT not ok (2):</b> The DUT fails to terminate all existing MHP connections with reception of Configuration.Status(NotOk).





## 3.4 Timing Tests For DSI

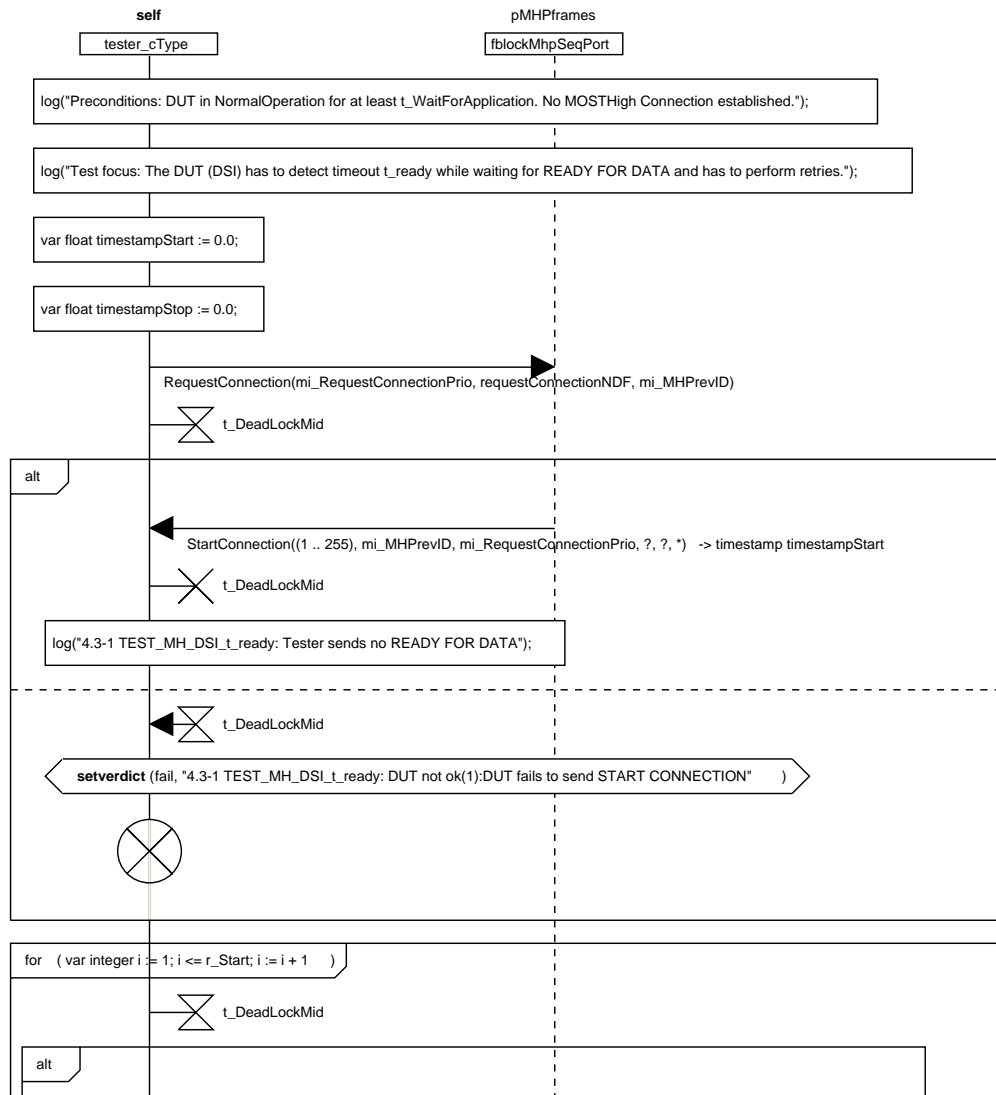
### 3.4.1 TEST\_MH\_DSI\_t\_ready

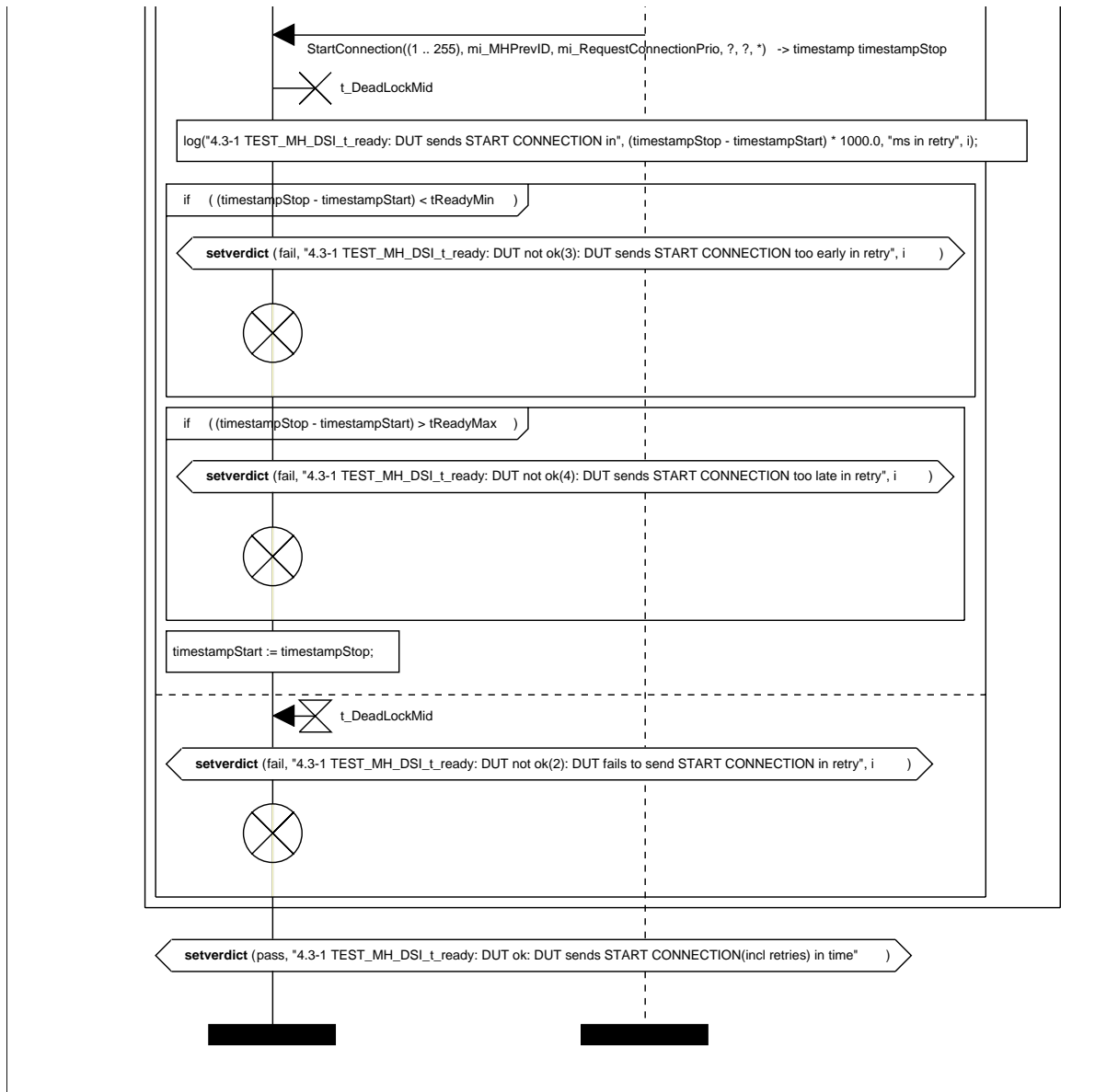
<b>Name of test</b>	TEST_MH_DSI_t_ready 4.3-1
<b>Reference to GFT</b>	f_MH_DSI_t_ready()
<b>Applicability</b>	All MHP supporting devices with DSI function
<b>Test focus</b>	The DUT (DSI) has to detect timeout t_ready while waiting for READY FOR DATA and has to perform retries.
<b>Value of Interest</b>	t_ready
<b>Reference to MOST Specification</b>	[2]; para 6.1, para 7.1.2, para 7.2.2
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	
<b>Results</b>	<b>DUT ok:</b> DUT sends START CONNECTION (incl. retries) in time. <b>DUT not ok (1):</b> DUT fails to send START CONNECTION. <b>DUT not ok (2):</b> DUT fails to send START CONNECTION in retry. <b>DUT not ok (3):</b> DUT sends START CONNECTION too early in retry. <b>DUT not ok (4):</b> DUT sends START CONNECTION too late in retry.



**function** f\_MH\_DSI\_t\_ready ( )

**runs on** tester\_cType



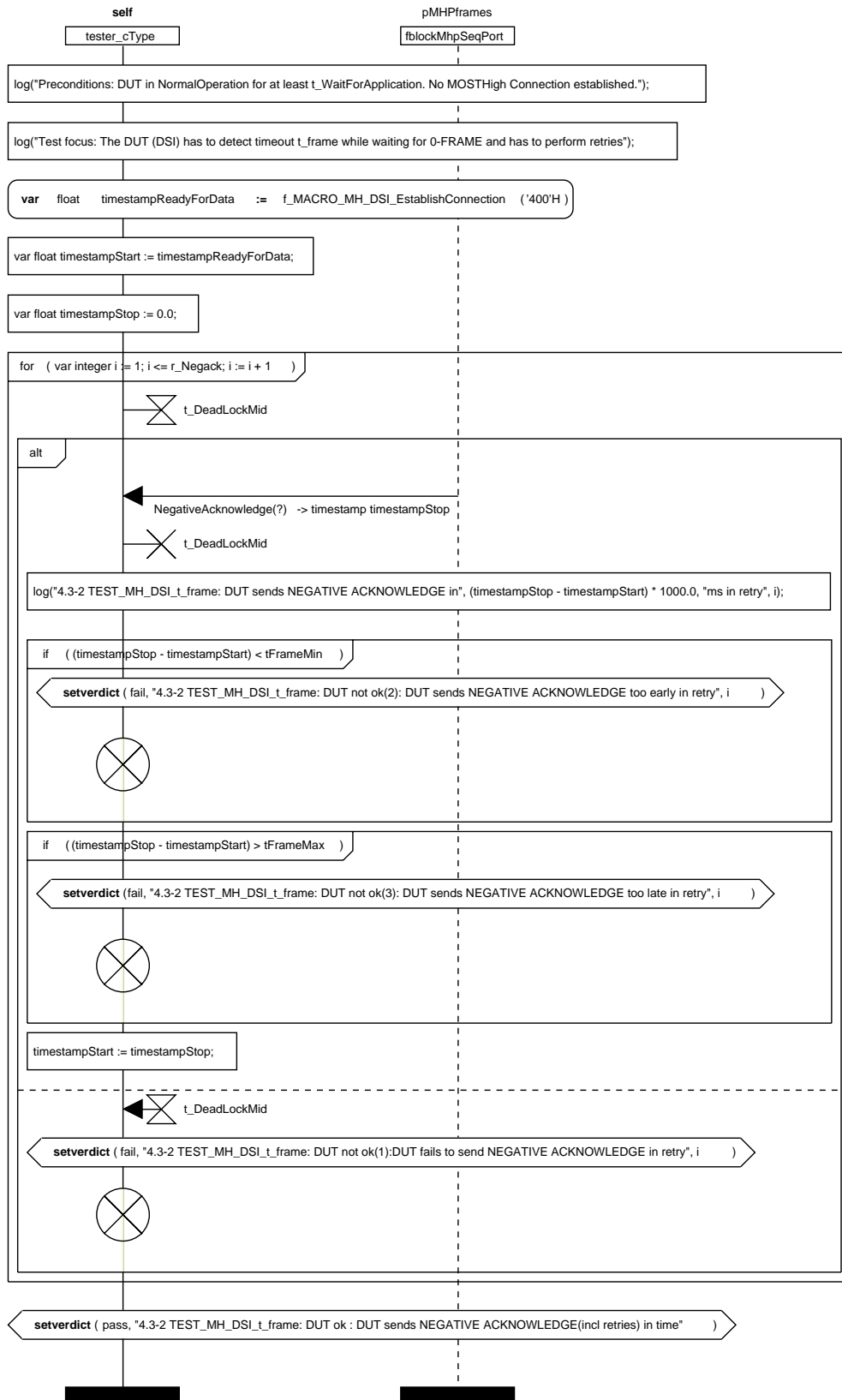


### 3.4.2 TEST\_MH\_DSI\_t\_frame

Name of test	TEST_MH_DSI_t_frame 4.3-2
Reference to GFT	f_MH_DSI_t_frame()
Applicability	All MHP supporting devices with DSI function
Test focus	The DUT (DSI) has to detect timeout t_frame while waiting for 0-FRAME and has to perform retries.
Value of Interest	t_frame
Reference to MOST Specification	[2]; para 6.1, para 7.2.9, para 7.2.10
Experimental set-up	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
Preconditions	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
Note	
Results	<b>DUT ok:</b> DUT sends NEGATIVE ACKNOWLEDGE (incl. retries) in time. <b>DUT not ok (1):</b> DUT fails to send NEGATIVE ACKNOWLEDGE in retry. <b>DUT not ok (2):</b> DUT sends NEGATIVE ACKNOWLEDGE too early in retry. <b>DUT not ok (3):</b> DUT sends NEGATIVE ACKNOWLEDGE too late in retry.

```
function f_MH_DSI_t_frame ( )
```

```
runs on tester_cType
```

