Multi-Channel Adaptation Protocol (MCAP)

***Bluetooth*® Test Suite**

- **Revision:** MCAP.TS.p5
- **Revision Date:** 2024-07-01
- **Prepared By:** BTI
- **Published during TCRL:** TCRL.2024-1

Contents

1 Scope

This Bluetooth document contains the Test Suite Structure (TSS) and test cases to test the implementation of the Bluetooth Multi-Channel Adaptation Protocol Specification (MCAP) with the objective to provide a high probability of air interface interoperability between the tested implementation and other manufacturers' Bluetooth devices.

2 References, definitions, and abbreviations

2.1 References

This document incorporates provisions from other publications by dated or undated reference. These references are cited at the appropriate places in the text, and the publications are listed hereinafter. Additional definitions and abbreviations can be found in [1], [2], and [3].

[1] Bluetooth Core Specification, Version 2.0 or later

[2] Test Strategy and Terminology Overview

[3] Multi-Channel Adaptation Protocol (MCAP) Specification

[4] ICS Proforma for Multi-Channel Adaptation Protocol (MCAP)

[5] IEEE Standards Style Manual, http://standards.ieee.org/guides/style/2000Style.pdf

[6] Multi-Channel Adaptation Protocol Implementation eXtra Information for Test, IXIT

2.2 Definitions

In this Bluetooth document, the definitions from [1], [2], and [3] apply.

Term	Definition
DCmax	Maximum number of simultaneously supported Data Channels. Refer to the identification of this term in the IXIT [6].
Initial Condition	Unless otherwise specified, the following Initial Conditions apply: Sink and Source are powered on, active, and within range of each other. The Lower Tester supports all device types (indicated by MDEP Data Type) that are also supported by the IUT (applies to test cases for standard op codes only).
Normal Condition	Unless otherwise specified, the following Normal Conditions apply: Standard environmental conditions such as nominal voltage, temperature and humidity for the IUT.

Table 2.1: MCAP definitions

2.3 Acronyms and abbreviations

In this Bluetooth document, the abbreviations from [1], [2], and [3] apply.

Abbreviation or Acronym	Meaning
CSP	Clock Synchronization Protocol
MCL	MCAP Communications Link
MDEP	MCAP Data End Point
MDL	MCAP Data Link

Table 2.2: Acronyms and abbreviations

3 Test Suite Structure (TSS)

3.1 Overview

This section defines the tree structure of the tests specified for MCAP also referred to as the Test Suite Structure (TSS). The TSS is composed of nested test groups organized in a top down approach.

The test groups are organized in 3 levels. The first level defines the procedure groups representing the profile services. The second level separates the procedures in functional modules. The last level in each branch contains the standard ISO groups BV and BI (not shown in TSS table).

3.2 Test Strategy

The test objectives are to verify the functionality of the Multi-Channel Adaptation Protocol within a Bluetooth Host and enable interoperability between Bluetooth Hosts on different devices. The testing approach covers mandatory and optional requirements in the specification and matches these to the support of the IUT as described in the ICS. Any defined test herein is applicable to the IUT if the ICS logical expression defined in the Test Case Mapping Table (TCMT) evaluates to true.

The test equipment provides an implementation of the Radio Controller and the parts of the Host needed to perform the test cases defined in this Test Suite. A Lower Tester acts as the IUT's peer device and interacts with the IUT over-the-air interface. The configuration, including the IUT, needs to implement similar capabilities to communicate with the test equipment. For some test cases, it is necessary to stimulate the IUT from an Upper Tester. In practice, this could be implemented as a special test interface, a Man Machine Interface (MMI), or another interface supported by the IUT.

This Test Suite contains Valid Behavior (BV) tests complemented with Invalid Behavior (BI) tests where required. The test coverage mirrored in the Test Suite Structure is the result of a process that started with catalogued specification requirements that were logically grouped and assessed for testability enabling coverage in defined test purposes.

3.3 Test groups

The following test groups have been defined:

- Connection Establishment
- Connection Management
- MCAP Error Handling
- Clock Synchronization Protocol

4 Test cases (TC)

4.1 Introduction

4.1.1 Test case identification conventions

Test cases are assigned unique identifiers per the conventions in [2]. The convention used here is: **<spec abbreviation>/<IUT role>/**<class>/**<feat>**/<func>/<subfunc>/<cap>/**<xx>-<nn>-<y>**.

Identifier Abbreviation	**Spec Identifier <spec abbreviation>**
MCAP	Multi-Channel Adaptation Protocol
Identifier Abbreviation	**Role Identifier <IUT role>**
SM	Sync Master
SRC-SNK	Source-Sink Role[1]
SS	Sync Slave
Identifier Abbreviation	**Feature Identifier <feat>**
CE	Connection Establishment
CM-ABT	Connection Management - Abort
CM-DEL	Connection Management - Delete
CM-DIS	Connection Management - Disconnect
CM-REC	Connection Management - Reconnect
CS-ERR	Clock Synchronization Error Handling
CS-I	Initiation of Clock Synchronization
CS-M	Clock Synchronization with Multiple MCLs
CS-R	Response to Clock Synchronization Request
ERR	MCAP Error Handling
INV	MCAP Invalid Behavior Tests

Table 4.1: MCAP TC feature naming conventions

4.1.2 Conformance

When conformance is claimed for a particular specification, all capabilities are to be supported in the specified manner. The mandated tests from this Test Suite depend on the capabilities to which conformance is claimed.

The Bluetooth Qualification Program may employ tests to verify implementation robustness. The level of implementation robustness that is verified varies from one specification to another and may be revised for cause based on interoperability issues found in the market.

Such tests may verify:

- That claimed capabilities may be used in any order and any number of repetitions not excluded by the specification
- That capabilities enabled by the implementations are sustained over durations expected by the use case
- That the implementation gracefully handles any quantity of data expected by the use case

[1] Test case where the IUT may be either Source or Sink, see also Section 4.1.3

- That in cases where more than one valid interpretation of the specification exists, the implementation complies with at least one interpretation and gracefully handles other interpretations
- That the implementation is immune to attempted security exploits

A single execution of each of the required tests is required to constitute a Pass verdict. However, it is noted that to provide a foundation for interoperability, it is necessary that a qualified implementation consistently and repeatedly pass any of the applicable tests.

In any case, where a member finds an issue with the test plan generated by the Bluetooth SIG qualification tool, with the test case as described in the Test Suite, or with the test system utilized, the member is required to notify the responsible party via an erratum request such that the issue may be addressed.

4.1.3 General Assumptions

Unless otherwise specified, the following assumptions apply to this document:

- Only a single ACL link exists between two MCAP devices.
- A point-to-point connection is used for all test cases.
- The MCAP IUT (implementation under test) may be a Source or Sink unless specified otherwise in a test case. In the role specific test cases, the MCAP IUT is required to be either in the Sync Slave or Sync Master role
- A suitable test system may be used for conformance tests.

4.1.4 Pass/Fail verdict conventions

Each test case has an Expected Outcome section. The IUT is granted the Pass verdict when all the detailed pass criteria conditions within the Expected Outcome section are met.

The convention in this Test Suite is that, unless there is a specific set of fail conditions outlined in the test case, the IUT fails the test case as soon as one of the pass criteria conditions cannot be met. If this occurs, then the outcome of the test is a Fail verdict.

4.2 Connection Establishment

Verify establishment of Data and Control Channels.

For this series of tests, the maximum number of simultaneously supported Data Channels that an IUT has declared in the IXIT [6] will be tested. The IXIT value defines DCmax to represent this quantity.

4.2.1 Control and Data Channel Establishment

Verify that support for establishment of Control and Data Channels is properly implemented.

MCAP/SRC-SNK/CE/BV-01-C [IUT initiates Control Channel and one or more Data Channels]

- Test Purpose

  Verify the connection procedure when the IUT initiates a Control Channel connection as well as one or more Data Channel connections.

- Reference

  [3] 4.1, 4.3

- Initial Condition
  - As defined in Section 2.2. Additionally, an ACL link has been established.
- Test Procedure
  1. The IUT initiates Control Channel connection.
  2. The Lower Tester accepts the Control Channel connection.
  3. The IUT uses MD_CREATE_MDL_REQ to initiate the maximum number of simultaneously supported Data Channel connections as declared in the PICS [4] (DCmax).
  4. The Lower Tester accepts the Data Channel connection(s).
  5. Data is exchanged on each of the Data Channels.
- Expected Outcome

  Pass verdict

  DCmax Data Channels can be opened simultaneously.

  Data is successfully exchanged on each Data Channel.

MCAP/SRC-SNK/CE/BV-02-C [IUT accepts Control Channel and one or more Data Channels]

- Test Purpose

  Verify the connection procedure when the IUT accepts a Control Channel connection as well as one or more Data Channel connections.
- Reference

  [3] 4.1.2
- Initial Condition
  - As defined in Section 2.2. Additionally, an ACL link has been established.
- Test Procedure
  1. The Lower Tester initiates a Control Channel connection.
  2. The IUT accepts the Control Channel connection.
  3. The Lower Tester uses MD_CREATE_MDL_REQ to initiate the maximum number of simultaneously supported Data Channel connection as declared in the PICS [4] (DCmax).
  4. The IUT accepts the Data Channel connection(s).
  5. Data is exchanged on each of the Data Channels.
- Expected Outcome

  Pass verdict

  DCmax Data Channels can be opened simultaneously.

  Data is successfully exchanged on each Data Channel.

MCAP/SRC-SNK/CE/BV-03-C [IUT initiates Control Channel and accepts one or more Data Channels]

- Test Purpose

  Verify a connection procedure when the IUT initiates a Control Channel connection and accepts one or more Data Channel connections.

- Reference

  [3] 4.1, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, an ACL link has been established.

- Test Procedure

  1. The IUT initiates a Control Channel connection.
  2. The Lower Tester accepts the Control Channel connection.
  3. The Lower Tester initiates the maximum number of simultaneously supported Data Channel connections as declared in the IXIT [6] (DCmax).
  4. The IUT accepts the Data Channel connection(s).
  5. Data is exchanged on each of the Data Channels.

- Expected Outcome

  Pass verdict

  DCmax Data Channels can be opened simultaneously.

  Data is successfully exchanged on each Data Channel.

MCAP/SRC-SNK/CE/BV-04-C [IUT accepts Control Channel and initiates one or more Data Channels]

- Test Purpose

  Verify a connection procedure when the IUT accepts a Control Channel connection and initiates one or more Data Channel connections.

- Reference

  [3] 4.1, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, an ACL link has been established.

- Test Procedure

  1. The Lower Tester initiates a Control Channel connection.
  2. The IUT accepts the Control Channel connection.
  3. The IUT initiates the maximum number of simultaneously supported Data Channel connections as declared in the IXIT [6] (DCmax).
  4. The Lower Tester accepts the Data Channel connection(s).
  5. Data is exchanged on each of the Data Channels.

- Expected Outcome

  Pass verdict

  DCmax Data Channels can be opened simultaneously.

  Data is successfully exchanged on each Data Channel.

4.3 Connection Management

Verify MCAP connection management procedures such as opening and disconnecting logical connections to devices, reconnecting, deleting MDLs, and aborting MDL connections. This set of tests validates support for the control commands as defined in MCAP other than the connection establishment commands.

For this series of tests, the maximum number of simultaneously supported Data Channels that an IUT has declared in the IXIT [6] will be tested. The IXIT value defines DCmax to represent this quantity.

4.3.1 MCL Disconnect

Verify that support for MCL Disconnect is properly implemented.

MCAP/SRC-SNK/CM-DIS/BV-01-C [IUT initiates MCL Disconnect]

- Test Purpose

  Verify proper behavior when the IUT initiates a graceful MCL Disconnect.

- Reference

  [3] 4.1, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, Control Channel and the maximum number of simultaneously supported Data Channel connections (DCmax) have been established.

- Test Procedure

  The IUT initiates an MCL disconnection.

- Expected Outcome

  Pass verdict

  All associated Data Channels are closed in addition to the Control Channel.

MCAP/SRC-SNK/CM-DIS/BV-02-C [IUT accepts MCL Disconnect]

- Test Purpose

  Verify proper behavior when the IUT accepts a graceful MCL Disconnect.

- Reference

  [3] 4.1, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and the maximum number of simultaneously supported Data Channel connections (DCmax) have been established.

- Test Procedure

  The Lower Tester initiates disconnection of all associated Data Channels followed by Control Channel disconnection.

- Expected Outcome

  Pass verdict

  All associated Data Channels are closed followed by the Control Channel.

  The IUT accepts the disconnection of the corresponding L2CAP channels.

MCAP/SRC-SNK/CM-DIS/BV-03-C [IUT in PENDING state while MCL is disconnected by the Lower Tester]

- Test Purpose

  Verify a proper response when the IUT is in PENDING state while the Control Channel is closed by the Lower Tester.

- Reference

  [3] 4.1, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established.

- Test Procedure

  1. The Lower Tester sends MD_CREATE_MDL to the IUT.
  2. The IUT sends “Success” response code and enters PENDING state.
  3. The Lower Tester closes the Control Channel.
  4. The Lower Tester reopens the Control Channel.

- Expected Outcome

  Pass verdict

  The Control Channel is closed.

  The Control Channel is successfully re-opened.

  The IUT returns to stable state and can process commands normally.

4.3.2 MDL Disconnect

Verify that support for MDL Disconnect is properly implemented.

MCAP/SRC-SNK/CM-DIS/BV-04-C [IUT initiates MDL Disconnect]

- Test Purpose

  Verify proper behavior when the IUT initiates a graceful MDL Disconnect.

- Reference

  [3] 4.1.2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and at least one Data Channel connection has been established.

- Test Procedure

  The IUT initiates an MDL disconnection.

- Expected Outcome

  Pass verdict

  The associated Data Channel is closed by the IUT.

MCAP/SRC-SNK/CM-DIS/BV-05-C [IUT accepts MDL Disconnect]

- Test Purpose

  Verify proper behavior when the IUT accepts a graceful MDL Disconnect.

- Reference

  [3] 4.1, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and at least one Data Channel connection has been established.

- Test Procedure

  The Lower Tester initiates MDL disconnection.

- Expected Outcome

  Pass verdict

  The associated Data Channel is closed by the Lower Tester.

  The IUT accepts the disconnection of the corresponding L2CAP Data Channel.

4.3.3 MDL Reconnect

Verify that MDL Reconnect command has been properly implemented.

MCAP/SRC-SNK/CM-REC/BV-01-C [IUT initiates MDL Reconnect following Control Channel disconnect]

- Test Purpose

  Verify proper behavior when the IUT initiates an MDL Reconnect following a graceful disconnection of the Control Channel.

- Reference

  [3] 4.1.3.7.2, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and the maximum number of simultaneously supported Data Channel connections (DCmax) have been established and data has been exchanged on each Data Channel.

- Test Procedure

  1. All Data Channel are gracefully disconnected by the IUT followed by the disconnection of the Control Channel (e.g., disconnect request before the IUT powers down).
  2. Conditions change to facilitate reconnection (e.g., the IUT is powered back on).
  3. A Control Channel connection is recreated by the IUT.
  4. The IUT initiates an MDL Reconnection of at least one Data Channel and the Lower Tester accepts (one reconnection request for each restored Channel).
  5. Data is exchanged on each of the restored Data Channels.

- Expected Outcome

  Pass verdict

  All associated Data Channels were disconnected by the IUT followed by the disconnection of the Control Channel.

  The Control Channel and at least one of the associated Data Channels are successfully restored.

  Data can be exchanged on each restored Data Channel.

MCAP/SRC-SNK/CM-REC/BV-02-C [IUT accepts MDL Reconnect following Control Channel disconnect]

- Test Purpose

  Verify proper behavior when the IUT accepts an MDL Reconnect following a graceful disconnection of a Control Channel.

- Reference

  [3] 4.1.3.7.2, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and the maximum number of simultaneously supported Data Channel connections (DCmax) have been established and data has been exchanged on each Data Channel.

- Test Procedure

  1. All Data Channel are gracefully disconnected by the Lower Tester (e.g., disconnect request before it powers down).
  2. Conditions change to facilitate reconnection (e.g., the Lower Tester is powered back on).
  3. A Control Channel connection is recreated by the Lower Tester.
  4. The Lower Tester initiates MDL Reconnection of at least one Data Channel and the IUT accepts (one reconnection request for each restored Channel).
  5. Data is exchanged on each of the restored Data Channels.

- Expected Outcome

  Pass verdict

  The Control Channel and all associated Data Channels were disconnected by the Lower Tester followed by the disconnection of the Control Channel.

  The Control Channel and at least one of the associated Data Channels are successfully restored.

  Data can be exchanged on each restored Data Channel.

MCAP/SRC-SNK/CM-REC/BV-03-C [IUT initiates MDL Reconnect following unintentional disconnect]

- Test Purpose

  Verify proper behavior when the IUT initiates an MDL Reconnect following unintentional disconnect.

- Reference

  [3] 4.1.3.7.2, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and the maximum number of simultaneously supported Data Channel connections (DCmax) have been established by the IUT and data has been exchanged on each Data Channel.

- Test Procedure

  1. Control and Data Channels are unintentionally (non-gracefully) disconnected (e.g., devices become “out of range”).
  2. Conditions change to facilitate reconnection (e.g., devices back “in range”).
  3. A Control Channel connection is recreated by the IUT.
  4. The IUT initiates an MDL Reconnection of at least one Data Channel and the Lower Tester accepts (one reconnection request for each restored Channel).
  5. Data is exchanged on each of the restored Data Channels.

- Expected Outcome

  Pass verdict

  A Control Channel and at least one of the associated Data Channels are successfully restored.

  Data can be exchanged on each restored Data Channel.

MCAP/SRC-SNK/CM-REC/BV-04-C [IUT accepts MDL Reconnect following unintentional disconnect]

- Test Purpose

  Verify proper behavior when the IUT accepts an MDL Reconnect following unintentional disconnect.

- Reference

  [3] 4.1.3.7.2, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and the maximum number of simultaneously supported Data Channel connections (DCmax) have been established by the Lower Tester and data has been exchanged on each Data Channel.

- Test Procedure

  1. Control and Data Channels are unintentionally (non-gracefully) disconnected (e.g., devices become “out of range”).
  2. Conditions change to facilitate reconnection (e.g., devices back “in range”).
  3. A Control Channel connection is recreated by the Lower Tester.
  4. The Lower Tester initiates an MDL Reconnection of at least one Data Channel and the IUT accepts (one reconnection request for each restored Channel).
  5. Data is exchanged on each of the restored Data Channels.

- Expected Outcome

  Pass verdict

  Control Channel and at least one of the associated Data Channels are successfully restored.

  Data can be exchanged on each restored Data Channel.

MCAP/SRC-SNK/CM-REC/BV-05-C [IUT initiates MDL Disconnect and Reconnect]

- Test Purpose

  Verify that the IUT can initiate an MDL Disconnect and Reconnect.

- Reference

  [3] 4.1.3.7.2, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and the maximum number of simultaneously supported Data Channel connections (DCmax) have been established and data has been exchanged on each Data Channel.

- Test Procedure

  1. One or more Data Channels is gracefully disconnected by the IUT by disconnecting the underlying L2CAP channels. For implementations that close all MDLs, the Control Channel may optionally be closed by the IUT.
  2. Once the IUT is prepared to reconnect, (after several seconds) the IUT initiates MDL Reconnect of one or more Data Channels and the Lower Tester accepts (one reconnect request for each restored Channel). If the Control Channel was disconnected by the IUT in the previous step, the IUT may need to re-establish the Control Channel before the MDL Reconnect.
  3. Data is exchanged on each of the restored Data Channels.