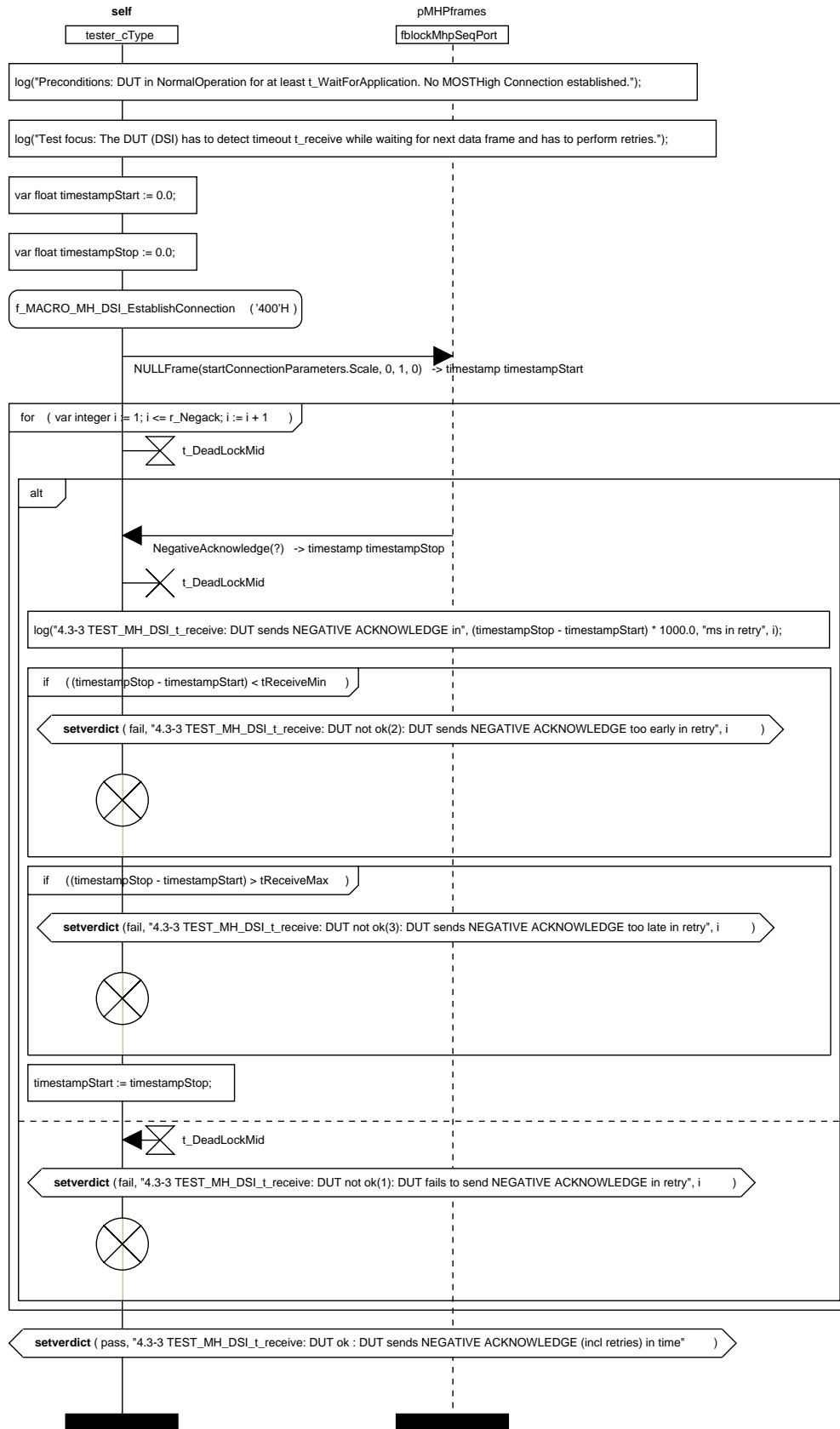


### 3.4.3 TEST\_MH\_DSI\_t\_receive

<b>Name of test</b>	TEST_MH_DSI_t_receive 4.3-3
<b>Reference to GFT</b>	f_MH_DSI_t_receive()
<b>Applicability</b>	All MHP supporting devices with DSI function
<b>Test focus</b>	The DUT (DSI) has to detect timeout t_receive while waiting for next data frame and has to perform retries.
<b>Value of Interest</b>	t_receive
<b>Reference to MOST Specification</b>	[2]; para 6.1, para 7.2.9, para 7.2.10
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	
<b>Results</b>	<b>DUT ok:</b> DUT sends NEGATIVE ACKNOWLEDGE (incl. retries) in time. <b>DUT not ok (1):</b> DUT fails to send NEGATIVE ACKNOWLEDGE in retry. <b>DUT not ok (2):</b> DUT sends NEGATIVE ACKNOWLEDGE too early in retry. <b>DUT not ok (3):</b> DUT sends NEGATIVE ACKNOWLEDGE too late in retry.

function f\_MH\_DSI\_t\_receive ( )

runs on tester\_cType

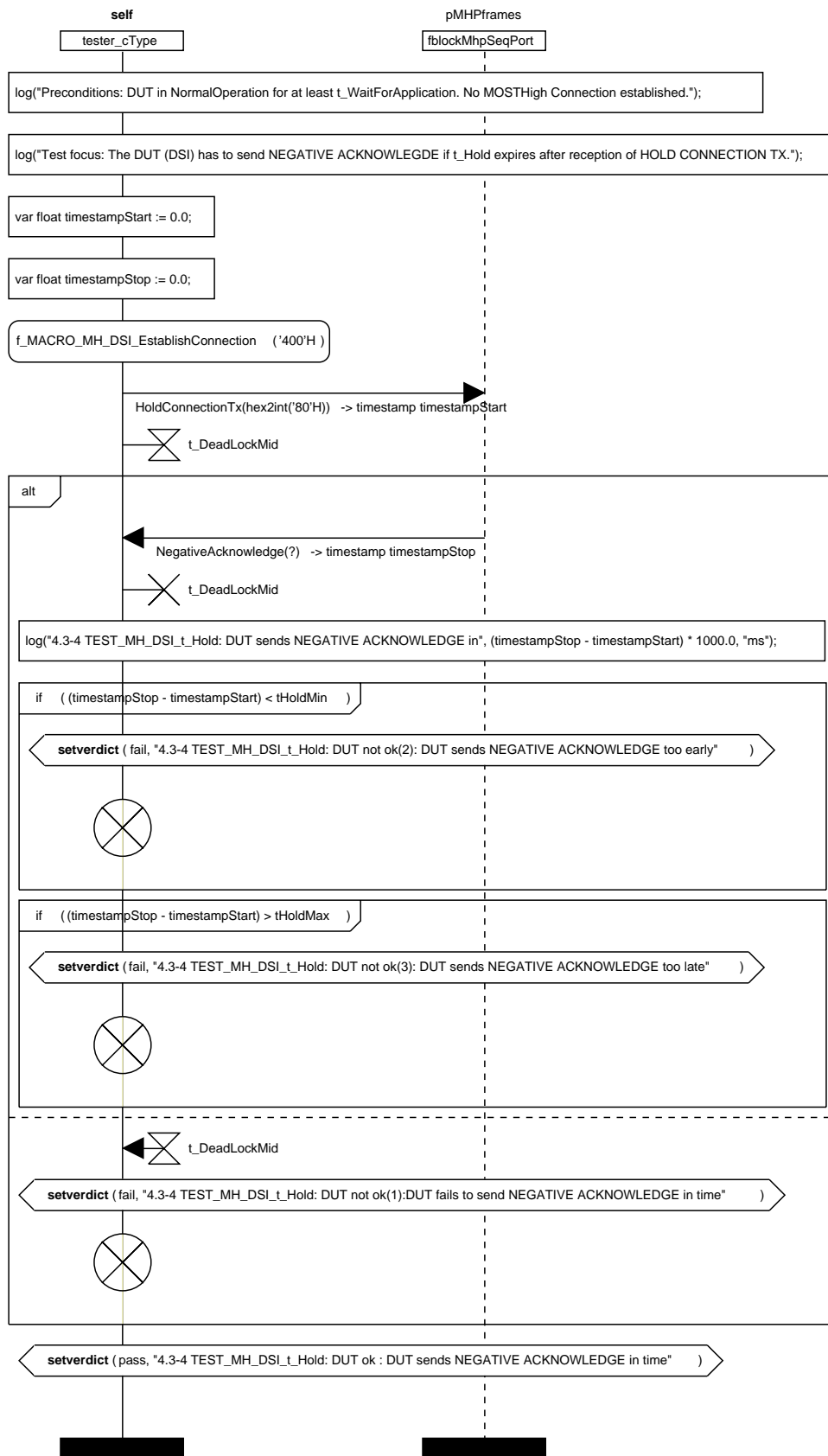


### 3.4.4 TEST\_MH\_DSI\_t\_Hold

Name of test	TEST_MH_DSI_t_Hold 4.3-4
Reference to GFT	f_MH_DSI_t_Hold()
Applicability	All MHP supporting devices with DSI function
Test focus	The DUT (DSI) has to send NEGATIVE ACKNOWLEDGE if t_Hold expires after reception of HOLD CONNECTION TX.
Value of Interest	t_Hold
Reference to MOST Specification	[2]; para 6.1, para 7.1.7
Experimental set-up	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
Preconditions	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
Note	
Results	<b>DUT ok:</b> DUT sends NEGATIVE ACKNOWLEDGE in time. <b>DUT not ok (1):</b> DUT fails to send NEGATIVE ACKNOWLEDGE in time. <b>DUT not ok (2):</b> DUT sends NEGATIVE ACKNOWLEDGE too early. <b>DUT not ok (3):</b> DUT sends NEGATIVE ACKNOWLEDGE too late.

function f\_MH\_DSI\_t\_Hold ( )