- Expected Outcome

  Pass verdict

  One or more Data Channels were disconnected by the IUT.

  One or more Data Channels were successfully reconnected by the IUT.

  Data was successfully exchanged on each restored Data Channel.

MCAP/SRC-SNK/CM-REC/BV-06-C [IUT accepts MDL Disconnect and Reconnect]

- Test Purpose

  Verify that the IUT can accept an MDL Disconnect and Reconnect.

- Reference

  [3] 4.1.3.7.2, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and the maximum number of simultaneously supported Data Channel connections (Dcmax) have been established and data has been exchanged on each Data Channel.

- Test Procedure

  1. One of the open Data Channels is gracefully disconnected by the Lower Tester by disconnecting the underlying L2CAP channel. For single MDL implementations, the Control Channel may optionally be closed by the IUT in reaction to all MDLs being disconnected.
  2. The Lower Tester initiates MDL Reconnection of the Data Channel and the IUT accepts. If the Control Channel was disconnected by the IUT in the previous step, the Lower Tester will need to re-establish the Control Channel before the MDL Reconnect.
  3. Data is exchanged on the restored Data Channel.

- Expected Outcome

  Pass verdict

  One of the open Data Channels was disconnected by the Lower Tester.

  The Data Channel was successfully reconnected by the Lower Tester.

  Data was successfully exchanged on the restored Data Channel.

4.3.4 MDL Delete

Verify that the MDL Delete command has been properly implemented.

MCAP/SRC-SNK/CM-DEL/BV-01-C [IUT initiates MDL Delete]

- Test Purpose

  Verify proper behavior when the IUT initiates MDL Delete.

- Reference

  [3] 4.1.3.7.4, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and one Data Channel have been established.

- Test Procedure

  1. The IUT initiates delete of an MDL ID.
  2. If there is a Data Channel open that is associated with MDL ID, the Lower Tester closes the associated Data Channel.
  3. The Lower Tester sends “Success” response code.
  4. The Lower Tester initiates reconnect of the MDL ID that was deleted.

- Expected Outcome

  Pass verdict

  Reconnect attempt fails and the IUT sends an “Invalid MDL” response code.

  The IUT continues to operate normally following a successful delete operation.

  The IUT returns to stable state and can process commands normally.

MCAP/SRC-SNK/CM-DEL/BV-02-C [IUT accepts MDL Delete]

- Test Purpose

  Verify proper behavior when the IUT accepts MDL Delete.

- Reference

  [3] 4.1.3.7.4/3, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and one Data Channel have been established.

- Test Procedure

  1. The Lower Tester initiates delete of MDL ID associated with the open Data Channel.
  2. The IUT closes the Data Channel.
  3. The IUT sends a “Success” response code.
  4. The Lower Tester initiates reconnect of the MDL ID that was deleted.
  5. The IUT sends an “Invalid MDL” response code.

- Expected Outcome

  Pass verdict

  The IUT closes Data Channel before the “Success” response code is sent.

  The IUT sends a “Success” response code.

  Reconnect attempt fails and the IUT sends an “Invalid MDL” response code.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/CM-DEL/BV-03-C [IUT initiates Delete of all MDLs using MDL ID 0xFFFF]

- Test Purpose

  Verify proper behavior when the IUT initiates use of reserved MDL ID 0xFFFF to delete all MDLs.

- Reference

  [3] 4.1.3.7.4/1

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and the maximum number of simultaneously supported Data Channel connections (DCmax) have been established.

- Test Procedure

  1. The IUT initiates delete of all MDLs by using MDL ID 0xFFFF and the Lower Tester accepts.
  2. If any Data Channels are open, the Lower Tester closes these.
  3. The Lower Tester sends “Success” response code.

- Expected Outcome

  Pass verdict

  The IUT continues to operate normally following a successful delete operation.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/CM-DEL/BV-04-C [IUT accepts Delete of all MDLs when MDL ID 0xFFFF is used]

- Test Purpose

  Verify proper behavior when the Lower Tester Initiates use of reserved MDL ID 0xFFFF to delete all MDLs.

- Reference

  [3] 4.1.3.7.4/1, 4.1.3.7.4/3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel and the maximum number of simultaneously supported Data Channel connections (DCmax) have been established.

- Test Procedure

  1. The Lower Tester initiates delete of all MDLs by using MDL ID 0xFFFF and the IUT accepts.
  2. The IUT closes all Data Channels.
  3. The IUT sends a “Success” response code.
  4. The Lower Tester initiates reconnect of each MDL ID that was deleted.
  5. The IUT sends an “Invalid MDL” response code for each request.

- Expected Outcome

  Pass verdict

  The IUT closes all Data Channels before the “Success” response code is sent.

  The IUT sends a “Success” response code.

  All reconnect attempts fail and the IUT sends “Invalid MDL” response codes for each request.

  The IUT returns to a stable state and can process commands normally.

4.3.5 MDL Abort

Verify that the MDL Abort command has been properly implemented.

MCAP/SRC-SNK/CM-ABT/BV-01-C [IUT sends MDL Abort]

- Test Purpose

  Verify proper behavior when the IUT sends MDL Abort to the Lower Tester.

- Reference

  [3] 4.1.3.7.3, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established.

- Test Procedure

  1. The IUT sends MD_CREATE_MDL to the Lower Tester.
  2. The Lower Tester sends a “Success” response code and enters the PENDING state.
  3. The IUT does not initiate the L2CAP connection.
  4. The IUT sends an Abort request for the same MDL used in the create request because of a) an Upper Tester (e.g., MMI action), or b) triggered by other means.
  5. The Lower Tester sends a “Success” response code.

- Expected Outcome

  Pass verdict

  Abort results in a “Success” response code from the Lower Tester.

  The IUT returns to the stable state and can process commands normally.

- Notes

  This test is applicable for IUTs that can support the ability to simulate sending of an Abort by some means; otherwise, this test is not applicable.

MCAP/SRC-SNK/CM-ABT/BV-02-C [IUT receives MDL Abort]

- Test Purpose

  Verify proper behavior when the IUT receives MDL Abort from the Lower Tester.

- Reference

  [3] 4.1.3.7.3, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established.

- Test Procedure

  1. The Lower Tester sends MD_CREATE_MDL to the IUT.
  2. The IUT sends a “Success” response code.
  3. The Lower Tester sends an Abort request for the same MDL used in the Create request.
  4. The IUT sends a “Success” response code.
  5. The Lower Tester attempts to open a Data Channel, but the connection is either refused or closed by the IUT immediately after being opened.

- Expected Outcome

  Pass verdict

  Create and Abort operations both result in “Success” response codes from the IUT.

  Attempt to open a Data Channel fails (either connection is refused, or channel is closed immediately after it is opened).

  The IUT returns to stable state and can process commands normally.

MCAP/SRC-SNK/CM-ABT/BV-03-C [MDL opened following earlier Abort by the Lower Tester]

- Test Purpose

  Verify proper behavior following an Abort by the Lower Tester.

- Reference

  [3] 4.1.3.7.3/2, 4.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established.

- Test Procedure

  1. The Lower Tester sends MD_CREATE_MDL to the IUT.
  2. The IUT Responds with a “Success” response code.
  3. The Lower Tester sends an abort request for the same MDL used in the create request.
  4. The IUT Responds with a “Success” response code.
  5. The Lower Tester sends a valid create request to the IUT (same or different MDL ID that was used in the create request of step 1).
  6. The IUT sends a “Success” response code.
  7. The Lower Tester opens a Data Channel and successfully exchanges data with the MDEP identified in the create operation of step 5.

- Expected Outcome

  Pass verdict

  The IUT responds to all operations with the “Success” response code.

  Data is exchanged with the correct MDL and correct MDEP.

4.4 MCAP Error Handling

Verify situations where protocol procedures may not be followed or cases such as race conditions where abnormal situations may arise. Some of the situations result in response codes and some do not. This set of tests provides further validation for the MCAP protocol.

4.4.1 Situations Resulting in Response Codes

Verify that support for error handling that involves response codes has been properly implemented. This includes situations where a response code is received by an IUT and situations that result in the generation of a response code by the IUT.

MCAP/SRC-SNK/ERR/BI-01-C [IUT receives invalid op code and sends Response Code]

- Test Purpose

  Verify proper response when the IUT receives an invalid op code.

- Reference

  [3] 4.1.3.2/3, 4.1.3.2/4, 4.1.3.4/1

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The Lower Tester sends a command to the IUT with an invalid op code (i.e., one that is not assigned in MCAP).
  2. The IUT sends the “Invalid Op Code” response code.

- Expected Outcome

  Pass verdict

  The IUT Response Packet includes op code of ERROR_RSP.

  The IUT sends “Invalid Op Code” response code.

  The IUT Response Packet includes MDL ID of 0x0000.

The IUT Response Packet Response Parameter Field is length zero.

The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-02-C [IUT receives Invalid Op Code Response Code]

- Test Purpose

  Verify proper response when the IUT receives an “Invalid Op Code” response code.

- Reference

  [3] 4.1.3.2/4, 4.1.3.4/1

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The IUT sends a command to the Lower Tester (i.e., any standard or clock synchronization command if clock synchronization is supported).
  2. The Lower Tester sends an “Invalid Op Code” response code to the IUT.

- Expected Outcome

  Pass verdict

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-03-C [IUT receives invalid parameter and sends Response Code]

- Test Purpose

  Verify proper response when the IUT receives an invalid parameter.

- Reference

  [3] 4.1.3.4/2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The Lower Tester sends a command to the IUT with invalid parameters (e.g., a MD_CREATE_MDL_REQ with 2 Bytes for the MDEP ID field).
  2. The IUT sends the “Invalid Parameter Value” response code.