runs on tester\_cType



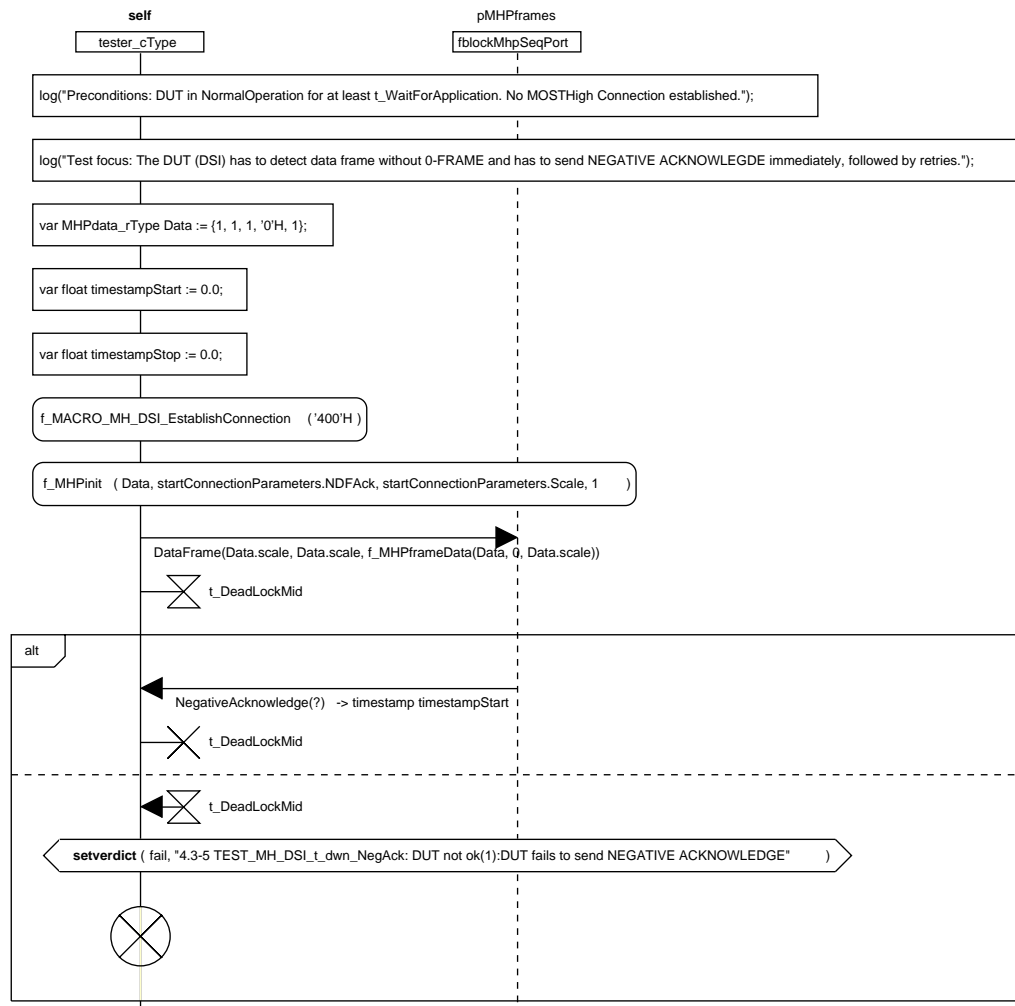


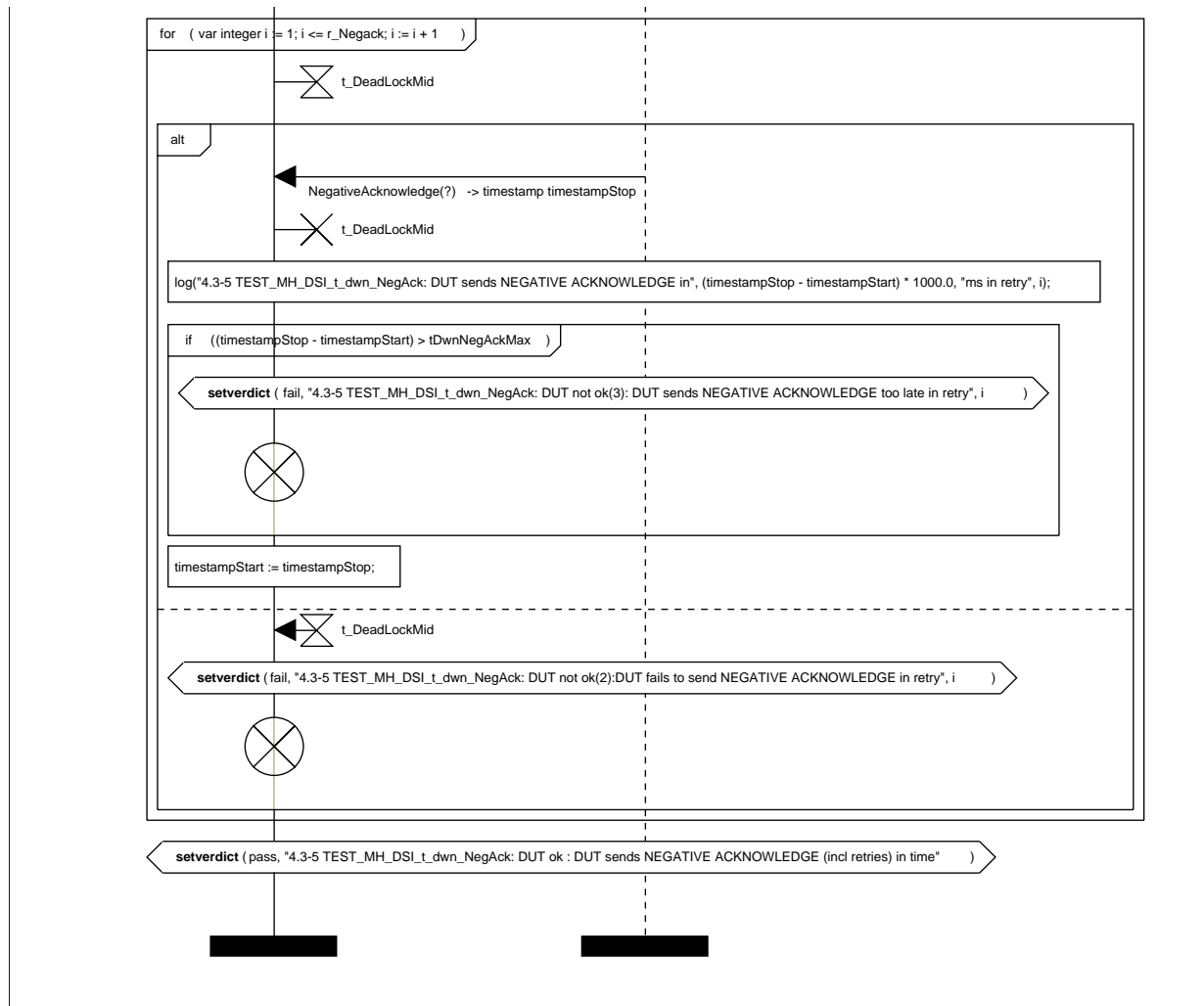
### 3.4.5 TEST\_MH\_DSI\_t\_dwn\_NegAck

<b>Name of test</b>	TEST_MH_DSI_t_dwn_NegAck 4.3-5
<b>Reference to GFT</b>	f_MH_DSI_t_dwn_NegAck()
<b>Applicability</b>	All MHP supporting devices with DSI function
<b>Test focus</b>	The DUT (DSI) has to detect data frame without 0-FRAME and has to send NEGATIVE ACKNOWLEDGE immediately, followed by retires
<b>Value of Interest</b>	t_dwn_NegAck
<b>Reference to MOST Specification</b>	[2]; para 6.1, para 7.2.3
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	
<b>Results</b>	<b>DUT ok:</b> DUT sends NEGATIVE ACKNOWLEDGE (incl. retries) in time. <b>DUT not ok (1):</b> DUT fails to send NEGATIVE ACKNOWLEDGE. <b>DUT not ok (2):</b> DUT fails to send NEGATIVE ACKNOWLEDGE in retry. <b>DUT not ok (3):</b> DUT sends NEGATIVE ACKNOWLEDGE too late in retry.

```
function f_MH_DSI_t_dwn_NegAck ( )
```

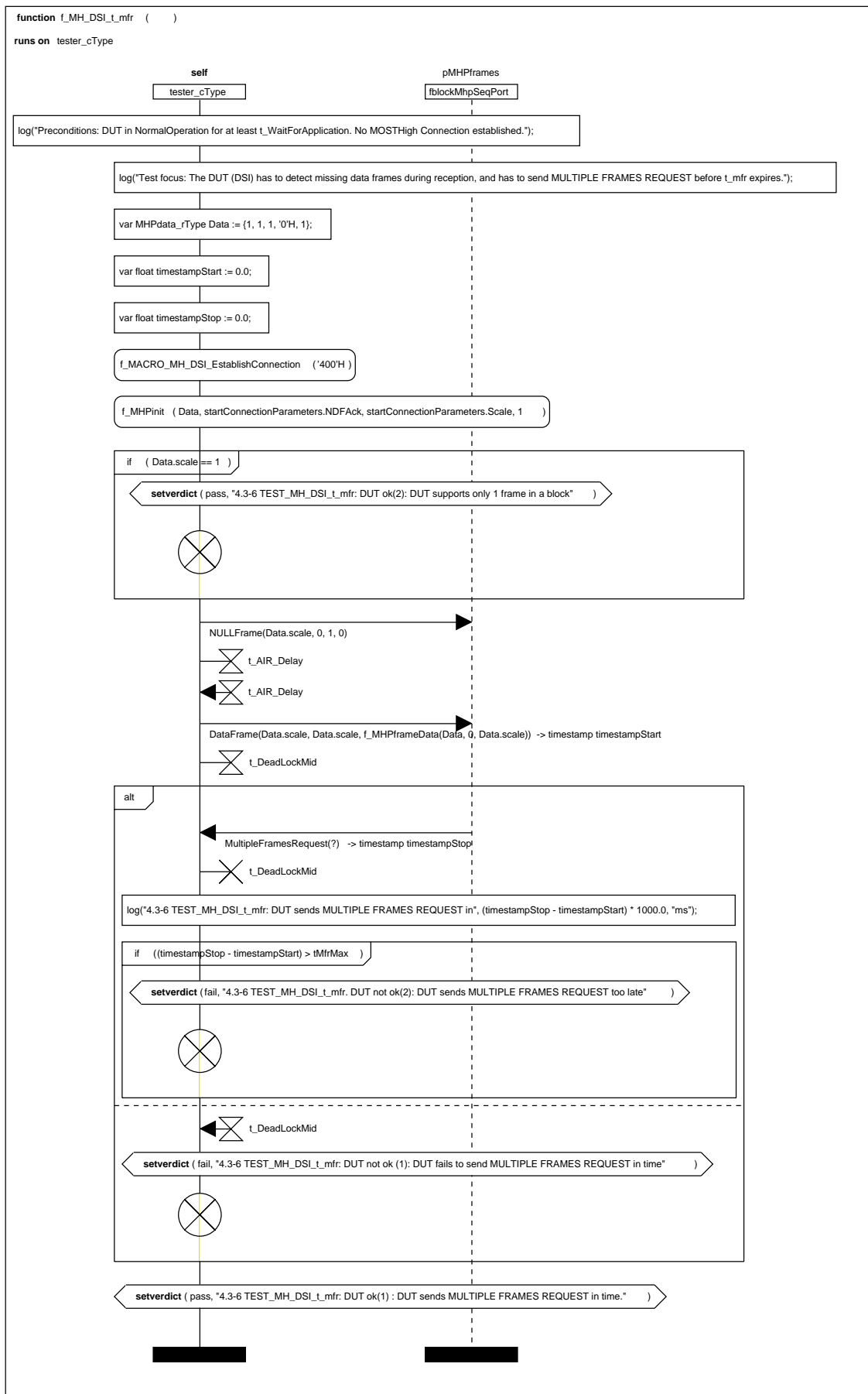
```
runs on tester_cType
```





### 3.4.6 TEST\_MH\_DSI\_t\_mfr

<b>Name of test</b>	TEST_MH_DSI_t_mfr 4.3-6
<b>Reference to GFT</b>	f_MH_DSI_t_mfr()
<b>Applicability</b>	All MHP supporting devices with DSI function
<b>Test focus</b>	The DUT (DSI) has to detect missing data frames during reception, and has to send MULTIPLE FRAMES REQUEST before t_mfr expires.
<b>Value of Interest</b>	t_mfr
<b>Reference to MOST Specification</b>	[2]; para 6.1, para 7.2.5
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	
<b>Results</b>	<b>DUT ok (1):</b> DUT sends MULTIPLE FRAMES REQUEST in time. <b>DUT ok (2):</b> DUT supports only 1 frame in a block. <b>DUT not ok (1):</b> DUT fails to send MULTIPLE FRAMES REQUEST in time. <b>DUT not ok (2):</b> DUT sends MULTIPLE FRAMES REQUEST too late.



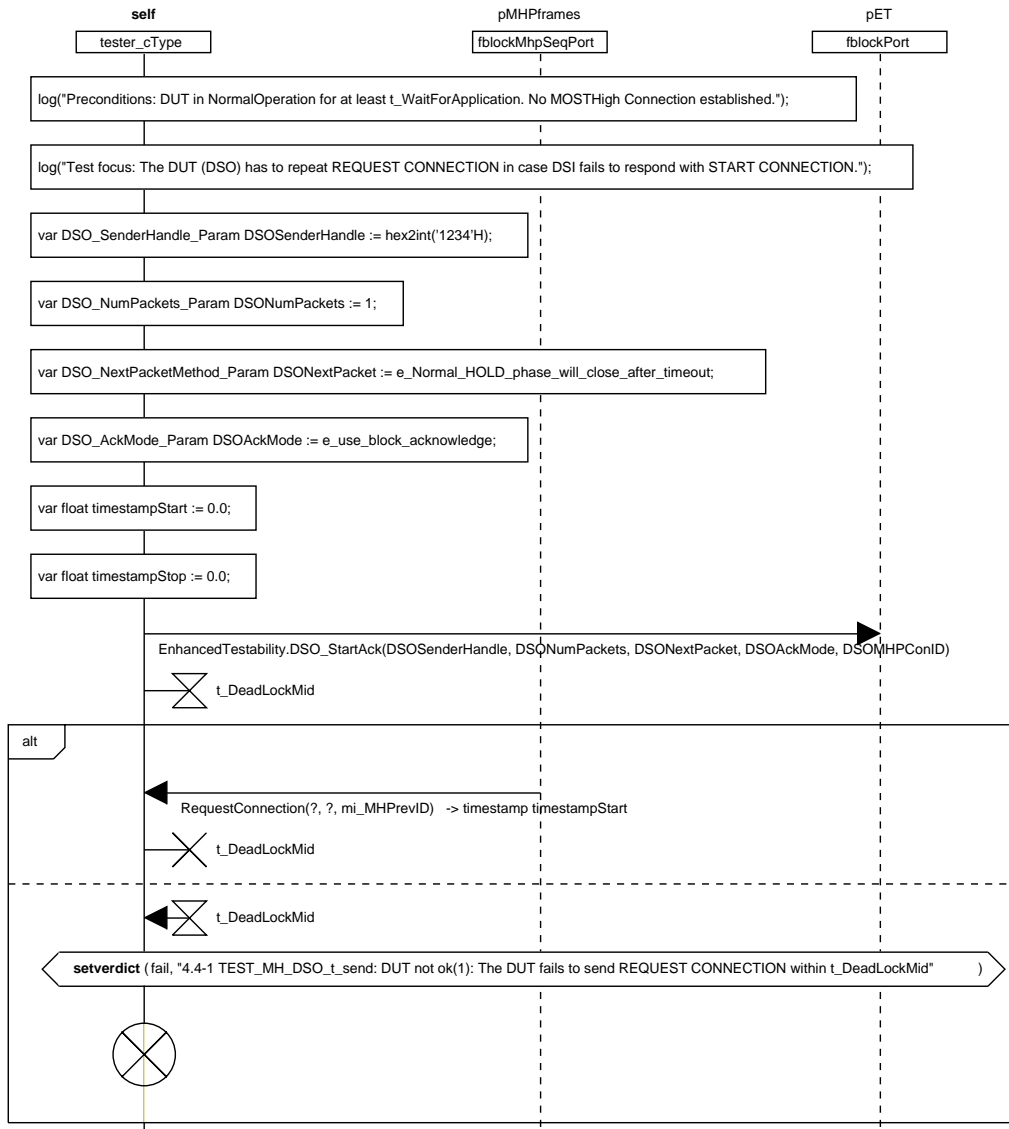
## 3.5 Timing Tests For DSO

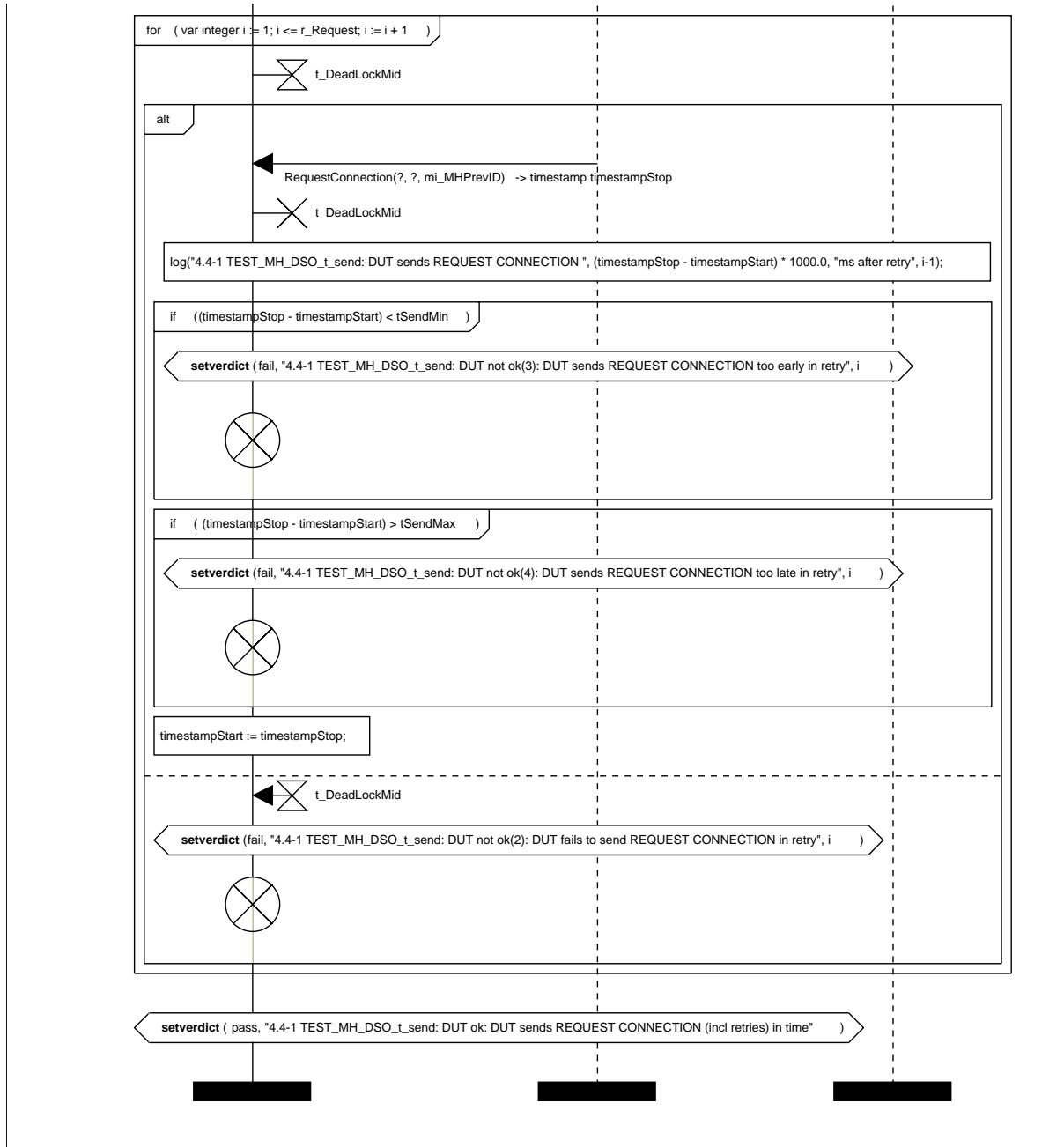
### 3.5.1 TEST\_MH\_DSO\_t\_send

<b>Name of test</b>	TEST_MH_DSO_t_send 4.4-1
<b>Reference to GFT</b>	f_MH_DSO_t_send()
<b>Applicability</b>	All MHP supporting devices with DSO function
<b>Test focus</b>	The DUT (DSO) has to repeat REQUEST CONNECTION in case the tester (DSI) fails to respond with START CONNECTION.
<b>Value of Interest</b>	t_send
<b>Reference to MOST Specification</b>	[2]; para 6.1, para 7.1.2, para 7.2.1
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	
<b>Results</b>	<b>DUT ok:</b> DUT sends REQUEST CONNECTION (incl. retries) in time. <b>DUT not ok (1):</b> The DUT fails to send REQUEST CONNECTION within t_DeadLockMid. <b>DUT not ok (2):</b> DUT fails to send REQUEST CONNECTION in retry. <b>DUT not ok (3):</b> DUT sends REQUEST CONNECTION too early in retry. <b>DUT not ok (4):</b> DUT sends REQUEST CONNECTION too late in retry.