- Expected Outcome

  Pass verdict

  The IUT sends the “Invalid Parameter Value” response code.

  The IUT Response Packet Response Parameter Field is length zero unless MD_CREATE_MDL command is used then this field is length of 1 octet.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-04-C [IUT receives Invalid Parameter Value Response Code]

- Test Purpose

  Verify proper response when the IUT receives an “Invalid Parameter Value” response code.

- Reference

  [3] 4.1.3.4/2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The IUT sends a command to the Lower Tester (i.e., any standard or clock synchronization command if clock synchronization is supported)
  2. The Lower Tester sends the IUT “Invalid Parameter Value” response code.

- Expected Outcome

  Pass verdict

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-05-C [IUT receives Data Channel connection request with invalid MDEP ID and sends Response Code]

- Test Purpose

  Verify proper response when the IUT receives an initial Data Channel connection request with an invalid MDEP ID.

- Reference

  [3] 4.1.3.4/3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The Lower Tester sends MD_CREATE_MDL_REQ with invalid MDEP ID (i.e., an MDEP ID that is not present in the device or an MDEP ID in the reserved range).
  2. The IUT sends an “Invalid MDEP” response code.

- Expected Outcome

  Pass verdict

  The IUT rejects the Data Channel connection.

  The IUT sends the “Invalid MDEP” response code.

  The IUT Response Packet Response Parameter Field is 1 octet length.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-06-C [IUT receives Invalid MDEP Response Code]

- Test Purpose

  Verify proper response to an “Invalid MDEP” response code.

- Reference

  [3] 4.1.3.4/3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The IUT sends the Lower Tester the MD_CREATE_MDL_REQ command.
  2. The Lower Tester sends an “Invalid MDEP” response code to the IUT.

- Expected Outcome

  Pass verdict

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-07-C [IUT is busy]

- Test Purpose

  Verify proper response when the IUT is busy.

- Reference

  [3] 4.1.3.4/4

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The Lower Tester creates significantly more MDLs to a single MDEP than the IUT would typically manage simultaneously.
  2. The IUT opens all MDLs or sends an appropriate error response code.

- Expected Outcome

  Pass verdict

  The IUT sends an “MDEP Busy” or “MDL Busy” or “Resource Unavailable” response code, or succeeds in opening all MDLs.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-08-C [IUT receives MDEP Busy Response Code]

- Test Purpose

  Verify a proper response to the “MDEP Busy” response code.

- Reference

  [3] 4.1.3.4/4

- Initial Condition
  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.
- Test Procedure
  1. The IUT sends the MD_CREATE_MDL_REQ or MD_RECONNECT_MDL_REQ commands to the Lower Tester.
  2. The Lower Tester sends the “MDEP Busy” response code to the IUT.
- Expected Outcome

  Pass verdict

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-09-C [IUT receives initial connection request with an invalid MDL ID and sends Response Code]

- Test Purpose

  Verify a proper response when the IUT receives an initial Data Channel connection request with an invalid MDL ID (i.e., 0x0000 or 0xFF00 - 0xFFFF).
- Reference

  [3] 4.1.3.4/5
- Initial Condition
  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.
- Test Procedure
  1. The Lower Tester sends MD_CREATE_MDL_REQ with invalid MDL (i.e., 0x0000 or 0xFF00 - 0xFFFF).
  2. The IUT rejects the MDL connection.
  3. The IUT sends the “Invalid MDL” response code.
- Expected Outcome

  Pass verdict

  The IUT sends the “Invalid MDL” response code.

  The IUT Response Packet Response Parameter Field is 1 octet in length.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-10-C [IUT receives reconnect request for MDL ID that is no longer valid and sends Response Code]

- Test Purpose

  Verify a proper response when a device attempts to reconnect to an MDL that has been deleted and is no longer valid.
- Reference

  [3] 4.1.3.4/5

- Initial Condition
  - As defined in Section 2.2. Additionally, a Control Channel and one Data Channel connection have been established.
- Test Procedure
  1. The Lower Tester deletes an MDL.
  2. The Lower Tester attempts to reconnect to the deleted MDL.
  3. The IUT sends the “Invalid MDL” response code.
- Expected Outcome

  Pass verdict

  The IUT sends the “Invalid MDL” response code.

  The IUT Response Packet Response Parameter Field is length zero unless MD_CREATE_MDL command was used; then, this field is 1 octet in length.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-11-C [IUT receives Invalid MDL Response Code]

- Test Purpose

  Verify a proper response to the “Invalid MDL” response code.
- Reference

  [3] 4.1.3.4/5
- Initial Condition
  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.
- Test Procedure
  1. The IUT sends the MD_CREATE_MDL_REQ or MD_RECONNECT_MDL_REQ commands to the Lower Tester.
  2. The Lower Tester sends the “Invalid MDL” response code to the IUT.
- Expected Outcome

  Pass verdict

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-12-C [IUT receives MDL Busy Response Code]

- Test Purpose

  Verify a proper response to the “MDL Busy” response code.
- Reference

  [3] 4.1.3.4/6
- Initial Condition
  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The IUT sends the MD_CREATE_MDL_REQ or MD_RECONNECT_MDL_REQ commands to the Lower Tester.
  2. The Lower Tester sends the “MDL Busy” response code.

- Expected Outcome

  Pass verdict

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-13-C [IUT receives invalid command while not in PENDING state and sends Response Code]

- Test Purpose

  Verify a proper response when the IUT receives a MD_ABORT_MDL command while not in the PENDING state.

- Reference

  [3] 4.1.3.4/7

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The IUT is not in the PENDING state.
  2. The Lower Tester sends the IUT an Abort request.
  3. The IUT sends the “Invalid Operation” response code.

- Expected Outcome

  Pass verdict

  The IUT sends the “Invalid Operation” response code.

  The IUT Response Packet Response Parameter Field is length zero.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-14-C [IUT receives invalid command while in PENDING state and sends Response Code]

- Test Purpose

  Verify a proper response when the IUT receives a standard (i.e., not clock synchronization) command other than MD_ABORT_MDL while in the PENDING state.

- Reference

  [3] 4.1.3.7/2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The Lower Tester sends MD_CREATE_MDL to the IUT.
  2. The IUT sends the “Success” response code and enters the PENDING state.
  3. The Lower Tester sends the invalid commands MD_CREATE_MDL, MD_RECONNECT_MDL, and MD_DELETE_MDL (i.e., any command other than MD_ABORT_MDL and clock synchronization commands).
  4. The IUT sends the “Invalid Operation” response code for each invalid command sent.

- Expected Outcome

  Pass verdict

  The IUT sends the “Invalid Operation” response code for each invalid command received.

  The IUT Response Packet Response Parameter Field is length zero unless MD_CREATE_MDL command was used; then, this field is 1 octet in length.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-15-C [IUT receives Invalid Operation Response Code]

- Test Purpose

  Verify a proper response to the “Invalid Operation” response code.

- Reference

  [3] 4.1.3.4/7

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The IUT sends the Lower Tester any standard command or any clock synchronization command.
  2. The Lower Tester sends the Invalid Operation response code.

- Expected Outcome

  Pass verdict

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-16-C [IUT is temporarily unavailable and sends Response Code]

- Test Purpose

  Verify a proper response when the Lower Tester attempts to communicate with an IUT that is temporarily unavailable.

- Reference

  [3] 4.1.3.4/8

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. Set the IUT in a condition that will result in it being “temporarily unavailable” (see Notes section).
  2. The Lower Tester attempts to communicate with the IUT by sending any command.
  3. The IUT sends the “Resource Unavailable” response code.

- Expected Outcome

  Pass verdict

  The IUT sends the “Resource Unavailable” response code.

  The IUT Response Packet Response Parameter Field is length zero unless the MD_CREATE_MDL command was used; then, this field is 1 octet in length.

  The IUT returns to a stable state and can process commands normally.

- Notes

  This test is applicable for IUTs that can support the ability to get into a “temporarily unavailable” condition by some means; otherwise, this test is not applicable.

MCAP/SRC-SNK/ERR/BI-17-C [IUT receives Resource Unavailable Response Code]

- Test Purpose

  Verify a proper response to the “Resource Unavailable” response code.

- Reference

  [3] 4.1.3.4/8

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The IUT sends any command to the Lower Tester.
  2. The Lower Tester sends the “Resource Unavailable” response code.

- Expected Outcome

  Pass verdict

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-18-C [IUT receives Unspecified Error Response Code]

- Test Purpose

  Verify a proper response to the “Unspecified Error” response code.

- Reference

  [3] 4.1.3.4, 4.1.3.4/9

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The IUT sends any command to the Lower Tester.
  2. The Lower Tester sends the “Unspecified Error” response code.

- Expected Outcome

  Pass verdict

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-19-C [IUT receives unsupported clock sync commands and sends Response Code]

- Test Purpose

  Verify a proper response when an IUT that does not support the Clock Synchronization Protocol receives clock synchronization commands.

- Reference

  [3] 4.1.3.2/4, 4.1.3.4/10, 5.0/1

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The Lower Tester attempts to initiate clock synchronization with an IUT that does not support the Clock Synchronization Protocol (e.g., the Lower Tester sends unsupported clock synchronization command).

- Expected Outcome

  Pass verdict

  The IUT sends the “Request Not Supported” response code.

  The op code field of the Response Packet is set to the Response op code that corresponds to the unsupported Request op code.

  The IUT Response Packet Response Parameter Field (bytes 3–9 for MD_SYNC_CAP_RSP or bytes 3–16 of MD_SYNC_SET_RSP) is set to zero.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/ERR/BI-20-C [IUT receives Request Not Supported Response Code]

- Test Purpose

  Verify a proper response when the IUT requests clock synchronization with a Lower Tester that does not support the Clock Synchronization Protocol and receives the “Request Not Supported” response code.