**function** f\_MH\_DSO\_t\_send ( )

**runs on** tester\_cType





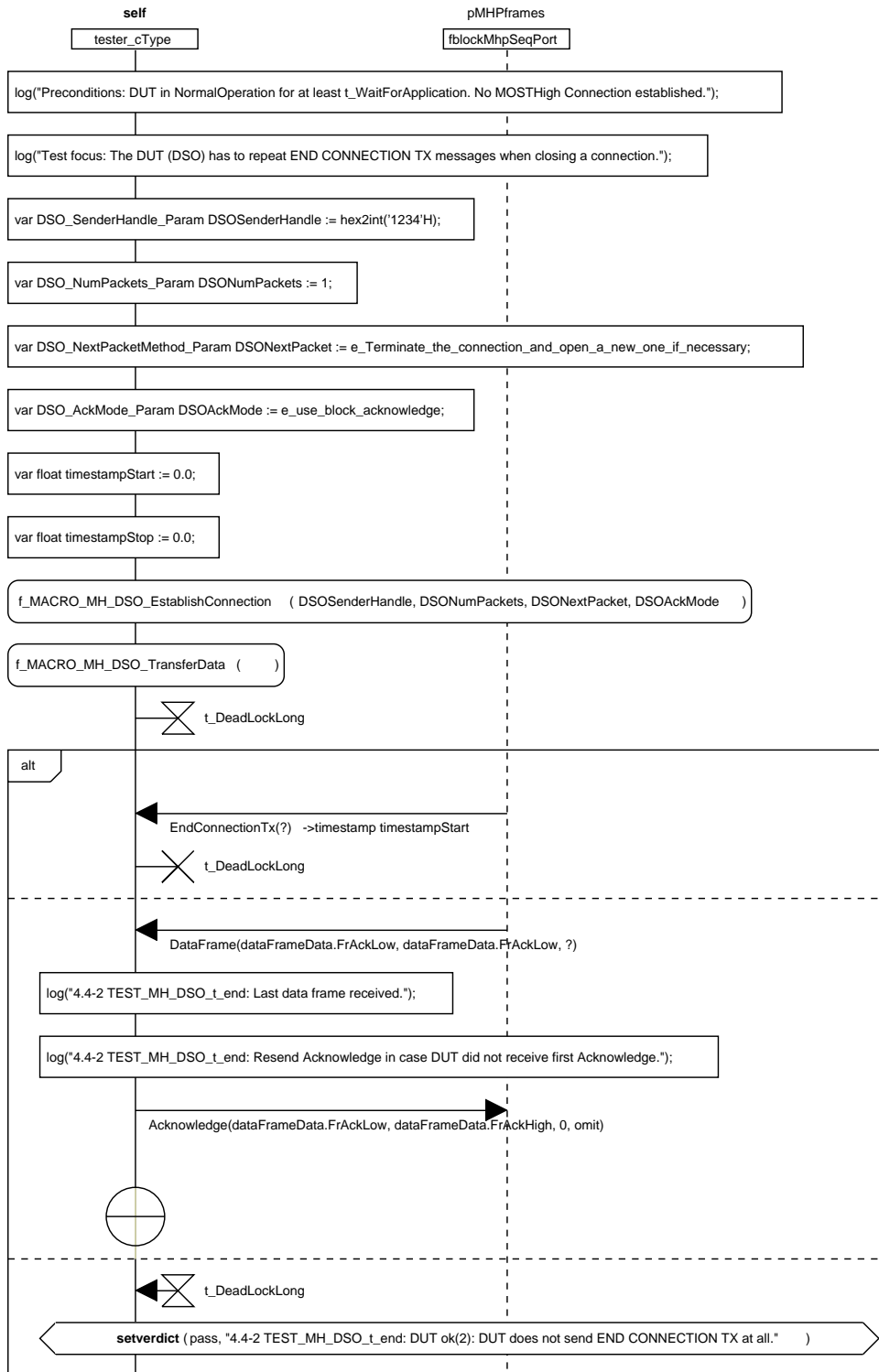


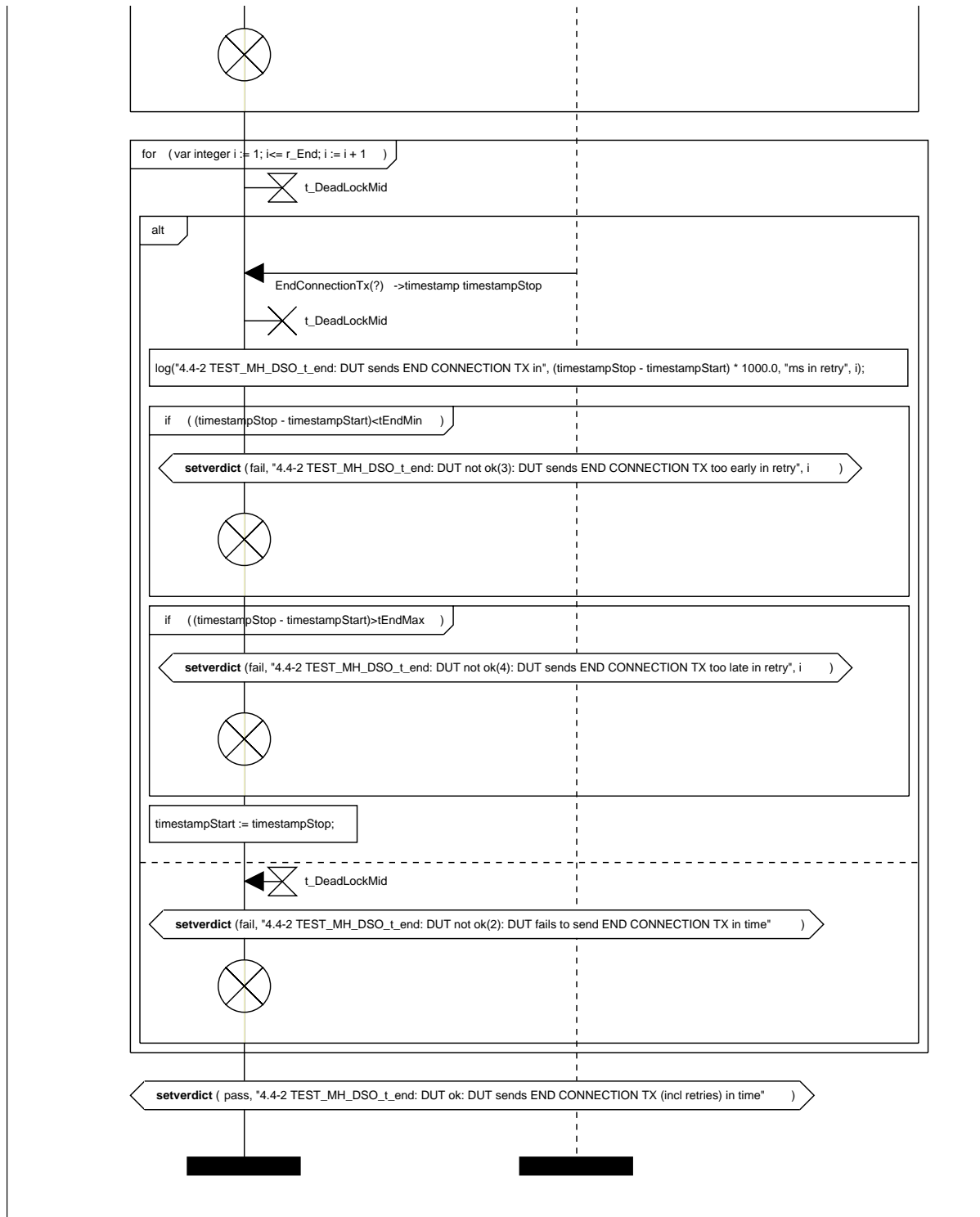
### 3.5.2 TEST\_MH\_DSO\_t\_end

<b>Name of test</b>	TEST_MH_DSO_t_end 4.4-2
<b>Reference to GFT</b>	f_MH_DSO_t_end()
<b>Applicability</b>	All MHP supporting devices with DSO function
<b>Test focus</b>	The DUT (DSO) has to repeat END CONNECTION TX messages when closing a connection. Alternatively, the DUT is allowed to close the connection without sending END CONNECTION TX at all.
<b>Value of Interest</b>	t_end
<b>Reference to MOST Specification</b>	[2]; para 6.1, para 7.1.5
<b>Experimental set-up</b>	<ul style="list-style-type: none"> <li>- Tester 1 in master mode or in slave mode (depends on DUT)</li> <li>- Tester 2 in spy mode</li> </ul>
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	
<b>Results</b>	<p><b>DUT ok (1):</b> DUT sends END CONNECTION TX (incl. retries) in time.</p> <p><b>DUT ok (2):</b> DUT does not send END CONNECTION TX at all.</p> <p><b>DUT not ok (1):</b> DUT fails to send END CONNECTION TX in t_DeadLockLong.</p> <p><b>DUT not ok (2):</b> DUT fails to send END CONNECTION TX in time.</p> <p><b>DUT not ok (3):</b> DUT sends END CONNECTION TX too early in retry.</p> <p><b>DUT not ok (4):</b> DUT sends END CONNECTION TX too late in retry.</p>

```
function f_MH_DSO_t_end ( )
```

```
runs on tester_cType
```



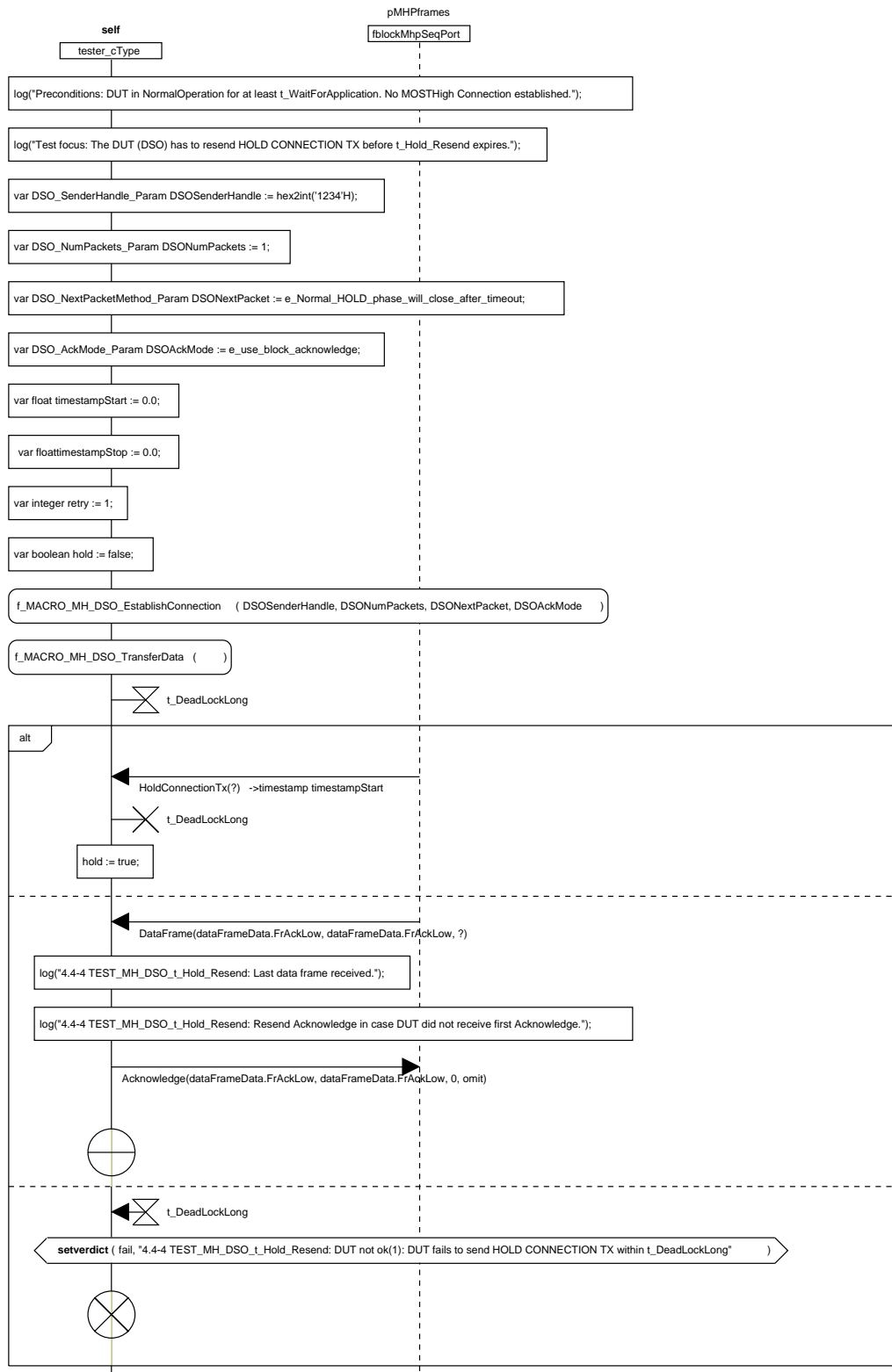


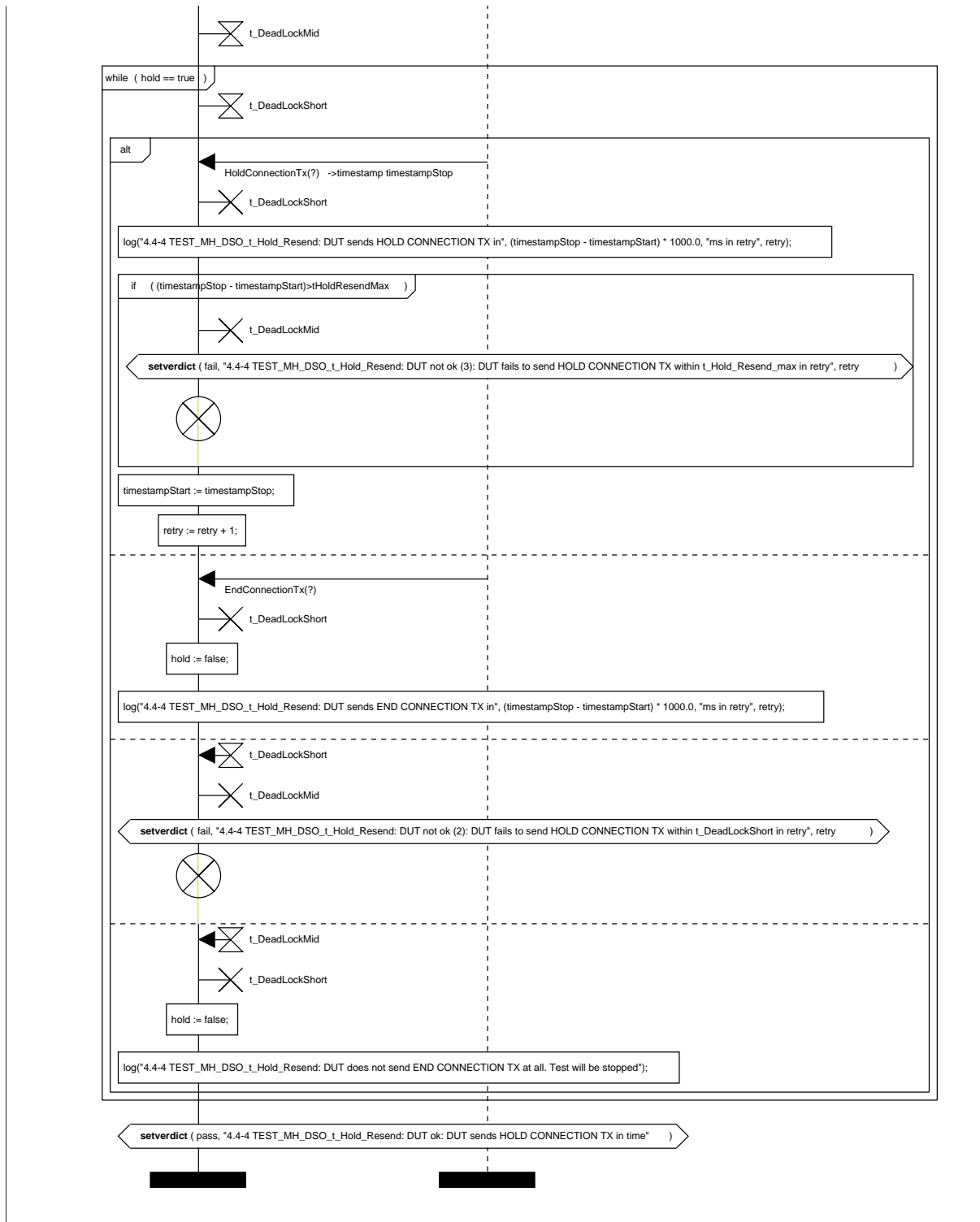
### 3.5.3 TEST\_MH\_DSO\_t\_Hold\_Resend

<b>Name of test</b>	TEST_MH_DSO_t_Hold_Resend 4.4-4
<b>Reference to GFT</b>	f_MH_DSO_t_Hold_Resend()
<b>Applicability</b>	All MHP supporting devices with DSO function
<b>Test focus</b>	The DUT (DSO) has to resend HOLD CONNECTION TX before t_Hold_Resend expires.
<b>Value of Interest</b>	t_Hold_Resend
<b>Reference to MOST Specification</b>	[2]; para 6.1, para 7.1.6, para 7.1.8
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	
<b>Results</b>	<b>DUT ok:</b> DUT sends HOLD CONNECTION TX in time. <b>DUT not ok (1):</b> DUT fails to send HOLD CONNECTION TX within t_DeadLockLong. <b>DUT not ok (2):</b> DUT fails to send HOLD CONNECTION TX within t_DeadLockShort in retry. <b>DUT not ok (3):</b> DUT fails to send HOLD CONNECTION TX within t_Hold_Resend_max in retry.

```
function f_MH_DSO_t_Hold_Resend ( )
```

```
runs on tester_cType
```