- Reference

  [3] 4.1.3.4/10, 5.0/2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The IUT requests clock synchronization with a Lower Tester that advertises support for the Clock Synchronization Protocol but does not support the Clock Synchronization Protocol (see Notes section).
  2. The Lower Tester sends the “Request Not Supported” response code.
  3. The IUT reacts by not continuing to send clock synchronization commands.

- Expected Outcome

  Pass verdict

  The IUT no longer sends clock synchronization commands.

  The IUT remains in a stable state and can process commands normally.

- Notes

  In this test, the Lower Tester may advertise support for the Clock Synchronization Protocol in SDP but may respond that it does not support the Clock Synchronization Protocol for the purpose of the Invalid Behavior test.

4.4.2 Situations Not Resulting in Response Codes

Verify that procedures for handling uncommon cases and corner cases in MCAP have been properly considered in implementation.

MCAP/SRC-SNK/INV/BI-01-C [Control Channel is closed before a response is sent]

- Test Purpose

  Verify a proper response when a Control Channel is closed by the Lower Tester while the IUT is waiting for a response.

- Reference

  [3] 4.1.3/3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established by the IUT.

- Test Procedure

  1. The IUT sends a command on the Control Channel.
  2. The Lower Tester does not respond and closes the Control Channel.

- Expected Outcome

  Pass verdict

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/INV/BI-02-C [Out-of-sequence operation - IUT as Acceptor]

- Test Purpose

  Verify a proper response for case when the IUT is the Control Channel Acceptor and receives a request from the Lower Tester while the IUT is waiting for a response from it.

- Reference

  [3] 4.1.3/5

- Initial Condition

  As defined in Section 2.2. Additionally, a Control Channel connection has been established by the Lower Tester.

- Test Procedure

  1. The IUT sends any request command on the Control Channel.
  2. The Lower Tester does not respond to the request but sends a request on the Control Channel.
  3. The IUT responds to the request from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT processes the request from the Lower Tester as usual.

  The IUT remains in a stable state and can process commands normally.

MCAP/SRC-SNK/INV/BI-03-C [Out-of-sequence operation - IUT as Initiator]

- Test Purpose

  Verify a proper response for the case when the IUT is the Control Channel Initiator and receives a request from the Lower Tester while waiting for a response from it.

- Reference

  [3] 4.1.3/5

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established by the IUT.

- Test Procedure

  1. The IUT sends any request command on the Control Channel.
  2. The Lower Tester does not respond to the request but sends a request on the Control Channel.
  3. The IUT does not respond to the request from the Lower Tester.
  4. After some delay, the Lower Tester responds to the original request as normal.

- Expected Outcome

  Pass verdict

  The IUT ignores the extraneous request from the Lower Tester.

  The IUT receives the response and processes it as usual.

  The IUT remains in a stable state and can process commands normally.

MCAP/SRC-SNK/INV/BI-04-C [Out-of-sequence operation - Response without corresponding Request]

- Test Purpose

  Verify a proper response for case when the IUT receives a response from a Lower Tester that does not correspond to an outstanding request.

- Reference

  [3] 4.1.3/6

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established by the IUT, no request is outstanding.

- Test Procedure

  1. The Lower Tester sends a response packet on the Control Channel.

- Expected Outcome

  Pass verdict

  The IUT ignores the response packet.

  The IUT remains in a stable state and can process commands normally.

MCAP/SRC-SNK/INV/BI-05-C [MDL ID already exists during an MDL Create by the Lower Tester]

- Test Purpose

  Verify a proper response when an MDL ID already exists for an MDL during an MD_CREATE_MDL by the Lower Tester.

- Reference

  [3] 4.1.3.7.1/2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The Lower Tester creates MDL ID 3.
  2. The Lower Tester creates MDL ID 3 again (without closing) with a fresh context.

- Expected Outcome

  Pass verdict

  At step 1, Data Channel is opened and “Success” is returned.

  At step 2, Data Channel is closed by the IUT and “Success” is returned. Another Data Channel is accepted by the IUT according to the new context.

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/INV/BI-06-C [Lower Tester attempts to connect to Control Channel of same device using same PSM as open Control Channel]

- Test Purpose

  Verify a proper response when the Lower Tester attempts to connect to the Control Channel of the same device using the same PSM as an already open channel.

- Reference

  [3] 4.1

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established.

- Test Procedure

  1. The Lower Tester attempts to connect to the Control Channel of an IUT using the same PSM as the Control Channel that is already open on the same device.
  2. The IUT either:

     a) Refuses the L2CAP connection and sends a “Resource Unavailable” response code

     OR

     b) Quickly closes down the L2CAP connection without sending or responding to any data traffic

- Expected Outcome

  Pass verdict

  A second connection attempt does not succeed (it is either refused or closed immediately by the IUT).

  The IUT returns to a stable state and can process commands normally.

MCAP/SRC-SNK/INV/BI-07-C [Lower Tester attempts to connect to Data Channel without having sent a Create or Reconnect request]

- Test Purpose

  Verify a proper response when the Lower Tester attempts to connect to the Data Channel of device without having sent a CREATE_MDL or RECONNECT_MDL request.

- Reference

  [3] 4.2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established.

- Test Procedure

  1. The Lower Tester attempts to create an L2CAP channel to the Data Channel PSM without first sending a CREATE_MDL or RECONNECT_MDL request.
  2. The IUT either:

     a) Refuses the L2CAP connection using L2CAP Connection Response “Connection refused – no resources available”

     OR

     b) Quickly closes down the L2CAP connection without sending or responding to any data traffic

- Expected Outcome

  Pass verdict

  The Data Channel connection attempt does not succeed (it is either refused or closed immediately by the IUT).

  The IUT returns to a stable state and can process commands normally.

4.5 MCAP Clock Synchronization

Verify Clock Synchronization Protocol procedures.

4.5.1 IUT Initiates Synchronization (IUT as Sync-Master)

Verify that clock synchronization has been properly implemented for the case where the IUT acts as the Sync-Master.

MCAP/SM/CS-I/BV-01-C [IUT requests clock synchronization]

- Test Purpose

  Verify proper behavior when the IUT requests clock synchronization for a case when the Bluetooth half-slot is set to valid number and TimeStamp_SyncTime differs from 0xFFFFFFFF FFFFFFFF (64 bits).

- Reference

  [3] 5.2.3.1, 5.2.3.2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established. The IUT supports the Clock Synchronization Protocol.

- Test Procedure

  1. The IUT requests clock synchronization capability and the Lower Tester responds.
  2. The IUT requests clock synchronization and the Lower Tester responds.
  3. Synchronization is performed.

- Expected Outcome

  Pass verdict

  The IUT sends an MD_SYNC_CAP_REQ to the Lower Tester with a TimeStamp_RequiredAccuracy value that will be accepted by the Lower Tester.

  The IUT receives the MD_SYNC_CAP_RSP from the Lower Tester.

  The IUT sends a MD_SYNC_SET_REQ with BluetoothClock_SyncTime to a valid Bluetooth clock value (Bluetooth clock might wrap around) that is at least SyncLeadTime but less than 60 seconds in the future of the current Bluetooth clock value so that the Lower Tester will be able to perform the operation. TimeStamp_SyncTime is set to a value that differs from 0xFFFFFFFF FFFFFFFF (64 bits).

  The IUT receives the MD_SYNC_SET_RSP.

  The Time-Stamp Clock is normalized to microseconds.

  The IUT does not perform piconet Role switch during the synchronization operation.

MCAP/SM/CS-I/BV-02-C [IUT requests instant synchronization]

- Test Purpose

  Verify proper behavior when the IUT requests instant synchronization for case when Bluetooth Clock set to 0xFFFFFFFF and TimeStamp_SyncTime differs from 0xFFFFFFFF FFFFFFFF (64 bits).

- Reference

  [3] 5.2.3.1, 5.2.3.2

- Initial Condition

  As defined in Section 2.2. Additionally, a Control Channel has been established. The IUT supports the Clock Synchronization Protocol.

- Test Procedure

  1. The IUT requests clock synchronization capability and the Lower Tester responds.
  2. The IUT requests instant clock synchronization by setting the Bluetooth Clock to 0xFFFFFFFF and the Lower Tester responds.
  3. Synchronization is performed.

- Expected Outcome

  Pass verdict

  The IUT sends an MD_SYNC_CAP_REQ to the Lower Tester with a TimeStamp_RequiredAccuracy value that will be accepted by the Lower Tester.

  The IUT receives an MD_SYNC_CAP_RSP from the Lower Tester.

  The IUT sends an MD_SYNC_SET_REQ with Bluetooth Clock set to 0xFFFFFFFF to the Lower Tester. TimeStamp_SyncTime is set to a value that differs from 0xFFFFFFFF FFFFFFFF (64 bits).

  The IUT receives an MD_SYNC_SET_RSP from the Lower Tester.

  TimeStamp_SyncTime does match parameter of the MD_SYNC_SET_REQ (with the TimeStamp_SampleAccuracy parameter taken in account).

  The Time-Stamp Clock is normalized to microseconds.

  The IUT does not perform piconet Role switch during the synchronization operation.

MCAP/SM/CS-I/BV-03-C [IUT acts as Sync-Master and Sync-Slave simultaneously]

- Test Purpose

  Verify proper behavior when the IUT acts as Sync-Master and Sync-Slave simultaneously.

- Reference

  [3] 5.2.3.1, 5.2.3.2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established. The IUT supports the Clock Synchronization Protocol.

- Test Procedure

  1. The IUT requests clock synchronization capability and the Lower Tester responds.
  2. The IUT requests clock synchronization with a Lower Tester that supports the CSP.