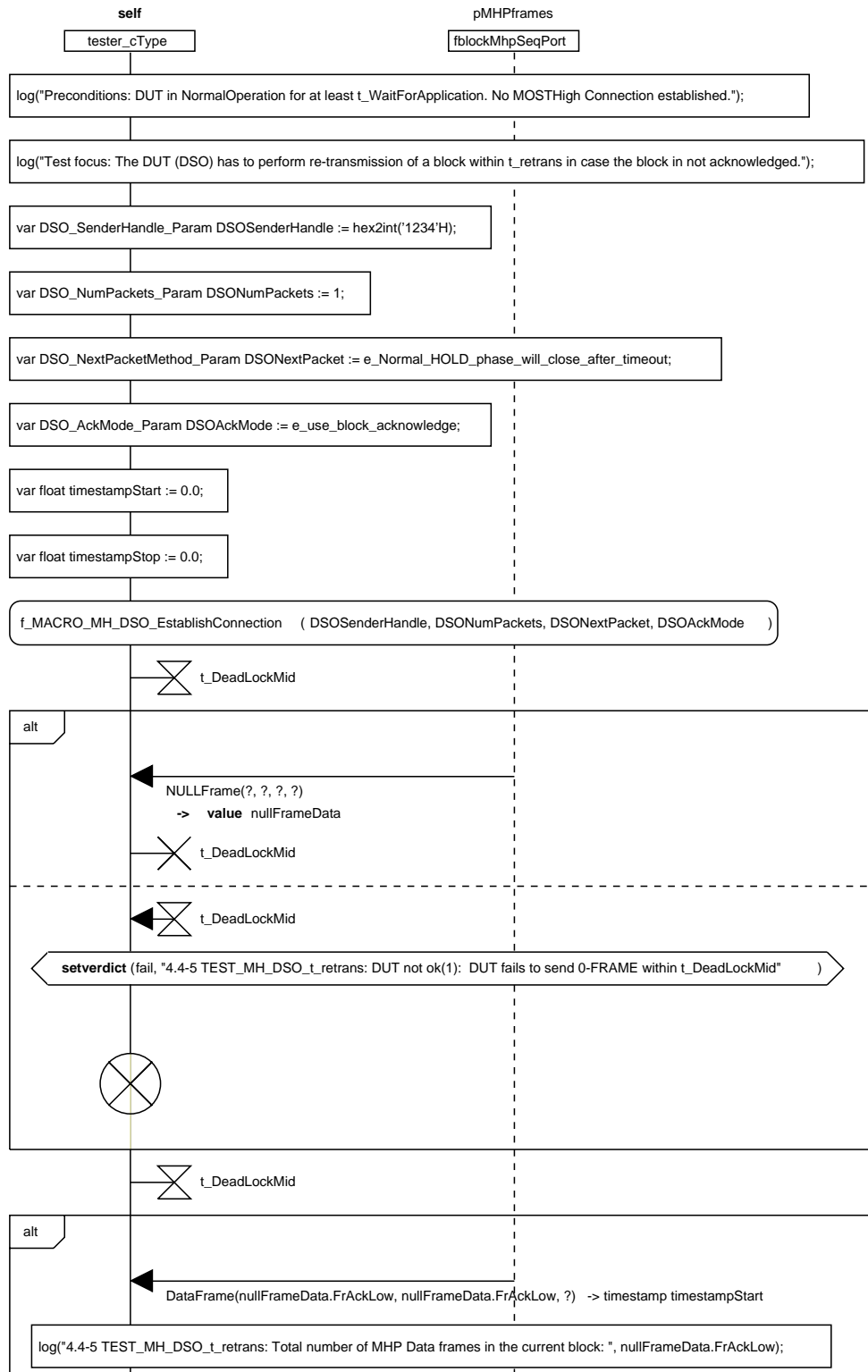


### 3.5.4 TEST\_MH\_DSO\_t\_retrans

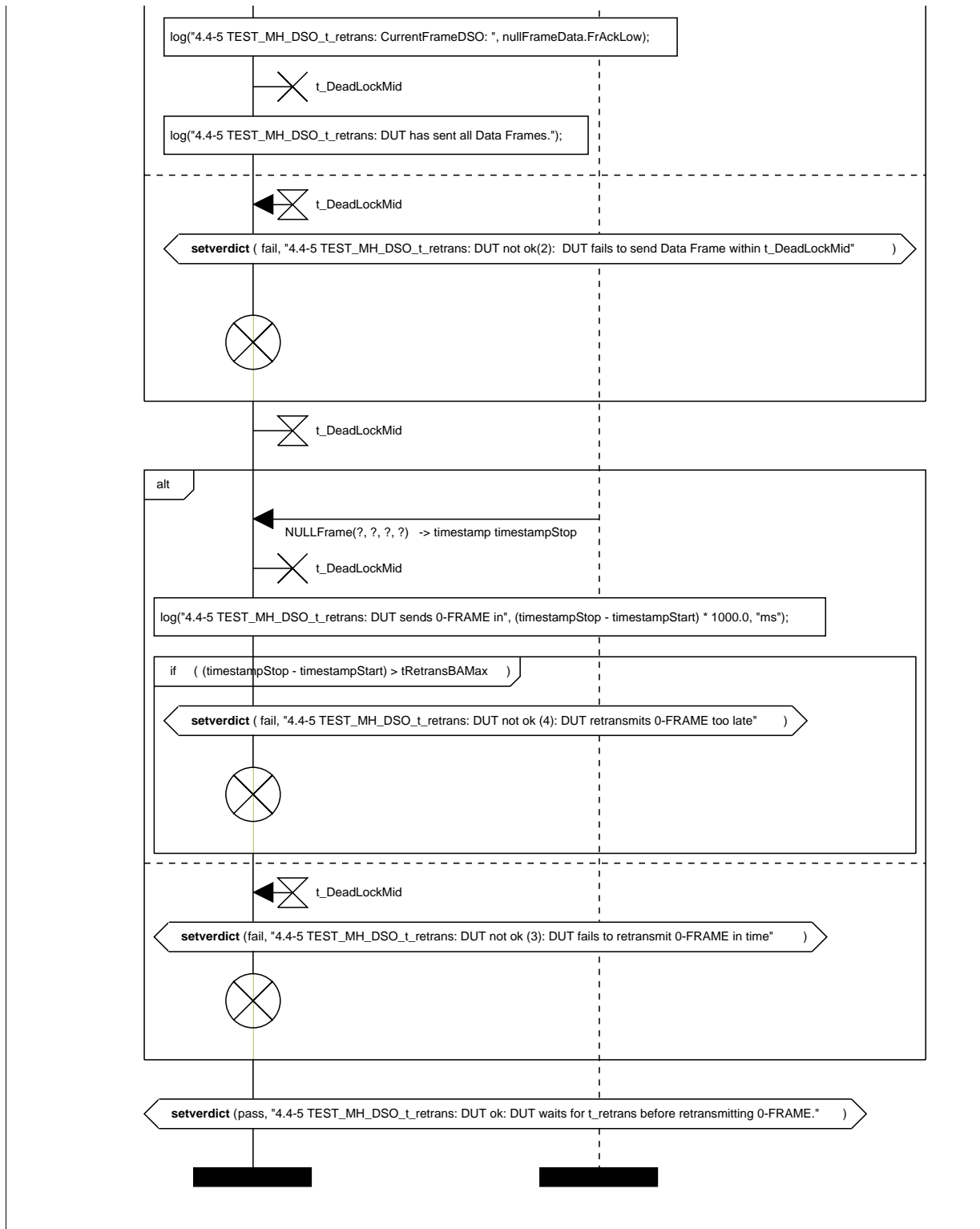
<b>Name of test</b>	TEST_MH_DSO_t_retrans 4.4-5
<b>Reference to GFT</b>	f_MH_DSO_t_retrans()
<b>Applicability</b>	All MHP supporting devices with DSO function
<b>Test focus</b>	The DUT (DSO) has to perform re-transmission of a block within t_retrans in case the block is not acknowledged.
<b>Value of Interest</b>	t_retrans
<b>Reference to MOST Specification</b>	[2]; para 6.1, para 7.2.8
<b>Experimental set-up</b>	- Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
<b>Preconditions</b>	DUT in NormalOperation for at least t_WaitForApplication. No MOSTHigh Connection established.
<b>Note</b>	
<b>Results</b>	<b>DUT ok:</b> DUT waits for t_retrans before retransmitting 0-FRAME. <b>DUT not ok (1):</b> DUT fails to send 0-FRAME within t_DeadLockMid. <b>DUT not ok (2):</b> DUT fails to send Data Frame within t_DeadLockMid. <b>DUT not ok (3):</b> DUT fails to retransmit 0-FRAME in time. <b>DUT not ok (4):</b> DUT retransmits 0-FRAME too late.

**function** f\_MH\_DSO\_t\_retrans ( )

**runs on** tester\_cType







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

Test Case Nr.	Designation	Measured Value	Measurement Uncertainty
4.0-1	TEST_MH_Identification	-	-
4.1-1	TEST_MH_DSI_RequestConnection_Ok	-	-
4.1-2	TEST_MH_DSI_Frame_Appropriate	-	-
4.1-3	TEST_MH_DSI_Frame_TooBig	-	-
4.1-4	TEST_MH_DSI_MultipleFramesRequest	-	-
4.1-5	TEST_MH_DSI_Basic	-	-
4.1-6	TEST_MH_DSI_MultiConnection	-	-
4.2-1	TEST_MH_DSO_RequestConnection	-	-
4.2-2	TEST_MH_DSO_RequestConnection_No Response	-	-
4.2-3	TEST_MH_DSO_RequestConnection_Ok	-	-
4.2-4	TEST_MH_DSO_Basic	-	-
4.2-5	TEST_MH_DSO_MultiConnection	-	-
4.3-1	TEST_MH_DSI_t_ready	t_ready	+/- 5ms
4.3-2	TEST_MH_DSI_t_frame	t_frame	+/- 5ms
4.3-3	TEST_MH_DSI_t_receive	t_receive	+/- 5ms
4.3-4	TEST_MH_DSI_t_Hold	t_Hold	+/- 5ms
4.3-5	TEST_MH_DSI_t_dwn_NegAck	t_dwn_NegAck	+/- 5ms
4.3-6	TEST_MH_DSI_t_mfr	t_mfr	+/- 5ms
4.4-1	TEST_MH_DSO_t_send	t_send	+/- 5ms
4.4-2	TEST_MH_DSO_t_end	t_end	+/- 5ms
4.4-4	TEST_MH_DSO_t_Hold_Resend	t_Hold_Resend	+/- 5ms
4.4-5	TEST_MH_DSO_t_retrans	t_retrans	+/- 5ms

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




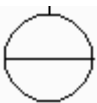


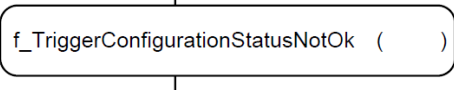
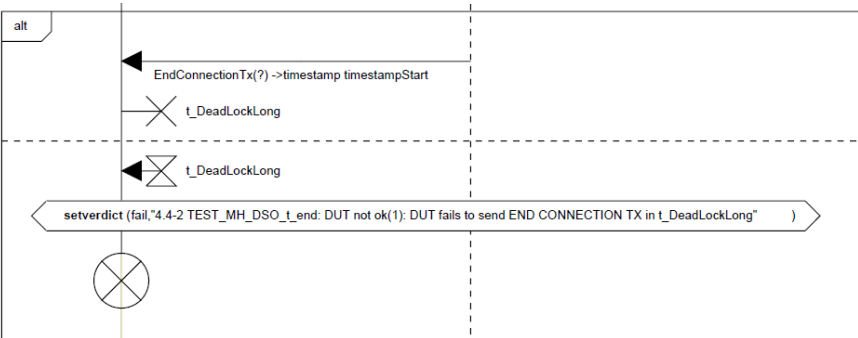
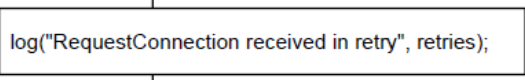
Note:

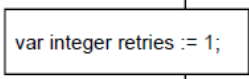
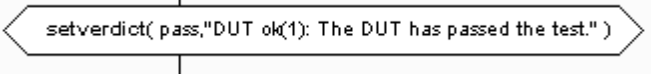
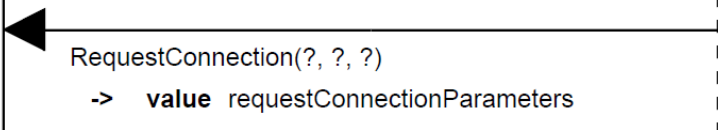
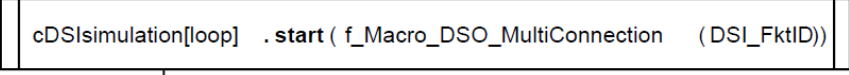
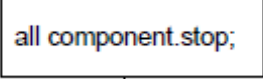
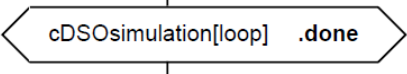
This table only contains FktIDs that will be used by at least one test case of Extended Core Compliance MHP.

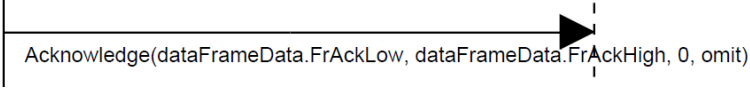
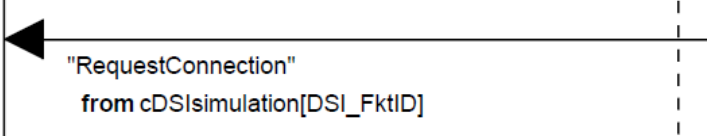
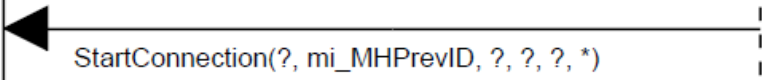

FktID	Name	Test Case / Macro	Page
0x3FD	DSIDSOCount	4.0-1 TEST_MH_Identification	32
0x3FE	DSO	MACRO_MH_DSO_RequestConnection	24
		4.0-1 TEST_MH_Identification	32
		4.2-1 TEST_MH_DSO_RequestConnection	51
		4.2-4 TEST_MH_DSO_Basic	58
		4.2-5 TEST_MH_DSO_MultiConnection	61
		4.4-1 TEST_MH_DSO_t_send	78
0x400 to 0x4FF	DSI	4.0-1 TEST_MH_Identification	32
		4.1-4 TEST_MH_DSI_MultipleFramesRequest	43
		4.1-5 TEST_MH_DSI_Basic	46

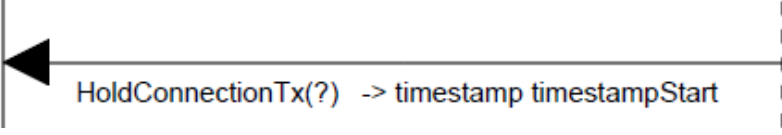
## Appendix 3: GFT Symbols (Informative)

Overview of GFT symbols, based on ETSI ES 201 873-3

Symbol	Description
 t_DeadLockShort	Start Timer
 t_DeadLockShort	Stop Timer
 t_DeadLockShort	Timeout Timer
	Repeat: A repeat statement is used in alt statements to restart the alt statement and wait for a new event in same alt statement.
	Stop: A stop statement is used to stop the current function and all associated functions within a test case. This stops the whole test case.
	Return: A return statement is used to leave the current function. It may be optionally associated with a return value.
 f_TriggerConfigurationStatusNotOk ( )	Reference: A reference symbol is used to call a macro or a function.
	Alternative statement: Each operand of the alternative statement is separated by a dashed line. Operands are evaluated top-down.
 log("RequestConnection received in retry", retries);	Logging: Writes some output into a log file.

Symbol	Description
	Variable declaration Optionally, declaration is done together with initialization, which is shown here in the symbol
	Condition: The test result is set by means of "setverdict()". It can be "pass" or "fail", followed by a description. It is possible to call setverdict() several times within a test case. Once it is set to "fail", it cannot be reset to "pass" by a subsequent setverdict() until test case is finished.
	Storage of parameter: The (optional) value-directive assigns received message to a variable.  Example: The parameter of "RequestConnection" is stored in "requestConnectionParameters" for later use.
	Start: Start execution of test component.
	Stop: Stop execution of test component.  Example: "All component.stop" will terminate all test components (e.g. macros) including their port instances.
	Done: Wait for termination of a PTC (Parallel Test Component)

Symbol	Description
 <p>A horizontal line with a solid black arrowhead pointing right. Below the line is the text: <code>Acknowledge(dataFrameData.FrAckLow, dataFrameData.FrAckHigh, 0, omit)</code>. A vertical dashed line is at the end of the arrow.</p>	<p>Parameter “omit”: The wildcard “omit” is used when an optional parameter is not used when sending a message.</p> <p>Example: Parameter HoldFlag of “ACKNOWLEDGE” is not used.</p>
 <p>A horizontal line with a solid black arrowhead pointing left. Below the line is the text: <code>"RequestConnection"</code> and <code>from cDSIsimulation[DSI_FktID]</code>. A vertical dashed line is at the end of the arrow.</p>	<p>Filter: The “from” statement is used to filter messages.</p> <p>Example: Only consider “Request Connections”, sent by a specific FktId. (filter of specific FktIDs)</p>
 <p>A horizontal line with a solid black arrowhead pointing left. Below the line is the text: <code>StartConnection(? , mi_MHPrevID, ? , ? , ? , *)</code>. A vertical dashed line is at the end of the arrow.</p>	<p>Optional Parameter “*”: Placeholder for optional parameter of a received message. Neither the existence nor the value of the parameter is evaluated when receiving the message.</p> <p>Example: The last parameter of “StartConnection” is optional.</p>
 <p>A horizontal line with a solid black arrowhead pointing left. Below the line is the text: <code>EndConnectionTx(?)</code>. A vertical dashed line is at the end of the arrow.</p>	<p>Parameter “?”: Placeholder for any value of a mandatory parameter of a received message. Only the existence but not the value of the parameter is evaluated when receiving the message. It could (optionally) be stored to a variable for later use (see “Storage of parameter”).</p>

Symbol	Description
	<p>Timestamp: Get the timestamp of received message.</p> <p>Example: The timestamp of reception of "HoldConnectionTx(?)" (when the message is detected on the MOST bus) is stored in variable "timestampStart".</p>

For further symbols and details, see ETSI ES 201 873-3 [3]

## Document History

Changes MOST Extended Core Compliance Test Specification: MOSTHigh V1.0.3 to MOST Extended Core Compliance Test Specification: MOSTHigh V1.0.3E1

Change Ref.	Section	Changes
V1.0.3E1_001	4.4-2	Test case updated to consider DUT that does not close a connection after timeout t_Delay_End
V1.0.3E1_002	4.4-4	Test case updated to consider DUT that does not close a connection after timeout t_Delay_End

Changes MOST Extended Core Compliance Test Specification: MOSTHigh V1.0.2 to MOST Extended Core Compliance Test Specification: MOSTHigh V1.0.3

Change Ref.	Section	Changes
V1.0.3_001	Document History	Moved to the end of the document
V1.0.3_002	Para 2.1.1	Remove para "Identification of MHP messages in MSCs" Minimum Average Interrupt Rate of Tester: AIR = 1000 Minimum requirements → requirements Support of 30 simultaneous MHP connections → Support of 30 simultaneous MHP connections at least - The tester is only allowed to deviate from MOST Specification and MHP Specification if indicated in the test case specification. - Timing measurement of the tester must ensure the measurement uncertainty given in Appendix 1. - The tester shall ensure that typical timing values are within ± 10% deviation.
V1.0.3_003	Para 2.2	Add: <b>Timing Values</b> If not otherwise stated, timing values in GFTs are specified in seconds.  Add description of functions: f_TriggerConfigurationStatusNotOk f_WaitForConfigurationStatusOk f_Get_t_AIR_Delay(integer DSI_AIR) f_analyse_DataPattern  Add pTrigger and pNetwork Add return value of functions Add function f_MHPcompareCRC  Update of Enum description Update of description of functions f_MHPinit and f_MHPframeData. Add explanation of NullFrame and NegativeAcknowledge regarding to number of parameter.



Change Ref.	Section	Changes
V1.0.3_004	Para 2.3	<p>Add note:</p> <p>The target FktID of MHP_ConID always starts with 0x400. In case of multiple connections, the value will be increased by one with each additional connection.</p> <p>The tester has to provide AIR value (testerAIR = 2000) with START CONNECTION.</p> <p>Value of NDF</p> <p>Config ok → System State OK</p> <p>Update:</p> <p>If it is proven that the DUT is not possible to reach precondition (such as modulated signal, lock, scan performed, System State OK), the test case will be marked as "DUT not ok".</p> <p>→</p> <p>If it is proven that the DUT is not able to reach precondition (such as modulated signal, lock, scan performed, System State OK), the test case will be marked as "DUT not ok (DUT not able to reach precondition)".</p> <p>Add definition of structure of ConID</p> <p>Remove definition "target FktID of MHP_ConID" from list of test procedure as now covered by definition of structure of ConID</p> <p>Remove "Preconditions are valid only for the described test case. The test flow may be reused in other test cases but with the preconditions, defined there." from list of test procedure</p>
V1.0.3_005	Para 2.4	This specification is applicable for all devices supporting MHP.
V1.0.3_006	Para 2.5	<p>SBC-Register only applicable for MOST 1V2 devices</p> <p>Remove "PrioAck"</p> <p>SBC-Register → Boundary descriptor / SBC-Register</p> <p>Add item "Support of NextPacketMethod = SendNext"</p>
V1.0.3_007	Para 2.7	<p>Change macro naming convention to "MACRO_MH_..."</p> <p>Add name of macros to test results</p> <p>MACRO_MH_MultipleFramesRequest → MACRO_MH_DSI_MultipleFramesRequest</p> <p>Macro_DSI_MultiConnection → MACRO_MH_DSI_EstablishConnection_Hold; Update note: "... returns 500ms delayed..." → "... returns delayed ..."</p> <p>MACRO_MH_DSI_EstablishConnection_Hold</p> <p>Add Node:</p> <p>Possibly, the macro terminates 500ms delayed after ConfigurationOk = False due to timeout t_Hold_Resend.</p> <p>New Macros:</p> <p>MACRO_MH_DSO_RequestConnection</p> <p>MACRO_MH_DSI_TransferData</p> <p>MACRO_MH_DSI_EstablishConnection_Hold:</p> <p>add note "Detection of ConfigurationOk depends on Configuration.Status of NetworkMaster."</p> <p>MACRO_MH_DSO_MultiConnection:</p> <p>Update of description.</p> <p>Add note "The macro itself only terminates in the error case. It must be terminated by the calling procedure."</p> <p>MACRO_MH_DSI_EstablishConnection:</p> <p>Remove note.</p>

Change Ref.	Section	Changes
V1.0.3_008	All test cases	Update of all GFTs
V1.0.3_009	4.0-1	Note how to treat test results with prefix "Warning". Update value of interest
V1.0.3_010	4.1-1	Remove t_send from test focus
V1.0.3_011	4.1-2	Test to be performed with block acknowledge mode  Update note: AckMode=0x01 → AckMode=BlockAcknowledge Add parameter DataFrameSizeIdentical
V1.0.3_012	4.1-3	DUT ok (3): This test is only applicable for MHP supporting devices with DSI function that are limited to packet size less than 1014 byte → This test is only applicable for MHP supporting devices with DSI function that are limited to packet size less than maximum possible value.
V1.0.3_013	4.1-5	Test performed with block acknowledge mode only.  Update note: AckMode=0x01 → AckMode=BlockAcknowledge
V1.0.3_014	4.1-6	Update Value of interest: Number of simultaneous connects → Number of parallel established connections Add "Behavior at Configuration.Status(NotOk) according to existing MHP connections." All connections have to be terminated in case of Configuration.Status(NotOk). → Termination of connections at Configuration.Status(NotOk).  Update Note: In case mi_DSICount exceeds this number, the value of mi_DSICount has to reduced to 30. --> In case mi_DSICount exceeds this number, mi_DSICount will be handled as if it was defined with the value 30.  Add note: The timer t_WaitBetweenConnections (= 100ms) specifies the pause, the tester has to wait between triggering new connections.
V1.0.3_015	4.2-1	Test has to be performed with block acknowledge mode only. Test has to be performed with the following specified "NextPacketMethod" Remove description of NextPacketMethod codes; replace by specific parameter NextPacketMethod = SendNext only if supported by DUT
V1.0.3_016	4.2-2	Update Note: Test has to be performed with "NextPacketMethod=0x00" (normal HOLD phase) and "AckMode=0x01" (block acknowledge). → Test has to be performed with NextPacketMethod=Hold and AckMode=BlockAcknowledge.  Update test focus: The DUT (DSO) has to perform r_request retries establishing a MOSTHigh connection. Then it has to stop further retries. → The DUT (DSO) has to perform at least r_request retries establishing a MOSTHigh connection.  Test results: Remove DUT not ok(2): DUT performs too many retries.

Change Ref.	Section	Changes
V1.0.3_017	4.2-3	<p>Update Note:  Test has to be performed with "NextPacketMethod=0x00" (normal HOLD phase) and "AckMode=0x01" (block acknowledge).  →  Test has to be performed with NextPacketMethod=Hold and AckMode=BlockAcknowledge.</p> <p>Test focus and Results:  delete "within t_Ready"</p>
V1.0.3_018	4.2-4	<p>Test has to be performed with block acknowledge mode only.  Test has to be performed with with the following specified "NextPacketMethod"  Remove description of NextPacketMethod codes; replace by specific parameter  NextPacketMethod = SendNext only if supported by DUT</p>
V1.0.3_019	4.2-5	<p>Value of interest:  Number of simultaneous connects → Number of parallel established connections  All connections have to be terminated in case of Configuration.Status(NotOk).  →  Termination of connections at Configuration.Status(NotOk).</p> <p>Update Note:  In case mi_DSOCcount exceeds this number, the value of mi_DSOCcount has to reduced to 30.  →  In case mi_DSOCcount exceeds this number, mi_DSOCcount will be handled as if it was defined with the value 30.</p> <p>Update Note:  Test has to be performed with "NextPacketMethod=0x00" (normal HOLD phase) and "AckMode=0x01" (block acknowledge).  →  Test has to be performed with NextPacketMethod=Hold and AckMode=BlockAcknowledge.</p>
V1.0.3_020	4.3-6	<p>Update test focus:  DSI has to detect missing data frames during reception, and to send MULTIPLE FRAMES REQUEST if t_mfr expires.  →  DSI has to detect missing data frames during reception, and to send MULTIPLE FRAMES REQUEST before t_mfr_max expires.</p>
V1.0.3_021	4.4-2	<p>DUT not ok (1): DUT fails to send END CONNECTION TX in t_DeadLockLong  →  DUT not ok (1): DUT fails to send END CONNECTION TX within t_DeadLockLong</p>
V1.0.3_022	Appendix 1	Remove measurement uncertainty of test cases 4.1-1, 4.2-2 and 4.2-3
V1.0.3_023	Appendix 2	<p>Remove FktID 0x3FF DSIHold from table.  Reason:  This table only contains FktIDs that will be used by at least one test case of Extended Core Compliance MHP.</p> <p>Macros considered, too</p>
V1.0.3_024	Appendix 3	Add appendix 3 (GFT Symbols)

## Changes MOST Extended Core Compliance Test Specification: MOSTHigh V1.0.1 to MOST Extended Core Compliance Test Specification: MOSTHigh V1.0.2