3. The Lower Tester requests clock synchronization capability and the IUT responds.
4. The Lower Tester requests clock synchronization with an IUT that supports the CSP and IUT responds.
5. The Lower Tester responds to the IUT clock synchronization request from step 2. This may occur before the IUT responds in step 4 depending on how the actual synchronization time is set.
6. Synchronization is performed.

- Expected Outcome

  Pass verdict

  The IUT sends an MD_SYNC_CAP_REQ to the Lower Tester with a TimeStamp_RequiredAccuracy value that will be accepted by the Lower Tester.

  The IUT receives the MD_SYNC_CAP_RSP from the Lower Tester.

  The IUT sends an MD_SYNC_SET_REQ with BluetoothClock_SyncTime to a valid Bluetooth clock value (Bluetooth clock might wrap around) that is at least SyncLeadTime but less than 60 seconds in the future of the current Bluetooth clock value so that the Lower Tester will be able to perform the operation. TimeStamp_SyncTime is set to a value that differs from 0xFFFFFFFF FFFFFFFF (64 bits).

  The IUT receives the MD_SYNC_CAP_REQ from the Lower Tester with a TimeStamp_RequiredAccuracy value that will be accepted by the IUT.

  The IUT sends the MD_SYNC_CAP_RSP to the Lower Tester.

  The IUT receives an MD_SYNC_SET_REQ with BluetoothClock_SyncTime to a valid Bluetooth clock value (Bluetooth clock might wrap around) that is at least SyncLeadTime but less than 60 seconds in the future of the current Bluetooth clock value so that the IUT will be able to perform the operation. TimeStamp_SyncTime is set to a value that differs from 0xFFFFFFFF FFFFFFFF (64 bits).

  The IUT sends the MD_SYNC_SET_RSP to the Lower Tester.

  The IUT receives the MD_SYNC_SET_RSP from the Lower Tester.

  The Time-Stamp Clock is normalized to microseconds.

  The IUT does not perform piconet Role switch during the synchronization operation.

  TimeStamp_SyncTime does match parameter of the MD_SYNC_SET_REQ (with the TimeStamp_SampleAccuracy parameter taken in account).

MCAP/SM/CS-I/BV-04-C [IUT initiates update of clock instances]

- Test Purpose

  Verify proper behavior when the IUT initiates an update of the clock instances to the Sync-Slave.

- Reference

  [3] 5.2.3.1, 5.2.3.2, 5.2.3.3

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established. Clock synchronization was previously performed successfully.

- Test Procedure

  1. The IUT initiates an update of the clock instances to the Lower Tester and the Lower Tester responds.
  2. Synchronization update is performed.

- Expected Outcome

  Pass verdict

  The IUT sends an MD_SYNC_SET_REQ with BluetoothClock_SyncTime to a valid Bluetooth clock value (Bluetooth clock might wrap around) that is at least SyncLeadTime but less than 60 seconds in the future of the current Bluetooth clock value so that the Lower Tester will be able to perform the operation. TimeStamp_SyncTime is set to a value that differs from 0xFFFFFFFF FFFFFFFF (64 bits). TimeStamp_UpdateInformation is set to 1.

  The IUT receives the MD_SYNC_SET_RSP from the Lower Tester.

  The IUT receives the MD_SYNC_INFO_IND from the Lower Tester.

  The Time-Stamp Clock is normalized to microseconds.

  The IUT does not perform piconet Role switch during the synchronization operation.

4.5.2 IUT Responds to Synchronization Request (IUT as Sync-Slave)

Verify that clock synchronization has been properly implemented for case where the IUT acts as the Sync-Slave.

MCAP/SS/CS-R/BV-01-C [IUT responds to clock synchronization request]

- Test Purpose

  Verify proper behavior when the IUT responds to a clock synchronization request when a Bluetooth half-slot is set to valid number and TimeStamp_SyncTime differs from 0xFFFFFFFF FFFFFFFF (64 bits).

- Reference

  [3] 5.2.3.1, 5.2.3.2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established. The IUT supports the Clock Synchronization Protocol.

- Test Procedure

  1. The Lower Tester requests clock synchronization capability and the IUT responds.
  2. The Lower Tester requests clock synchronization and the IUT responds.
  3. Synchronization is performed.

- Expected Outcome

  Pass verdict

  The IUT receives the MD_SYNC_CAP_REQ from the Lower Tester with a TimeStamp_RequiredAccuracy value that will be accepted by the IUT.

  The IUT sends the MD_SYNC_CAP_RSP to the Lower Tester.

  The IUT receives an MD_SYNC_SET_REQ with BluetoothClock_SyncTime to a valid Bluetooth clock value (Bluetooth clock might wrap around) that is at least SyncLeadTime but less than 60 seconds in the future of the current Bluetooth clock value so that the IUT will be able to perform the operation. TimeStamp_SyncTime is set to a value that differs from 0xFFFFFFFF FFFFFFFF (64 bits).

  The IUT sends the MD_SYNC_SET_RSP.

The Time-Stamp Clock is normalized to microseconds.

The IUT does not perform piconet Role switch during the synchronization operation.

TimeStamp_SyncTime does match parameter of the MD_SYNC_SET_REQ (with the TimeStamp_SampleAccuracy parameter taken in account).

MCAP/SS/CS-R/BV-02-C [IUT responds to instant synchronization request]

- Test Purpose

  Verify proper behavior when the Lower Tester requests instant synchronization for a case when Bluetooth Clock is set to 0xFFFFFFFF and TimeStamp_SyncTime differs from 0xFFFFFFFF FFFFFFFF (64 bits).

- Reference

  [3] 5.2.3.1, 5.2.3.2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established. The IUT supports the Clock Synchronization Protocol.

- Test Procedure

  1. The Lower Tester requests clock synchronization capability and the IUT responds.
  2. The Lower Tester requests instant clock synchronization by setting the Bluetooth Clock to 0xFFFFFFFF and the IUT responds.
  3. Synchronization is performed.

- Expected Outcome

  Pass verdict

  The IUT receives the MD_SYNC_CAP_REQ from the Lower Tester with a TimeStamp_RequiredAccuracy value that will be accepted by the IUT.

  The IUT sends the MD_SYNC_CAP_RSP to the Lower Tester.

  The IUT receives the MD_SYNC_SET_REQ with Bluetooth Clock set to 0xFFFFFFFF. TimeStamp_SyncTime is set to a value that differs from 0xFFFFFFFF FFFFFFFF (64 bits).

  The IUT sends the MD_SYNC_SET_RSP.

  The Time-Stamp Clock is normalized to microseconds.

  The IUT does not perform piconet Role switch during the synchronization operation.

  TimeStamp_SyncTime does match parameter of the MD_SYNC_SET_REQ (with the TimeStamp_SampleAccuracy parameter taken in account).

MCAP/SS/CS-R/BV-03-C [IUT responds to request for update of clock instances from Sync-Master]

- Test Purpose

  Verify proper behavior when the IUT responds to a request for an update of the clock instances from the Sync-Master.

- Reference

  [3] 5.2.3.1, 5.2.3.2

- Initial Condition
  - As defined in Section 2.2. Additionally, a Control Channel has been established. The IUT supports the Clock Synchronization Protocol.
- Test Procedure
  1. The Lower Tester (Sync-Master) initiates an update of the clock instances to the IUT (Sync-Slave) and the IUT responds.
  2. Synchronization update is performed.
- Expected Outcome

  Pass verdict

  TimeStamp_SyncTime is set to a value that differs from 0xFFFFFFFF FFFFFFFF (64 bits). TimeStamp_UpdateInformation is set to 1.

  The IUT sends the MD_SYNC_SET_RSP to the Lower Tester.

  The IUT sends the MD_SYNC_INFO_IND to the Lower Tester within the SyncLeadTime.

  The IUT sends the MD_SYNC_INFO_IND to the Lower Tester again within the Time-Stamp Update Interval (see Notes below).

  The Time-Stamp Clock is normalized to microseconds.

  The IUT does not perform piconet Role switch during the synchronization operation.

  TimeStamp_SyncTime does match parameter of the MD_SYNC_SET_REQ (with the TimeStamp_SampleAccuracy parameter taken in account).
- Notes

  The Time-Stamp Update Interval value is not directly available but can be easily calculated as TimeStamp_RequiredAccuracy (from Sync-Master) divided by TimeStamp_NativeAccuracy (from Sync-Slave).

MCAP/SS/CS-R/BV-04-C [IUT in PENDING state responds to clock synchronization request]

- Test Purpose

  Verify proper behavior when the IUT in the PENDING state responds to a clock synchronization request when a Bluetooth half-slot is set to 0xFFFFFFFF and TimeStamp_SyncTime differs from 0xFFFFFFFF FFFFFFFF (64 bits).
- Reference

  [3] 5.2.3.1, 5.2.3.2, 4.1.2
- Initial Condition
  - As defined in Section 2.2. Additionally, a Control Channel has been established. The IUT supports the Clock Synchronization Protocol in addition to MCAP standard op codes.
- Test Procedure
  1. The Lower Tester sends MD_CREATE_MDL to the IUT.
  2. The IUT sends a “Success” response code and enters the PENDING state (awaiting L2CAP connection request).
  3. The Lower Tester requests clock synchronization capability and the IUT responds.

4. The Lower Tester requests clock synchronization and the IUT responds.
5. Synchronization is performed.
6. The Lower Tester sends L2CAP connection request and the IUT accepts.
7. Data is exchanged on the created Data Channel.

- Expected Outcome

  Pass verdict

  The IUT responds with “Success” for the MD_CREATE_MDL.

  The IUT receives the MD_SYNC_CAP_REQ from the Lower Tester with a TimeStamp_RequiredAccuracy value that will be accepted by the IUT.

  The IUT sends the MD_SYNC_CAP_RSP to the Lower Tester.

  The IUT receives an MD_SYNC_SET_REQ with BluetoothClock_SyncTime to 0xFFFFFFFF. TimeStamp_SyncTime is set to a value that differs from 0xFFFFFFFF FFFFFFFF (64 bits).

  The IUT sends the MD_SYNC_SET_RSP.

  The Time-Stamp Clock is normalized to microseconds.

  The IUT does not perform piconet Role switch during the synchronization operation.

  TimeStamp_SyncTime does match the parameter of the MD_SYNC_SET_REQ (with the TimeStamp_SampleAccuracy parameter taken in account).

  Following clock synchronization, the IUT accepts the Data Channel connection.

  Data is successfully exchanged on the created Data Channel.

4.5.3 Multi-MCL Handling

Verify proper behavior when the IUT supports multiple MCLs simultaneously.

MCAP/SRC-SNK/CS-M/BV-01-C [IUT responds to clock synchronization when multiple MCLs exist]

- Test Purpose

  Verify proper behavior when the IUT as Sync-Slave receives clock synchronization with a device when two or more MCLs exist.

- Reference

  [3] 5.3

- Initial Condition

  - As defined in Section 2.2. Additionally, the Control Channels for two MCLs (MCL-A, MCL-B) and optional additional MCLs have been established. The IUT supports the Clock Synchronization Protocol.

- Test Procedure

  1. The Lower Tester requests clock synchronization capability on MCL-A and the IUT responds.
  2. The Lower Tester requests clock synchronization capability on MCL-B and the IUT responds.
  3. The Lower Tester requests clock synchronization on MCL-A with TimeStamp_SyncTime to 0x00000000 00000000 (64 bits) and the IUT responds.
  4. The Lower Tester requests clock synchronization on MCL-B with TimeStamp_SyncTime to 0x80000000 00000000 (64 bits) and the IUT responds.

5. The Lower Tester requests clock synchronization on MCL-A with BluetoothClock_SyncTime to 0xFFFFFFFF. TimeStamp_SyncTime is set to 0xFFFFFFFF FFFFFFFF (64 bits).
6. The Lower Tester requests clock synchronization on MCL-B with BluetoothClock_SyncTime to 0xFFFFFFFF. TimeStamp_SyncTime is set to 0xFFFFFFFF FFFFFFFF (64 bits).
7. Synchronization is performed for MCL-A and MCL-B.

- Expected Outcome

  Pass verdict

  The Time-Stamp Clocks are normalized to microseconds.

  The IUT does not perform piconet Role switch during the synchronization operation.

  TimeStamp_SyncTimes in the IUT responses match the parameter of the MD_SYNC_SET_REQs (with the TimeStamp_SampleAccuracy parameter taken in account).

4.5.4 Clock Synchronization Error Handling

Verify that support for error handling has been properly implemented. This includes situations where a response code is received by an IUT and situations that result in the generation of a response code.

MCAP/SRC-SNK/CS-ERR/BI-01-C [IUT receives unsupported standard op code command and sends Response Code]

- Test Purpose

  Verify proper response when an IUT that does not support Standard Op Codes receives an unsupported standard op code command.

- Reference

  [3] 4.1.3.2/4, 4.1.3.4/10, 5.0/1

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel connection has been established.

- Test Procedure

  1. The Lower Tester sends standard op codes (Create, Reconnect, and Abort) to an IUT that does not support standard op codes.

- Expected Outcome

  Pass verdict

  The IUT sends a “Request Not Supported” response code for each unsupported op code.

  The op code field of the Response Packet is set to the Response Op Code that corresponds to the unsupported Request Op Code.

  The IUT returns to stable state and can process commands normally.

MCAP/SS/CS-ERR/BI-02-C [IUT detects invalid Bluetooth clock setting (Invalid slot number)]

- Test Purpose

  Verify proper response when an invalid Bluetooth clock setting is detected by the IUT (Invalid slot number).

- Reference

  [3] 5.2.3.1, 5.2.3.2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established.

- Test Procedure

  1. The Lower Tester initiates clock synchronization with the IUT with a Bluetooth clock set to an invalid Bluetooth Clock time slot number (i.e., any value between 0x10000000 to i.e., 0xFFFFFFFE).
  2. The IUT sends an “Invalid Parameter Value” response code.

- Expected Outcome

  Pass verdict

  TimeStamp_SyncTime is set to a value other than 0xFFFFFFFF FFFFFFFF (64 bits).

  The IUT sends an MD_SYNC_SET_RSP with response code set to 0x02 (Invalid Parameter Value).

  Response Parameters in the Response Packet (bytes 3–16) are set to zero.

  The IUT returns to a stable state and can process commands normally.

MCAP/SS/CS-ERR/BI-03-C [IUT detects invalid Bluetooth clock setting (slot number exceeds max value)]

- Test Purpose

  Verify a proper response when an invalid Bluetooth clock setting is detected by the IUT due to a Bluetooth Clock time slot number that exceeds the maximum value.

- Reference

  [3] 5.2.3.1, 5.2.3.2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established.

- Test Procedure

  1. The Lower Tester initiates clock synchronization with the IUT with a Bluetooth clock set to a valid slot number, but a slot number that the piconet Bluetooth clock will reach after more than 60 seconds in the future (i.e., the Baseband Half-Slot Instant indicated by BluetoothClock_SyncTime was more than SyncLeadTime plus 60 seconds in the future at the time the request was received).
  2. The IUT sends an “Invalid Parameter Value” response code.

- Expected Outcome

  Pass verdict

  TimeStamp_SyncTime is set to a value other than 0xFFFFFFFF FFFFFFFF (64 bits).

  The IUT sends an MD_SYNC_SET_RSP with response code set to 0x02 (Invalid Parameter Value).

  Response Parameters in the Response Packet (bytes 3–16) are set to zero.

  The IUT returns to a stable state and can process commands normally.

MCAP/SS/CS-ERR/BI-04-C [IUT detects invalid Bluetooth clock setting (slot number below min value)]

- Test Purpose

  Verify a proper response when an invalid Bluetooth clock setting is detected by the IUT due to a Bluetooth Clock time slot number that is below the minimum value.

- Reference

  [3] 5.2.3.1, 5.2.3.2

- Initial Condition

  - As defined in Section 2.2. Additionally, a Control Channel has been established.

- Test Procedure

  1. The Lower Tester initiates clock synchronization with the IUT with a Bluetooth clock set to a valid slot number, but a slot number that the piconet Bluetooth clock will reach to before the indicated synchronization preparation time in the future (i.e., the Baseband Half-Slot Instant indicated by BluetoothClock_SyncTime was less than SyncLeadTime in the future at the time the request was received).
  2. The IUT sends an “Invalid Parameter” response code.

- Expected Outcome

  Pass verdict

  TimeStamp_SyncTime is set to a value other than 0xFFFFFFFF FFFFFFFF (64 bits).

  The IUT sends a MD_SYNC_SET_RSP with response code set to 0x02 (Invalid Parameter Value).

  Response Parameters in the Response Packet (bytes 3–16) are set to zero.

  The IUT returns to a stable state and can process commands normally.

5 Test case mapping

The Test Case Mapping Table (TCMT) maps test cases to specific requirements in the ICS. The IUT is tested in all roles for which support is declared in the ICS document.

The columns for the TCMT are defined as follows:

Item: Contains a logical expression based on specific entries from the associated ICS document. Contains a logical expression (using the operators AND, OR, NOT as needed) based on specific entries from the applicable ICS document(s). The entries are in the form of y/x references, where y corresponds to the table number and x corresponds to the feature number as defined in the ICS document for Multi-Channel Adaptation Protocol (MCAP) [4].

Feature: A brief, informal description of the feature being tested.

Test Case(s): The applicable test case identifiers are required for Bluetooth Qualification if the corresponding y/x references defined in the Item column are supported. Further details about the function of the TCMT are elaborated in [2].

For the purpose and structure of the ICS/IXIT, refer to [2].

Item	Feature	Test Case(s)
MCAP 3/2 OR MCAP 5/2	Control and Data Channel Establishment	MCAP/SRC-SNK/CE/BV-01-C
MCAP 3/3 OR MCAP 5/3	Control and Data Channel Establishment	MCAP/SRC-SNK/CE/BV-02-C
(MCAP 3/2 AND MCAP 3/3) OR (MCAP 5/2 AND MCAP 5/3)	Control and Data Channel Establishment	MCAP/SRC-SNK/CE/BV-03-C
(MCAP 3/2 AND MCAP 3/3) OR (MCAP 5/2 AND MCAP 5/3)	Control and Data Channel Establishment	MCAP/SRC-SNK/CE/BV-04-C
MCAP 3/4 OR MCAP 5/4	Connection Establishment	MCAP/SRC-SNK/CM-DIS/BV-01-C
MCAP 3/5 OR MCAP 5/5	Connection Establishment	MCAP/SRC-SNK/CM-DIS/BV-02-C
MCAP 3/2 OR MCAP 5/2	Connection Establishment	MCAP/SRC-SNK/CM-DIS/BV-03-C
MCAP 3/3 OR MCAP 5/3	Connection Establishment	MCAP/SRC-SNK/CM-DIS/BV-04-C
(MCAP 3/2 AND MCAP 3/3) OR (MCAP 5/2 AND MCAP 5/3)	Connection Establishment	MCAP/SRC-SNK/CM-DIS/BV-05-C
(MCAP 3/2 AND MCAP 3/3) OR (MCAP 5/2 AND MCAP 5/3)	Connection Establishment	MCAP/SRC-SNK/CM-REC/BV-01-C
MCAP 3/4 OR MCAP 5/4	MCL Disconnection	MCAP/SRC-SNK/CM-REC/BV-02-C
MCAP 3/5 OR MCAP 5/5	MCL Disconnection	MCAP/SRC-SNK/CM-REC/BV-03-C
MCAP 3/5 OR MCAP 5/5	MCL Disconnection	MCAP/SRC-SNK/CM-REC/BV-04-C
MCAP 3/6 OR MCAP 5/6	MDL Disconnection	MCAP/SRC-SNK/CM-REC/BV-05-C
MCAP 3/7 OR MCAP 5/7	MDL Disconnection	MCAP/SRC-SNK/CM-REC/BV-06-C
(MCAP 3/4 AND MCAP 3/8) OR (MCAP 5/4 AND MCAP 5/8)	MDL Reconnection	MCAP/SRC-SNK/CM-DEL/BV-01-C
MCAP 3/11 OR MCAP 5/11	MDL Delete	MCAP/SRC-SNK/CM-DEL/BV-02-C

Item	Feature	Test Case(s)
MCAP 3/12 OR MCAP 5/12	MDL Delete	MCAP/SRC-SNK/CM-DEL/BV-03-C
MCAP 3/13 OR MCAP 5/13	MDL Delete	MCAP/SRC-SNK/CM-DEL/BV-04-C
(MCAP 3/3 AND MCAP 3/14) OR (MCAP 5/3 AND MCAP 5/14)	MDL Abort	MCAP/SRC-SNK/CM-ABT/BV-01-C
(MCAP 3/3 AND MCAP 3/15) OR (MCAP 5/3 AND MCAP 5/15)	MDL Abort	MCAP/SRC-SNK/CM-ABT/BV-02-C
(MCAP 3/3 AND MCAP 3/14) OR (MCAP 5/3 AND MCAP 5/14)	MDL Abort	MCAP/SRC-SNK/CM-ABT/BV-03-C
MCAP 1/1 OR MCAP 1/2	Error Handling - Invalid Op Code/ Invalid Parameter Value/	MCAP/SRC-SNK/ERR/BI-01-C MCAP/SRC-SNK/ERR/BI-02-C MCAP/SRC-SNK/ERR/BI-03-C MCAP/SRC-SNK/ERR/BI-04-C
MCAP 3/3 OR MCAP 5/3	Error Handling - Invalid MDEP	MCAP/SRC-SNK/ERR/BI-05-C
MCAP 3/2 OR MCAP 5/2	Error Handling - Invalid MDEP	MCAP/SRC-SNK/ERR/BI-06-C
MCAP 3/3 OR MCAP 5/3	Error Handling - IUT Busy	MCAP/SRC-SNK/ERR/BI-07-C
MCAP 3/2 OR MCAP 5/2	Error Handling - MDEP Busy	MCAP/SRC-SNK/ERR/BI-08-C
MCAP 3/3 OR MCAP 5/3	Error Handling - Invalid MDL	MCAP/SRC-SNK/ERR/BI-09-C
MCAP 3/9 OR MCAP 5/9	Error Handling - Invalid MDL	MCAP/SRC-SNK/ERR/BI-10-C
MCAP 3/2 OR MCAP 5/2	Error Handling - Invalid MDL	MCAP/SRC-SNK/ERR/BI-11-C
MCAP 3/2 OR MCAP 5/2	Error Handling - MDL Busy	MCAP/SRC-SNK/ERR/BI-12-C
MCAP 3/2 OR MCAP 5/2	Error Handling - Invalid Operation	MCAP/SRC-SNK/ERR/BI-13-C
MCAP 3/2 OR MCAP 5/2	Error Handling - Invalid Operation	MCAP/SRC-SNK/ERR/BI-14-C
MCAP 3/2 OR MCAP 5/2	Error Handling - Invalid Operation	MCAP/SRC-SNK/ERR/BI-15-C
MCAP 1/1 OR MCAP 1/2	Error Handling - Resource Unavailable	MCAP/SRC-SNK/ERR/BI-16-C
MCAP 3/2 OR MCAP 5/2	Error Handling - Resource Unavailable	MCAP/SRC-SNK/ERR/BI-17-C
MCAP 3/2 OR MCAP 5/2	Error Handling - Unspecified Error	MCAP/SRC-SNK/ERR/BI-18-C
MCAP 1a/1 AND (NOT MCAP 1a/2)	Error Handling - Request not Supported	MCAP/SRC-SNK/ERR/BI-19-C
MCAP 1a/2 OR MCAP 1/4	Error Handling - Request not Supported	MCAP/SRC-SNK/ERR/BI-20-C
MCAP 1/1 OR MCAP 1/2	Control Channel Request and Response rules	MCAP/SRC-SNK/INV/BI-01-C
MCAP 3/3 OR MCAP 5/3	Control Channel Request and Response rules	MCAP/SRC-SNK/INV/BI-02-C

Item	Feature	Test Case(s)
MCAP 3/2 OR MCAP 5/2	Control Channel Request and Response rules	MCAP/SRC-SNK/INV/BI-03-C
MCAP 1/1 OR MCAP 1/2	Control Channel Request and Response rules	MCAP/SRC-SNK/INV/BI-04-C
MCAP 3/3 OR MCAP 5/3	Control Channel Request and Response rules	MCAP/SRC-SNK/INV/BI-05-C
MCAP 3/3 OR MCAP 5/3	Control Channel Request and Response rules	MCAP/SRC-SNK/INV/BI-06-C
MCAP 3/3 OR MCAP 5/3	Control Channel Request and Response rules	MCAP/SRC-SNK/INV/BI-07-C
MCAP 7/2 AND MCAP 7/4	Clock Synchronization - IUT as Sync-Master	MCAP/SM/CS-I/BV-01-C
MCAP 7/2	Clock Synchronization - IUT as Sync-Master	MCAP/SM/CS-I/BV-02-C
MCAP 6/1 AND MCAP 7/2 AND MCAP 7/4	Clock Synchronization - IUT as Sync-Master	MCAP/SM/CS-I/BV-03-C
MCAP 7/2 AND MCAP 7/4	Clock Synchronization - IUT as Sync-Master	MCAP/SM/CS-I/BV-04-C
MCAP 6/1 AND MCAP 6/4	Clock Synchronization - IUT as Sync-Slave	MCAP/SS/CS-R/BV-01-C MCAP/SS/CS-R/BV-03-C
MCAP 6/1	Clock Synchronization - IUT as Sync-Slave	MCAP/SS/CS-R/BV-02-C
MCAP 1a/1 AND MCAP 6/1	Clock Synchronization - IUT as Sync-Slave	MCAP/SS/CS-R/BV-04-C
MCAP 6/1 AND MCAP 6/3 AND MCAP 6/4	Clock Synchronization - Multi-MCL Handling	MCAP/SRC-SNK/CS-M/BV-01-C
MCAP 1a/2 AND NOT MCAP 1a/1	Clock Synchronization - Error Handling	MCAP/SRC-SNK/CS-ERR/BI-01-C
MCAP 6/1 AND MCAP 6/4	Clock Synchronization - Error Handling	MCAP/SS/CS-ERR/BI-02-C MCAP/SS/CS-ERR/BI-03-C
MCAP 6/1	Clock Synchronization - Error Handling	MCAP/SS/CS-ERR/BI-04-C

Table 5.1: Test case mapping

6 Revision history and acknowledgments

Revision History

Publication Number	Revision Number	Date	Comments
0	1.0.0	2008-06-25	Prepare for publication.
	1.0.1r0	2008-09-17	TSE 2655: TP/MCAP/CM-DIS/BV-05-C, Change Pass verdict TSE 2671: TP/MCAP/CM-ABT/BV-01-C: update test case
1	1.0.1	2008-12-03	Prepare for publication.
	1.0.2r0	2009-05-01	TSE 2755: TP/MCAP/CS-I/BV-02-I Pass verdict change TSE 2760 TP/MCAP/CM-DEL/BV-01-C, TP/MCAP/CM-DEL/BV-03-C: update test procedures
2	1.0.2	2009-08-10	Prepare for publication.
	1.0.3r0	2011-01-06	TSE 2848: TP/MCAP/ERR/BI-03-C, TP/MCAP/ERR/BI-05-C, TP/MCAP/ERR/BI-09-C, TP/MCAP/ERR/BI-14-C, TP/MCAP/ERR/BI-16-C TE 3013: TP/MCAP/CM-REC/BV-01-C, TP/MCAP/CM-REC/BV-02-C, TP/MCAP/CM-REC/BV-03-C, TP/MCAP/CM-REC/BV-04-C, TP/MCAP/CM-REC/BV-05-C, TP/MCAP/CM-REC/BV-06-C TSE 3019: TP/MCAP/ERR/BI-19-C: Pass verdict TSE 3387: TP/MCAP/ERR/BI-20-C: change TCMT
3	1.0.3	2011-07-21	Prepare for publication.
	1.0.4r00	2017-02-07	Updated Test Spec Template.
	1.0.4r02	2017-06-01	TSE 8571: Test Spec template update. Updated instances of Test System to Lower and Upper Tester as appropriate.
	1.0.4r03	2017-06-14	Converted to new Test Case ID conventions as defined in TSTO v4.1
4	1.0.4	2017-07-03	Approved by BTI. Prepared for TCRL 2017-1 publication.
	p5r00	2023-10-25	TSE 23888 (rating 1): Converted -I tests to -C tests as appropriate; updated the TCMT and TCRL accordingly. Updated to align the document with the latest standards.
5	p5	2024-07-01	Approved by BTI on 2024-05-22. Prepared for TCRL 2024-1 publication.

Acknowledgments

Name	Company
Jerry Wang	A & D Medical
Hermanni Suominen	Bluegiga
Alicia Courtney	Broadcom
Mats Andersson	connectBlue

Name	Company
Peter Flittner	CSR
Ulf Karlsson	Cybercom
Nick Hunn	Ezurio
Tomas O'Raghallaigh	Frontline Test Equipment
Michael Nidd	IBM
Robert D. Hughes	Intel Corporation (editor)
Krishna Shingala	MindTree
Jayant Parthasarathy	Nonin Medical
Andrea Cucchi	ST Microelectronics
Karsten Aalders	Stollmann