Change Ref.	Section	Changes
V1.0.2_001	para 2	<p>(1) "2.2 GFT" added for description of GFT and functions</p> <p>(2) Definition of "States of DUT"</p> <p>(3) Update of "Input by device manufacturer"</p> <p>(4) Delete "Supported MHP version" from list of input by device manufacturer as covered by mi_MHPrevID</p>
V1.0.2_002	para 2.1.2	<p>Packet length of 1014 byte --&gt; Maximum possible packet length of 1014 byte (MOST 2) resp. 1524 byte (MOST 3)</p>
V1.0.2_003	para 2.2	<p>Use experimental setup #1 of MOST Core Compliance Test Specification [1]. --&gt; Use experimental setup of MOST Core Compliance Test Specification [1].</p>
V1.0.2_004	para 2.3	<p>Device Types: Deleted " All tests are applicable for MOST25, MOST50 oPHY and MOST50 ePHY devices."  (based on errata)</p>
V1.0.2_005	para 2.3	<p>Update test procedure:</p> <p>(1) If it is not possible to reach precondition, the test case will be marked as "DUT not ok". --&gt; If it is proven that the DUT is not possible to reach precondition (such as modulated signal, lock,scan performed, Config ok), the test case will be marked as "DUT not ok".</p> <p>(2) Because of reproducibility reasons after each test case the DUT has to be disconnected from power. --&gt; Because of reproducibility reasons after each test case the DUT has to be disconnected from power. The DUT shall be shutdown regularly before disconnecting from power.</p> <p>(3) In case DUT = DSI: The tester has to consider t_AIR_Delay during testing. In case DUT = DSO: The tester has to provide AIR value with START CONNECTION it is capable to handle.</p> <p>(4) Wait until buffer capacitors are empty.</p> <p>(5) Unknown MHP connections that are not considered by the test case are ignored by the tester.</p>
V1.0.2_006	para 2.4	<p>This specification is independent from speed grade and physical interface. --&gt; This specification is applicable for MOST25 devices.</p>

Change Ref.	Section	Changes
V1.0.2_007	para 2.5	Add chapter for check of CRC32 implementation of FB ET (based on errata)
V1.0.2_008	para 2.7	Introduction of macros
V1.0.2_009	para 3.4	New chapter "Timing Tests for DSI"
V1.0.2_010	para 3.5	New chapter "Timing Tests for DSO"
V1.0.2_011	all test cases	Preconditions: "DUT in NormalOperation" --> "DUT in NormalOperation for at least t_WaitForApplication"
V1.0.2_012	all test cases	Add description of experimental set-up: - Tester 1 in master mode or in slave mode (depends on DUT) - Tester 2 in spy mode
V1.0.2_013	4.0-1	DUT ok (5) and DUT not ok(4) now cover MHP Rev. 2.3 and 2.3.1
V1.0.2_014	4.1-1	(1) Value of interest: StartConnection  (2) t_send --> t_send_max
V1.0.2_015	4.1-2	(1) Value of interest: Acknowledge  Delete note: "Deviation from MOST High Spec Rev. 2.3: Start transmission with CurrentFrameDSO =1 and continue transmission as long as (CurrentFrameDSO <= NumFramesInBlock)"  (2) "sender handle" --> "SenderHandle"  (3) t_retrans --> t_retrans_max  (4) FrameAck --> FRAME_ACKNOWLEDGE / NEGATIVE_ACKNOWLEDGE
V1.0.2_016	4.1-3	(1) Remove note: "Deviation from MOST High Spec Rev. 2.3: Start transmission with CurrentFrameDSO =1 and continue transmission as long as (CurrentFrameDSO <= NumFramesInBlock)"  (2) FrameAck --> FRAME_ACKNOWLEDGE / NEGATIVE_ACKNOWLEDGE  (3) t_retrans --> t_retrans_max; t_send --> t_send_max

Change Ref.	Section	Changes
V1.0.2_017	4.1-4	<p>(1)</p> <p>When a device answers with a scale ==1 (only 1 frame within a block) test should stop with a new kind of Result "DUT ok (2): DUT supports only 1 frame in a block". Therefore "DUT ok" will be renamed to "DUT ok (1)"</p> <p style="text-align: right;">(based on errata)</p> <p>(2)</p> <p>t_frame --&gt; t_frame_max --&gt; t_mfr_max</p> <p>(3)</p> <p>Update of Note:</p> <p>t_Delay_end --&gt; t_Hold_Resend_max</p>
V1.0.2_018	4.1-6	<p>Add note:</p> <p>The tester has to support up to 30 simultaneous MHP connections. In case mi_DSICount exceeds this number, the value of mi_DSICount has to reduced to 30.</p>
V1.0.2_019	4.2-1	Value of interest: RequestConnection
V1.0.2_020	4.2-2	<p>(1)</p> <p>Remove t_Send from value of interest</p> <p>(2)</p> <p>t_send --&gt; t_send_min; t_DeadLockShort --&gt; t_send_max</p>
V1.0.2_021	4.2-3	<p>(1)</p> <p>Value of interest: ReadyForData</p> <p>(2)</p> <p>t_ready --&gt; t_ready_max</p>
V1.0.2_022	4.2-4	<p>(1)</p> <p>Add note to last exception of MSC:</p> <p>" In case AckOptypes in MHP_ConID are used, the DUT (DSO) has to send the described dummy data pattern (0x00, 0x01, etc.) via MOST High either with or without a SenderHandle. Consequently, the tester has to check both alternatives."</p> <p style="text-align: right;">(based on errata)</p> <p>(2)</p> <p>Improve note:</p> <p>DUT is triggered to send 20 MOSTHigh packets.</p> <p>--&gt;</p> <p>DUT is triggered to send 20 times the MOSTHigh packet (application message) with suitable PacketSize (less than 64 kBytes) and the well-known data pattern.</p> <p style="text-align: right;">(based on errata)</p>
V1.0.2_023	4.2-5	<p>(1)</p> <p>Add note:</p> <p>The tester has to support up to 30 simultaneous MHP connections. In case mi_DSOCcount exceeds this number, the value of mi_DSOCcount has to reduced to 30.</p> <p>(2)</p> <p>Adapt Note:</p> <p>For every MOSTHigh connection, a unique TargetFBlockID, TargetInstID and TargetFktID has to be used.</p> <p>--&gt;</p> <p>For every MOSTHigh connection, a unique TargetFBlockID, TargetInstID and TargetFktID has to be used. For each trigger of MHP connection via FB ET using ET.DSO a different SenderHandle must be used.</p> <p style="text-align: right;">(based on errata)</p>

Change Ref.	Section	Changes
V1.0.2_024	Appendix 3	Deleted
V1.0.2_025	Document reference	Add reference to TTCN-3; part3

### Changes MOST Extended Core Compliance Test Specification: MOSTHigh V1.0 to MOST Extended Core Compliance Test Specification: MOSTHigh V1.0.1

Change Ref.	Section	Changes
V1.0.1_001	all	Consistent wording: FB_ET → ET (MHP-1)
V1.0.1_002	4.0-1	Implementation of ET.DSIDSOCCount.Get to replace ET.FktIDs.Get (MHP-2)
V1.0.1_003	4.1-4	MSC updated: FrameAck respond of DUT to NullFrame deleted as BlockAcknowledge mode is used. (MHP-3)
V1.0.1_004	4.1-4	MSC updated: Missing "ET" added in some comments (MHP-4)
V1.0.1_005	4.1-4	MSC updated: Now, tester will respond all requested frames to DUT until no further MultipleFramesRequest will be received. (MHP-5)
V1.0.1_006	4.1-4	MSC updated: Tester waits for reception of DSI.ResultAck before MHP connection will be closed. (MHP-6)
V1.0.1_007	4.1-2	MSC updated: Tester also checks for negative acknowledge → DUT not ok (7) (MHP-8)
V1.0.1_008	4.1-6	MSC updated: Deleted first part that uses scan of FB ET to determine number of supported MHP connections → DUT not ok (1) obsolete (MHP-9)
V1.0.1_009	4.1-1	MSC updated: Tester send "ReadyForData" after reception of "StartConnection" from DUT to cover complete flow of MHP Spec "mhc MH_Gen_EstablishConnection" (MHP-10)
V1.0.1_010	4.1-2	MSC updated: Deleted "ReadyForData" as already covered by test case 4.1-1 TEST_MH_DSI_RequestConnection_Ok (MHP-10)
V1.0.1_011	4.1-3	MSC updated: Deleted "ReadyForData" as already covered by test case 4.1-1 TEST_MH_DSI_RequestConnection_Ok (MHP-10)
V1.0.1_012	Appendices	Update according to changes above

Change Ref.	Section	Changes
V1.0.1_013	Document References; Input by device manufacturer; 4.0-1 4.1-1 4.1-3 4.1-4 4.2-1 4.2-2 4.2-3	MHP Test Spec 1.0.1 only for MHP Spec 2V3 (D104-4)
V1.0.1_014	4.1-6 4.2-5	MSC updated: ET.Configuration.Status → Configuration.Status
V1.0.1_015	4.1-2 4.1-3	MSC updated: "Wait for AIR..." deleted
V1.0.1_016	4.1-1 4.1-3	MSC Updated: NDF='1014' → NDF='1006'
V1.0.1_017	4.1-3	MSC updated: a) Note at 0-frame updated: " Value "NoFramesInBlock" must be greater than supported by DUT" → ""NoFramesInBlock" = Scale (reported by DUT via StartConnection)" b) Delete Exception, resulting in DUT ok (1) (DUT detects failure after reception of 0-frame)
V1.0.1_018	4.2-5	MSC update: ET.MHP_ConID.HoldConnectionRx(Event='0x02') and ET.MHP_ConID.HoldConnectionRx(Event='0x83' replaced by ET.MHP_ConID.HoldConnectionRx(Event='0x80')')
V1.0.1_019	4.1-5	Definition of test pattern to be sent to DUT: The first two bytes of the data pattern contain the sender handle. Then data pattern analog to FB.ET.DSO (0x3FE): The first data byte shall be 0x00, the seconds shall be 0x01, etc. The value is increased with every data byte. If 0xFF is reached, the next data byte starts with 0x00 again.
V1.0.1_020	4.1-2	Note of test description revised (how many data to be sent)
V1.0.1_021	4.1-2 4.1-3	MSC revised; Retry-Mechanism of DataFrame updated: DataFrame will be sent until timeout (r_trans+1)*t_trans; repetition delay: t_retrans
V1.0.1_022	4.2-4	MSC: t_1: t_Send → t_DeadLockShort Value of interest: Data pattern, transmitted via MOSTHigh
V1.0.1_023	4.2-2 4.2-3 4.2-5	Update note of test description: Test has to be performed with "NextPackageMode=0x00" (normal HOLD phase) and "AckMethod=0x01" (block acknowledge).
V1.0.1_024	whole document	NextPackageMode → NextPacketMethod AckMethod → AckMode NoFramesInBlock → NumFramesInBlock
V1.0.1_025	4.1-1	Value of interest deleted. T_Send deleted from measurement uncertainty table. (D106-8)
V1.0.1_026	4.1-2	T_Retrans not value of interest
V1.0.1_027	4.1-2 4.1-3	T_loop deleted; only t_2 (t_DeadLockMid) will be used (D106-3)



Change Ref.	Section	Changes
V1.0.1_028	4.1-2 4.1-3	Deviation from MHP Spec Rev. 2.3: : a) Start with CurrentFrameDSO = 1 b) When (CurrentFrameDSO < NumFramesInBlock) → when (CurrentFrameDS= <= NumFramesInBlock) (D106-4)
V1.0.1_029	4.1-2	Update Note of test description: Test has to be performed twice: 1. with data frame size smaller than NDFack: Number of data bytes of all frames: NDFack - 5 Total amount of user data in block = (NDFack – 5) * SCALE <i>Remark: “-5” because of quadlet oriented data transfer</i> 2. with data frame size identical to NDFack: Number of frames per block: SCALE Total amount of user data in block = NDFack * SCALE (D106-6)
V1.0.1_030	4.1-3	Device type: “All MHP supporting devices with DSI function” In case packet size not less 1014 byte, test case will end with “DUT ok (3)”. (D106-10)
V1.0.1_031	4.1-4	Add timeout t_DeadLockLong for reception of DSI.ResultAck → DUT not ok (4)
V1.0.1_032	4.1-4	Update MSI: At the end of test, DSI.ResultAck from DUT and end connection by tester will be performed at the same time (parallel box) (D106-9)
V1.0.1_033	4.2-1 4.2-4	Test case updated: In case NextPacketMethod=0x02 is not supported, test case will end with “DUT ok (2)”. (D106-7)
V1.0.1_034	4.2-3	T_ready_max → t_ready Value of interest (t_ready) deleted. T_ready deleted from measurement uncertainty table. (D106-11)
V1.0.1_035	4.2-3	Update MSC: Scale='64k_divided_by_NDF_of_DUT' → Scale=min('64k_divided_by_NDF_of_DUT',255) R_ready → r_start (D106-8; erroneously addressed to 4.1-1)
V1.0.1_036	4.2-4	ET.DSO.Error → ET.DSO.ErrorAck
V1.0.1_037	4.2-4	T_DeadLockMid → t_DeadLockLong (D106-5)
V1.0.1_038	4.2-4	MSC update: Delete “Stop t_1” before calling macro “MH_Gen_BasicFlow”; Add Note to stop t_1 as soon as first message received from DUT within the macro.
V1.0.1_039	Document References	Add reference to MOST Core Compliance Spec 3.0 and FB ET 3.0 (A106-5)
V1.0.1_040	Appendix 3	New appendix, containing GFTs

Notes:

Notes: