Reconnection Configuration Service (RCS)

Bluetooth® **Test Suite**

- ▪ **Revision:** RCS.TS.p2
- ▪ **Revision Date:** 2022-01-25
- ▪ **Group Prepared By:** Medical Devices Working Group

Contents

1 Scope

This Bluetooth document contains the Test Suite Structure (TSS) and test cases to test the implementation of the Bluetooth Reconnection Configuration Service 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, by dated or undated reference, provisions from other publications. These references are cited at the appropriate places in the text and the publications are listed hereinafter.

[1] Bluetooth Core Specification, Version 4.0 or later

[2] Test Strategy and Terminology Overview

[3] Reconnection Configuration Service, Version 1.0 or later

[4] ICS Proforma for Reconnection Configuration Service

[5] GATT Test Suite, GATT.TS

[6] Reconnection Configuration Service Implementation eXtra Information for Test, IXIT

[7] Reconnection Configuration Service, Version 1.0.1

[8] Appropriate Language Mapping Tables document

2.2 Definitions

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

2.3 Acronyms and abbreviations

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

Certain terms that were identified as inappropriate have been replaced. For a list of the original terms and their replacement terms, see the Appropriate Language Mapping Tables document [8].

3 Test Suite Structure (TSS)

3.1 Test strategy

The test objectives are to verify functionality of the Reconnection Configuration Service within a Bluetooth Host and enable interoperability between Bluetooth Hosts on different devices.

The testing approach is to cover mandatory and optional requirements in the specification and to match 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.

3.2 Test groups

The following test groups have been defined:

- Generic GATT Integrated Tests

- Configure Notification and Indication

- Reconnection Configuration Features

- Reconnection Configuration Control Point procedures

- Reconnection Settings

- Limited Access Behavior

- End to End - Cyclic redundancy check

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>**.

Additionally, testing of this specification includes tests from the GATT Test Suite [5] referred to as Generic GATT Integrated Tests (GGIT); when used, the GGIT tests are referred to through a TCID string using the following convention:
<spec abbreviation>/<IUT role>/<GGIT test group>/< GGIT class >/<xx>-<nn>-<y>.

Identifier Abbreviation	Spec Identifier <spec abbreviation>
RCS	Reconnection Configuration Service
Identifier Abbreviation	**Role Identifier <IUT role>**
SR	Server Role
Identifier Abbreviation	**Reference Identifier <GGIT test group>**
SGGIT	Server Generic GATT Integrated Tests
Identifier Abbreviation	**Reference Identifier <GGIT class>**
CHA	Characteristic
ISFC	Indication Supported Features Characteristic
SER	Service
Identifier Abbreviation	**Feature and Behaviors Identifier <feat>**
CON	Configure Notification and Indication
RCFEA	Reconnection Configuration Features
RCCP	Reconnection Configuration Control Point
RCSET	Reconnection Settings
RCLAB	Limited Access Behavior
E2ECRC	End to End - Cyclic redundancy check

Table 4.1: RCS TC feature naming convention

4.1.2   Conformance

When conformance is claimed for a particular specification, all capabilities are to be supported in the specified manner (process-mandatory). 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 that is 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

- 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 Launch Studio, 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 errata request such that the issue may be addressed.

4.1.3 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, the outcome of the test is a Fail verdict.

4.2 Setup preambles

The procedures defined in this section are used to achieve specific conditions on the IUT and the test equipment within the tests defined in this document. The preambles here are commonly used to establish initial conditions.

4.2.1 ATT Bearer on LE transport

- Preamble Procedure

    1. Establish an LE transport connection between the IUT and the Lower Tester.
    2. Establish an L2CAP channel 0x0004 between the IUT and the Lower Tester over that LE transport.

4.2.2 Configure Server for Notifications and Indications

- Preamble Purpose

    This preamble procedure enables the Lower Tester to configure the Client Characteristic Configuration Descriptor (CCCD) of the RC Settings characteristic for notification and the CCCD of the RC Control Point characteristic for indication

- Preamble Procedure

    1. Establish an ATT Bearer connection between the Lower Tester and IUT as described in Section 4.2.1.
    2. The Lower Tester knows the handle of the RC Settings CCCD by executing RCS/SR/SGGIT/CHA/BV-04-C [Characteristic GGIT – RC Settings - Notify] or by other means.
    3. The Lower Tester knows the handle of the RC Control Point CCCD by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point] or by other means.

4. The Lower Tester executes the GATT Write Characteristic Value sub-procedure with the handle of the RC Settings Client Characteristic Configuration descriptor and the 16-bit attribute value of 0x0001.
5. The Lower Tester executes the GATT Write Characteristic Value sub-procedure with the handle of the RC Control Point Client Characteristic Configuration descriptor and the 16-bit attribute value of 0x0002.

4.3 Generic GATT Integrated Tests

Execute the Generic GATT Integrated Tests defined in Section 6.3, Server test procedures (SGGIT), in [5] using Table 4.2 below as input:

TCID	Service / Characteristic	Reference	Properties	Value Length (Octets)	Type
RCS/SR/SGGIT/SER/BV-01-C [Service GGIT – Reconnection Configuration]	Reconnection Configuration Service	[3] 2	-	-	Not defined
RCS/SR/SGGIT/CHA/BV-02-C [Characteristic GGIT – RC Feature]	RC Feature characteristic	[3] 3.1	0x02 (Read)	Skip	-
RCS/SR/SGGIT/CHA/BV-03-C [Characteristic GGIT – RC Settings]	RC Settings characteristic	[3] 3.2	0x02 (Read)	Skip	-
RCS/SR/SGGIT/CHA/BV-04-C [Characteristic GGIT – RC Settings - Notify]	RC Settings characteristic	[3] 3.2	0x12 (Read, Notify)	Skip	-
RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point]	Reconnection Configuration Control Point (RCCP) characteristic	[3] 3.3	0x28 (Write, Indicate)	Skip	-
RCS/SR/SGGIT/CHA/BV-06-C [Characteristic GGIT – RC Feature - Indicate]	RC Feature characteristic	[7] 3.1	0x22 (Read, Indicate)	Skip	-

Table 4.2: Input table for the GGIT Server tests

4.3.1 Generic GATT Indication Supported Features characteristic

Execute the Generic GATT Indication Supported Features Characteristic tests defined in Section 6.3, Server test procedures (SGGIT), in [5] using Table 4.3 below as input:

TCID	Characteristic	Reference	TC Configuration
RCS/SR/SGGIT/ISFC/BV-07-C [Characteristic GGIT – RC Feature indication]	RC Feature	[7] 3.1.1	N/A

Table 4.3: Input table for the GGIT Indication Supported Features Characteristic tests

4.4   Configure notification and indication

• Test Purpose

This test group verifies compliant operation in response to enable and disable characteristic indication or notification, and reading of the client characteristic configuration descriptor value.

• Reference

[3] 3

• Initial Condition

- The handle of each characteristic value referenced in the Test Cases below has been previously discovered by the Lower Tester during the test procedure in Section 4.3 or is known to the Lower Tester by other means.

- The handle of the client characteristic configuration descriptor of each characteristic referenced in the Test Cases below has been previously discovered by the Lower Tester during the test procedure in Section 4.3 or is known to the Lower Tester by other means.

- Establish an ATT Bearer connection between the Lower Tester and the IUT as described in Section 4.2.1.

• Test Case Configuration

Test Case	Value (Requirements)
RCS/SR/CON/BV-01-C [Configure Notification – RC Settings]	0x0001 (Section 3.2 in [3])
RCS/SR/CON/BV-02-C [Configure Indication – Reconnection Configuration Control Point]	0x0002 (Section 3.3 in [3])

Table 4.4: Configure indication and notification test cases

• Test Procedure

The following test procedure applies to the Test Cases listed in Table 4.4:

1. The Lower Tester sends an *ATT_Write_Request*, to disable indication or notification, by writing value 0x0000 to the client characteristic configuration descriptor of the characteristic.
2. The Lower Tester reads the value of the client characteristic configuration descriptor.
3. If the Test Case is for notification, enable notification by writing value 0x0001 to the client characteristic configuration descriptor of the characteristic. If the Test Case is for indication, enable indication by writing value 0x0002 to the client characteristic configuration descriptor of the characteristic.
4. The Lower Tester reads the value of the client characteristic configuration descriptor.

• Expected Outcome

Pass verdict

The characteristic descriptor is successfully written, and the value returned when read is consistent with the value written.

4.5   Reconnection Configuration Features

- Test Purpose

    This test group contains one or more test cases to read and verify that the indicated features are reflected in the value of the RC Feature characteristic.

- Reference

    [3] 3.1.1

- Initial Condition

    - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

    - The Lower Tester knows the handle of the RC Feature characteristic value declaration by executing RCS/SR/SGGIT/CHA/BV-02-C [Characteristic GGIT – RC Feature]. or RCS/SR/SGGIT/CHA/BV-06-C [Characteristic GGIT – RC Feature - Indicate].

- Test Case Configuration

Test Case	Bit position	Required Bit Value
RCS/SR/RCFEA/BV-01-C [E2E-CRC Supported]	0	1
RCS/SR/RCFEA/BV-02-C [E2E-CRC not Supported]	0	0
RCS/SR/RCFEA/BV-03-C [Enable Disconnect Supported]	1	1
RCS/SR/RCFEA/BV-04-C [Enable Disconnect not Supported]	1	0
RCS/SR/RCFEA/BV-05-C [Ready for Disconnect Supported]	2	1
RCS/SR/RCFEA/BV-06-C [Ready for Disconnect not Supported]	2	0
RCS/SR/RCFEA/BV-07-C [Propose Reconnection Timeout Supported]	3	1
RCS/SR/RCFEA/BV-08-C [Propose Reconnection Timeout not Supported]	3	0
RCS/SR/RCFEA/BV-09-C [Propose Connection Interval Supported]	4	1
RCS/SR/RCFEA/BV-10-C [Propose Connection Interval not Supported]	4	0
RCS/SR/RCFEA/BV-11-C [Propose Peripheral Latency Supported]	5	1
RCS/SR/RCFEA/BV-12-C [Propose Peripheral Latency not Supported]	5	0
RCS/SR/RCFEA/BV-13-C [Propose Supervision Timeout Supported]	6	1
RCS/SR/RCFEA/BV-14-C [Propose Supervision Timeout not Supported]	6	0
RCS/SR/RCFEA/BV-15-C [Propose Advertisement Interval Supported]	7	1
RCS/SR/RCFEA/BV-16-C [Propose Advertisement Interval not Supported]	7	0
RCS/SR/RCFEA/BV-17-C [Propose Advertisement Count Supported]	8	1
RCS/SR/RCFEA/BV-18-C [Propose Advertisement Count not Supported]	8	0
RCS/SR/RCFEA/BV-19-C [Propose Advertisement Repetition Supported]	9	1
RCS/SR/RCFEA/BV-20-C [Propose Advertisement Repetition not Supported]	9	0
RCS/SR/RCFEA/BV-21-C [Advertisement Configuration 1 Supported]	10	1
RCS/SR/RCFEA/BV-22-C [Advertisement Configuration 1 not Supported]	10	0
RCS/SR/RCFEA/BV-23-C [Advertisement Configuration 2 Supported]	11	1
RCS/SR/RCFEA/BV-24-C [Advertisement Configuration 2 not Supported]	11	0
RCS/SR/RCFEA/BV-25-C [Advertisement Configuration 3 Supported]	12	1
RCS/SR/RCFEA/BV-26-C [Advertisement Configuration 3 not Supported]	12	0
RCS/SR/RCFEA/BV-27-C [Advertisement Configuration 4 Supported]	13	1

Test Case	Bit position	Required Bit Value
RCS/SR/RCFEA/BV-28-C [Advertisement Configuration 4 not Supported]	13	0
RCS/SR/RCFEA/BV-29-C [Upgrade to LESC Only Supported]	14	1
RCS/SR/RCFEA/BV-30-C [Upgrade to LESC Only not Supported]	14	0
RCS/SR/RCFEA/BV-31-C [Next Pairing OOB Supported]	15	1
RCS/SR/RCFEA/BV-32-C [Next Pairing OOB not Supported]	15	0
RCS/SR/RCFEA/BV-33-C [Use of Filter Accept List Supported]	16	1
RCS/SR/RCFEA/BV-34-C [Use of Filter Accept List not Supported]	16	0
RCS/SR/RCFEA/BV-35-C [Disable RCS (for other Clients) Supported]	17	1
RCS/SR/RCFEA/BV-36-C [Disable RCS (for other Clients) not Supported]	17	0
RCS/SR/RCFEA/BV-37-C [RFUs]	18-22	0
RCS/SR/RCFEA/BV-38-C [Feature Extension]	23	0

Table 4.5: RC Feature test cases

- Test Procedure

    1. The Lower Tester executes the GATT Read Characteristic Value sub-procedure to read the RC Feature characteristic value from the IUT, using the given handle.
    2. For every test case to be executed, the required bit value is checked against the requirements.

- Expected Outcome

    <u>Pass verdict</u>

    The required bit value must be equal to the bit value at the given bit position in the RC Feature characteristic value.

4.6   Reconnection Configuration Control Point Procedures

RCS/SR/RCCP/BV-01-C [Enable Disconnect Procedure]

- Test Purpose

    Verify that the IUT can execute the Enable Disconnect procedure.

- Reference

    [3] 3.3.1, 3.3.2, 3.3.2.1

- Initial Condition

    - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

    - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

    - If the IUT is not ready to disconnect, do whatever is necessary to make the IUT ready for disconnecting.

    - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

    - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

    1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to one octet containing the «Enable Disconnect» opcode.
    2. The RC server IUT responds with an *ATT_Write_Response*.
    3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to three octets containing the «Procedure Response» opcode, followed by the «Enable Disconnect» opcode and the result code value 0x01 (Success).
    4. The Lower Tester sends an *ATT_Handle_Value_Confirmation* PDU.

- Expected Outcome

    Pass verdict

    The test procedure executes in the given order without any errors and PDUs contain the described values.

    If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

RCS/SR/RCCP/BI-01-C [Enable Disconnect Procedure, Type 2]

- Test Purpose

    Verify that the IUT responds with a Device Busy error code when the Enable Disconnect procedure is executed and the IUT cannot start the disconnect procedure.

- Reference

    [3] 3.3.1, 3.3.2, 3.3.2.1

- Initial Condition

    - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

    - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

    - If the IUT is ready to disconnect, do whatever is necessary to make the IUT become **not** ready for disconnecting.

    - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

    - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

    1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to one octet containing the «Enable Disconnect» opcode.
    2. The RC server IUT responds with an *ATT_Write_Response*.
    3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to three octets

containing the «Procedure Response» opcode, followed by the «Enable Disconnect» opcode and the result code value 0x07 (Device Busy).

4. The Lower Tester sends an *ATT_Handle_Value_Confirmation* PDU.

- Expected Outcome

Pass verdict

The test procedure executes in the given order without any errors and PDUs contain the described values.

If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

RCS/SR/RCCP/BV-02-C [Get Actual Communication Parameters]

- Test Purpose

Verify that the IUT can execute the Get Actual Communication Parameters procedure.

- Reference

[3] 3.3.1, 3.3.2, 3.3.2.2

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

  - The connection interval, the connection client latency, and the connection timeout of the current connection between the IUT and Lower Tester are known, either by explicitly establishing the connection with given parameters or by monitoring the connection process and connection parameter update procedures.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to one octet containing the «Get Actual Communication Parameters» opcode.
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Client Parameter Indication» opcode, followed by the actual used Communication Parameters.
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation* PDU.

- Expected Outcome

Pass verdict

The test procedure executes in the given order without any errors and PDUs contain the described values.

The received Response Parameters must contain the currently used communication parameters.

If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

RCS/SR/RCCP/BV-03-C [Propose Reconnection Timeout Settings]

- Test Purpose

  Verify that the IUT can execute the Propose Settings procedure with changed reconnection timeout settings.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.3

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - Change the reconnection timeout parameter of the RC server to a value that is easily distinguishable from the value from the stored parameter-set 0.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

  - The parameters of the stored parameter-set 0 are known from the IXIT [6].

- Test Procedure

  The test procedure is executed twice with two different values for the reconnection timeout. Choose two values that are clearly distinguishable.

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Propose Settings» and an Operand Parameter with the chosen reconnection timeout. All other fields in the Operand Parameter are filled with the value 0xFFFF.
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Propose Settings» request opcode and the Result Code value of 0x01 (Success).
  4. Disconnect the Lower Tester from the IUT and let the IUT start to advertise.
  5. The IUT reconnection timeout timer starts.
  6. Wait until the reconnection timeout time has been reached.
  7. Reestablish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.
  8. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to one octet containing the «Get Actual Communication Parameters» opcode.
  9. The RC server IUT responds with an *ATT_Write_Response*.

10. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Client Parameter Indication» opcode, followed by the actual used Communication Parameters.
11. The Lower Tester sends an *ATT_Handle_Value_Confirmation* PDU.

- Expected Outcome

  Pass verdict

  The reconnection timer parameter that is used after the timeout has been reached is changed to the value from the known, stored parameters-set 0.

  If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

- Notes

  The test objective is to verify the behavior, not the exact timing, so a tolerance of ±5s is acceptable.

RCS/SR/RCCP/BV-04-C [Disable Reconnection Timer]

- Test Purpose

  Verify that the IUT can execute the Propose Settings procedure with a reconnection timeout of 0 to disable the Reconnection Timer.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.3

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - Change all advertising parameters that are supported by the RC server to values that are easily distinguishable from the value from the stored parameter-set 0 by observing the advertising behavior of the IUT.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

  - The advertising parameters of the stored parameter-set 0 are known.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Propose Settings» and an Operand Parameter with the reconnection timeout set to 0. All other fields in the Operand Parameter are filled with the value 0xFFFF.
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure

Response» opcode and followed by the «Propose Settings» request opcode and the Result Code value of 0x01 (Success).

4. Disconnect the Lower Tester from the IUT and let the IUT start to advertise.
5. The IUT sends advertising PDUs with the parameters from the stored parameter-set 0.

- Expected Outcome

Pass verdict

Despite the changed advertising parameters in the initial condition, the IUT advertises with the values from the known, stored parameters-set 0 for those parameters that are supported by the IUT.

If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

4.6.1 Propose Connection Parameter Settings

- Test Purpose

This test group verifies that the IUT can change specific parameters of the current connection by executing the Propose Settings procedure.

- Reference

[3] 3.3.1, 3.3.2, 3.3.2.3

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

  - The connection interval, the peripheral latency of the current connection between the IUT and Lower Tester is known, either by explicitly establishing the connection with given parameters or by monitoring the connection process and connection parameter update procedures.

- Test Case Configuration

Test Case	Operand Fields to Fill	Value to Fill
RCS/SR/RCCP/BV-05-C [Propose Connection Interval Settings]	Minimum Connection Interval and Maximum Connection Interval (Field-Nr. 1 and 2)	If the currently used connection interval between the IUT and Lower Tester is lower than or equal to 2 seconds, use 2.5 seconds as minimum connection interval and 4 seconds as maximum connection interval. If the currently used connection interval is larger than 2 seconds, use 1.5 seconds as maximum connection interval and 7.5ms as minimum connection interval.

Test Case	Operand Fields to Fill	Value to Fill
RCS/SR/RCCP/BV-06-C [Propose Peripheral Latency Settings]	Peripheral Latency (Field-Nr. 3)	Chose a new peripheral latency that is different from the peripheral latency of the current connection and is in the range of 0 to 499.
RCS/SR/RCCP/BV-07-C [Propose Supervision Timeout Settings]	Supervision Timeout (Field-Nr 4)	Chose a new supervision timeout that is different from the current connection and is in the range of 100ms to 32s.
RCS/SR/RCCP/BV-08-C [Propose Connection Parameter Settings]	From the test cases above, all fields for which the IUT indicates support.	From the test cases above, all values for which the IUT indicates support.

Table 4.6: Propose Connection Parameters Settings test cases

- Test Procedure

    1. Choose operand parameter values as described in Table 4.6. If the chosen parameters are out of the supported range of the IUT, choose the nearest values that are within the supported range. The core specification required that the connection supervision timeout be larger than (1 + connPeripheralLatency) * connInterval *2. If this requirement is violated by the chosen parameter, change another supported parameter so that the requirement will be fulfilled. If the change of other parameters is not supported, apply a minimum chosen parameter, so that the requirement will be fulfilled.
    2. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Propose Settings» and an Operand Parameter with Parameters as described by Table 4.6, for the specific test case. All other fields in the Operand Parameter are filled with the value 0xFFFF.
    3. The RC server IUT responds with an *ATT_Write_Response*.
    4. The IUT initiates the GAP Connection Parameter Update Procedure with the chosen parameters.
    5. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Propose Settings» request opcode and the Result Code value of 0x09 (Proposal Accepted).
    6. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.
    7. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Client Parameter Indication» opcode, followed by the actual used Communication Parameters.
    8. The Lower Tester sends an *ATT_Handle_Value_Confirmation* PDU.
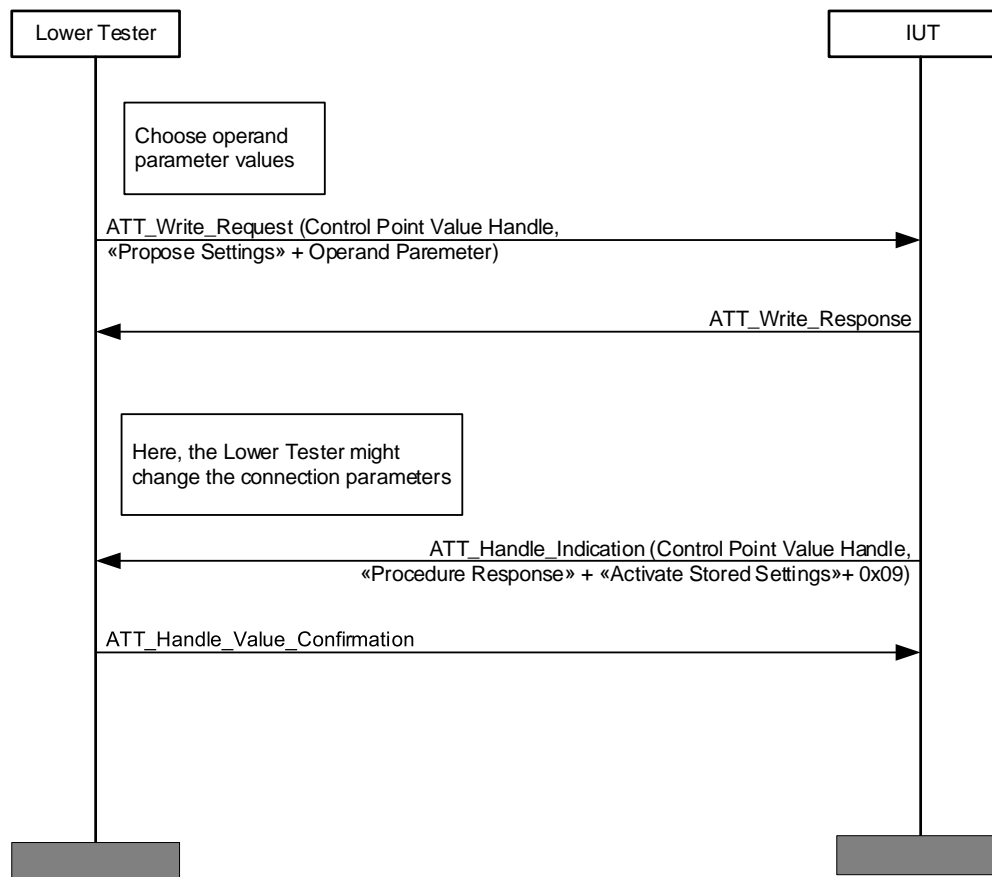
Figure 4.1: Message sequence chart for Propose Connection Parameter Settings

- Expected Outcome

  <u>Pass verdict</u>

  The IUT sends the requested PDUs in the given order.

  The IUT initiates the GAP Connection Parameter Update Procedure.

  The GAP Connection Parameter Update request, contains the changed parameter as described for the test case and for those parameters that are not changed, the currently used connection parameters.

  If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDUs must be present.

- Notes

  Depending on the IUT's Link Layer role, the set of implemented features in both link layers, and their respective core versions, the initiation of the connection parameter change might be done by sending a L2CAP Connection Parameter Update Request, a LL Connection Update Request, or a LL Connection Parameter Request.

4.6.2 Propose Advertising Parameter Settings

- Test Purpose

  This test group verifies that the IUT can change specific advertising parameters by executing the Propose Settings procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.3

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - Configure the IUT to use the stored parameter-set 0.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

  - The advertising parameters of the stored parameter-set 0 are known from the IXIT [6].

- Test Case Configuration

Test Case	Operand Fields to Fill	Value to Fill
RCS/SR/RCCP/BV-09-C [Propose Advertisement Interval Settings]	Advertisement Interval (Field-Nr. 5)	Chose a new advertisement interval that is different from that from the stored parameter-set 0 and in the range 20ms to 10.24s.
RCS/SR/RCCP/BV-10-C [Propose Advertisement Count Settings]	Advertisement Count (Field-Nr. 6)	Chose a new advertisement count value that is different from that from the stored parameter-set 0 and in the range 1 to 1000.
RCS/SR/RCCP/BV-11-C [Propose Advertisement Repetition Time Settings]	Advertisement Repetition (Field-Nr 7)	Chose a new advertisement repetition value that is different from that from the stored parameter-set 0 and in the range 1 to 10000.
RCS/SR/RCCP/BV-12-C [Propose Advertisement Parameter Settings]	From the test cases above, all fields for which the IUT indicates support.	From the test cases above, all values for which the IUT indicates support.

Table 4.7: Propose Advertising Parameters Settings test cases

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Propose Settings» and an Operand Parameter with Parameters as described by the Table 4.7, for the specific test case. All other fields in the Operand Parameter are filled with the value 0xFFFF.
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure

Response» opcode and followed by the «Propose Settings» request opcode and the Result Code value of 0x01 (Success).

4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.
5. Disconnect the Lower Tester from the IUT and do whatever is necessary for the IUT to start advertising.
6. The IUT sends advertising PDUs.
7. The Lower Tester observes the PDUs send by the IUT.

Figure 4.2: Message sequence chart for Propose Advertising Parameter Settings

- Expected Outcome

Pass verdict

The IUT sends the requested PDUs in the given order.

If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDUs must be present.

The observed advertising PDUs follow the changed advertising parameters for those parameters that were changed by the test.

The observed advertising PDUs follow parameters from the stored parameter-set 0 for those parameters that were not to be changed by the test.

RCS/SR/RCCP/BI-02-C [Propose Invalid Connection Interval]

- Test Purpose

  Verify that the IUT tests the connection interval part of the parameters of a Propose Settings procedure request.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.3

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

  - The connection parameters, acceptable by the IUT, are known.

- Test Procedure

  Choose a maximum and minimum connection interval parameter that are known to be not acceptable by the IUT.

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Propose Settings» and an Operand Parameter with the Minimum Connection Interval set to the chosen maximum connection interval and the Maximum Connection Interval set to the chosen minimum connection interval. All other fields in the Operand Parameter are filled with the value 0xFFFF.
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode, followed by the «Propose Settings» request opcode, the Result Code value of 0x05 (Communication Parameter out of range) and the Result Parameter 0x06.
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDUs in the given order.

  If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDUs must be present.

4.6.3 Activate Stored Settings

- Test Purpose

  This test group verifies that the IUT can execute the Activate Stored Settings procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.4

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

  - The values of the stored parameter-sets are known from the IXIT [6].

- Test Case Configuration

Test Case	Parameter-Set to be used
RCS/SR/RCCP/BV-13-C [Activate Stored Parameter-Set 0 Settings]	Use parameter-set 0.
RCS/SR/RCCP/BV-14-C [Activate Stored Setting]	Use parameter-set 1.

Table 4.8: Activate Stored Settings test cases

- Test Procedure

  For every test case, there is exactly one parameter-set to use. The parameter to the activate stored settings procedure is the number of the parameter-set.

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Activate Stored Settings» and an operand octet with the parameter-set number.
  2. The RC server IUT responds with an *ATT_Write_Response*.

  Depending on the differences between the actual communication parameters and the values of the stored parameters-set 0, the test has two different outcomes:

  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Activate Stored Settings» request opcode and the Result Code value of 0x01 (Success).

  Or

  4. The IUT starts a GAP Connection Parameter Update Procedure, with the connection parameters set to the values of the used parameter-set.

5. The Lower Tester successfully executes the GAP Connection Parameter Update Procedure with the IUT, resulting in the changed connection parameters applied to the current link layer connection.
6. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Activate Stored Settings» request opcode and the Result Code value of 0x09 (Proposal Accepted).

In both cases, these are the next steps:

7. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.
8. Disconnect the Lower Tester from the IUT and do whatever is necessary for the IUT to start advertising.
9. The IUT sends advertising PDUs.
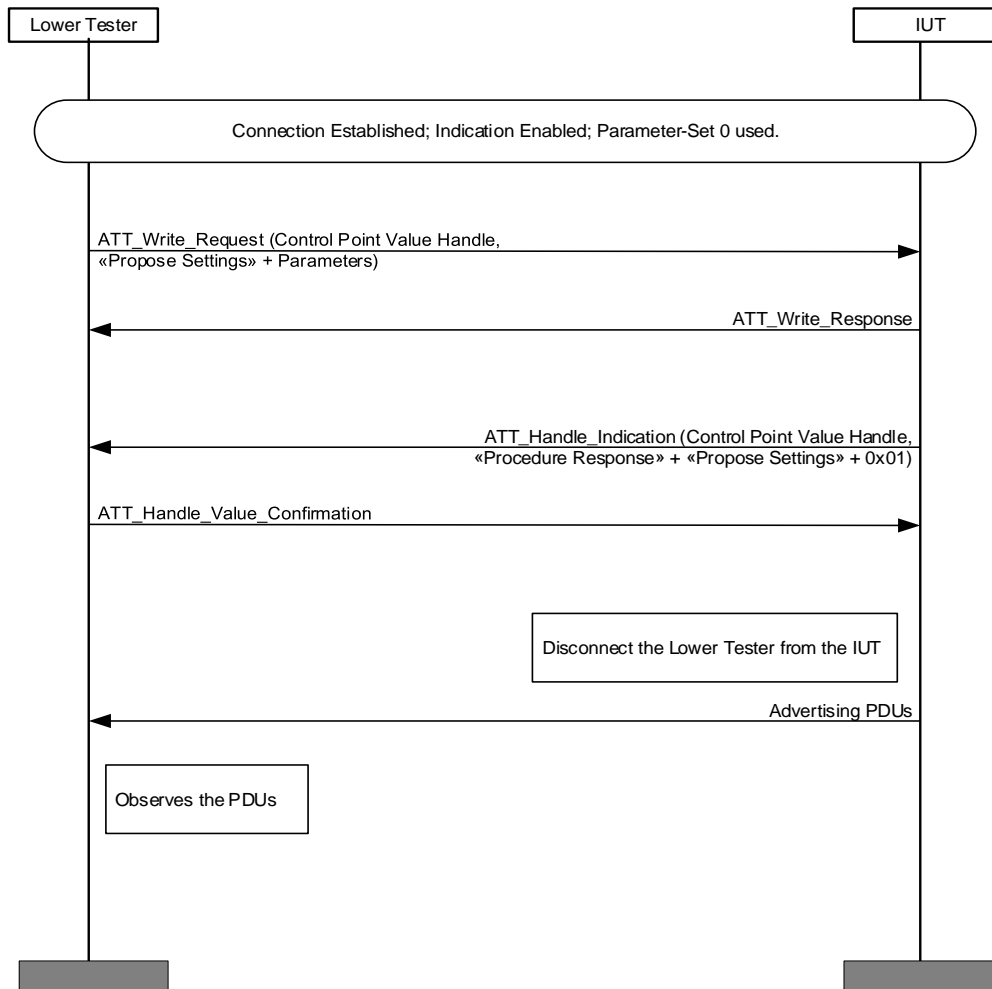10. The Lower Tester observes the PDUs send by the IUT.
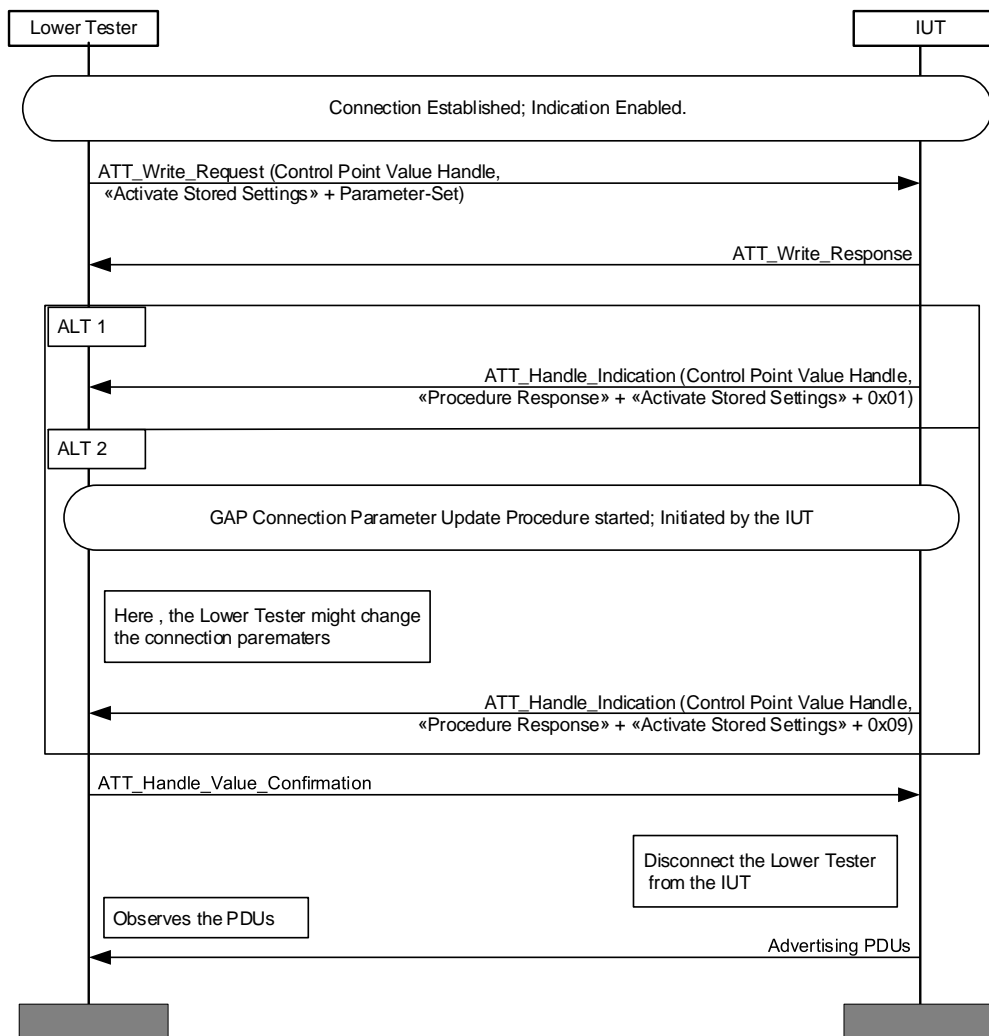
Figure 4.3: Message sequence chart for Activate Stored Settings

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDUs in the given order.

  Either the steps 1 to 3 or 1 to 2 and 4 to 6 are executed.

  In both cases the steps 7 to 10 are executed.

  If step 4 is executed, the L2CAP Connection Parameter Update Request PDU contains the connection parameters of the used stored parameter-set.

  If the IUT supports E2E-CRC, the checksum sent in the ATT_Handle_Value_Indication PDU must be present.

  The observed advertising PDUs follow parameters from the stored parameter-set used.

RCS/SR/RCCP/BI-03-C [Activate Stored Settings - Invalid Operand]

- Test Purpose

  Verify that the IUT responds with an error code when the activate stored settings procedure is started with a not supported stored parameter-set.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.4, 3.3.3

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

  - The number of supported stored parameter-sets by the IUT are known from the IXIT [6].

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Activate Stored Settings» and an operand octet with the number of the first not supported stored parameter-set.
  2. The RC server IUT responds with an *ATT_Write_Response.*
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Activate Stored Settings» request opcode and the Result Code value of 0x03 (Invalid Operand).
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation.*

- Expected Outcome

  <u>Pass verdict</u>

  The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Activate Stored Settings» request opcode and the Result Code value of 0x03 (Invalid Operand).

  If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDUs must be present.

RCS/SR/RCCP/BV-15-C [Get Max Values]

- Test Purpose

  Verify that the IUT can execute the Get Max Values procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.5

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Get Max Values».
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Communication Parameter Response» opcode, followed by the «Get Max Values» request opcode and a Response Parameter, containing the IUT's maximum values.
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

  <u>Pass verdict</u>

  The IUT sends the requested PDUs in the given order.

  If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

  All reported maximum values are within their specified ranges.

RCS/SR/RCCP/BV-16-C [Get Min Values]

- Test Purpose

  Verify that the IUT can execute the Get Min Values procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.6

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Get Min Values».
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Communication Parameter Response» opcode, followed by the «Get Min Values» request opcode and a Response Parameter, containing the IUT's minimum values.
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDUs in the given order.

  If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

  All reported minimum values are within their specified ranges.

4.6.4 Get Stored Values

- Test Purpose

  This test group verifies that the IUT can execute the Get Stored Values procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.7

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

- The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

- The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Case Configuration

Test Case	Parameter-Set to be used
RCS/SR/RCCP/BV-17-C [Get Stored Parameter-Set 0 Values]	Use parameter-set 0.
RCS/SR/RCCP/BV-18-C [Get Stored Values]	Use parameter-set 1.

Table 4.9: Get Stored Values test cases

- Test Procedure

For every test case, there is exactly one parameter-set to use. The parameter to the get stored values procedure is the number of the parameter-set.

1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Get Stored Values» and an operand octet with the parameter-set number.
2. The RC server IUT responds with an *ATT_Write_Response*.
3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Communication Parameter Response» opcode, followed by the «Get Stored Values» request opcode and a Response Parameter, containing the addressed parameter-set values.
4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

Pass verdict

The IUT sends the requested PDUs in the given order.

If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

All fields of the returned Response Parameter contain either 0xFFFF or a value that is within the allowed range of the field.

RCS/SR/RCCP/BI-04-C [Get Stored Values - Invalid Operand]

- Test Purpose

Verify that the IUT responds with an error code when the get stored values procedure is started with a stored parameter-set that is not supported by the IUT.

- Reference

[3] 3.3.1, 3.3.2, 3.3.2.7, 3.3.3

- Initial Condition

- Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

- Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

- The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

- The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- The number of supported stored parameter-sets by the IUT are known.

- Test Procedure

    1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Get Stored Values» and an operand octet with the number of the first not supported stored parameter-set.
    2. The RC server IUT responds with an *ATT_Write_Response*.
    3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Get Stored Values» request opcode and the Result Code value of 0x03 (Invalid Operand).
    4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

    Pass verdict

    The IUT sends the requested PDUs in the given order.

    If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDUs must be present.

RCS/SR/RCCP/BV-19-C [Set Filter Accept List Timer]

- Test Purpose

    Verify that the IUT can execute the Set Filter Accept List Timer procedure.

- Reference

    [3] 3.3.1, 3.3.2, 3.3.2.8

- Initial Condition

    - Establish a connection between the Lower Tester 1 and IUT using the Preamble in Section 4.2.1.

    - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

    - Create an entry in the IUT's filter accept list that contains the Lower Tester 1's device address and device address type.

    - For Lower Tester 2, change the Lower Tester 1's device address to an address not within the IUT's filter accept list, or use a different Lower Tester with an address not within the IUT's filter accept list.

    - The Lower Tester 2 knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

- The Lower Tester 1 has discovered the IUT's supported features using the test cases in Section 4.5.

- The filter accept list timer values accepted by the IUT are known by executing RCS/SR/RCCP/BV-21-C [Get Filter Accept List Timer] or by any other means.

- Test Procedure

   1. The Lower Tester 1 sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Set Filter Accept List Timer» and a 32-bit unsigned operand set to the chosen filter accept list timer value of 120 seconds.
   2. The RC server IUT responds with an *ATT_Write_Response*.
   3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Set Filter Accept List Timer» request opcode and the Result Code value of 0x01 (Success).
   4. The Lower Tester 1 sends an *ATT_Handle_Value_Confirmation*.
   5. Disconnect the Lower Tester 1 from the IUT.
   6. Start a timer.
   7. Try to connect the Lower Tester 2 to the IUT while the filter accept list timer is still running
   8. Connect the Lower Tester 2 to the IUT after the filter accept list timer value has been reached by the started timer.

- Expected Outcome

   Pass verdict

   The IUT sends the requested PDUs in the given order.

   If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

   Until the filter accept list timer value has been reached the RC server IUT will not reply to any data channel PDUs and let the Lower Tester's connection attempts time out.

   After the filter accept list timer value has been reached, the RC server IUT will respond to connection attempts by replying on data channel PDUs after the connection request was received.

RCS/SR/RCCP/BV-20-C [Disable Filter Accept List]

- Test Purpose

   Verify that the IUT can execute the Set Filter Accept List Timer procedure to disable the Filter Accept List.

- Reference

   [3] 3.3.1, 3.3.2, 3.3.2.8

- Initial Condition

   - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

   - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

   - Create an entry in the IUT's filter accept list that contains the Lower Tester's device address and device address type.

- The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

- The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- The filter accept list timer values accepted by the IUT are known by executing RCS/SR/RCCP/BV-21-C [Get Filter Accept List Timer] or by any other means.

- Test Procedure

    1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Set Filter Accept List Timer» and a 32-bit unsigned operand set to 0.
    2. The RC server IUT responds with an *ATT_Write_Response*.
    3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Set Filter Accept List Timer» request opcode and the Result Code value of 0x01 (Success).
    4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.
    5. Disconnect the Lower Tester from the IUT.
    6. Change the Lower Tester's device address to an address not within the IUT's filter accept list, or use a different Lower Tester with an address not within the IUT's filter accept list.
    7. Try to connect the Lower Tester to the IUT.

- Expected Outcome

    Pass verdict

    The IUT sends the requested PDUs in the given order.

    If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

    The RC server IUT will confirm the connection attempts by replying on data channel PDUs after the connection request was received.

RCS/SR/RCCP/BV-21-C [Get Filter Accept List Timer]

- Test Purpose

    Verify that the IUT can execute the Get Filter Accept List Timer procedure.

- Reference

    [3] 3.3.1, 3.3.2, 3.3.2.9

- Initial Condition

    - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

    - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

- The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

- The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Get Filter Accept List Timer» without any operands.
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Filter Accept List Timer Response» opcode and followed by 3 32-bit unsigned integers.
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDUs in the given order.

  If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

4.6.5 Set Advertisement Configuration

- Test Purpose

  This test group verifies that the IUT can execute the Set Advertisement Configuration procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.10

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Case Configuration

Test Case	Advertisement Configuration	Procedure Operand	Expected Advertising PDU Type
RCS/SR/RCCP/BV-22-C [Set Connectable Undirected Advertisement Configuration]	1	0	ADV_IND

Test Case	Advertisement Configuration	Procedure Operand	Expected Advertising PDU Type
RCS/SR/RCCP/BV-23-C [Set Scannable Undirected Advertisement Configuration]	2	1	ADV_SCAN_IND
RCS/SR/RCCP/BV-24-C [Set Non-Connectable Undirected Advertisement Configuration]	3	2	ADV_NONCONN_IND
RCS/SR/RCCP/BV-25-C [Set Connectable Directed Advertisement Configuration]	4	3	ADV_DIRECT_IND

Table 4.10: Set Advertisement Configuration test cases

- Test Procedure

    1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Set Advertisement Configuration» and an 8-bit unsigned operand set to the procedure operand from Table 4.10.
    2. The RC server IUT responds with an *ATT_Write_Response*.
    3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Set Advertisement Configuration» request opcode and the Result Code value of 0x01 (Success).
    4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.
    5. Disconnect the Lower Tester from the IUT.
    6. Do whatever is necessary to let the IUT start advertising.
    7. The Lower Tester observes the IUT's advertising PDUs.

- Expected Outcome

    Pass verdict

    The IUT sends the requested PDUs in the given order.

    If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

    The Lower Tester observes the expected advertising PDU type.

RCS/SR/RCCP/BV-26-C [Upgrade to LESC Only Pairing]

- Test Purpose

    Verify that the IUT can upgrade to LESC Only pairing by executing the Upgrade to LESC Only procedure.

- Reference

    [3] 3.3.1, 3.3.2, 3.3.2.11

- Initial Condition

    - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

    - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

- The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

- The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- The Lower Tester is not bonded to the IUT.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Upgrade to LESC Only» and an 8-bit unsigned operand set to 0xFF.
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Upgrade to LESC Only» request opcode and the Result Code value of 0x01 (Success).
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDUs in the given order.

  If E2E-CRC is supported, the checksum is present in the *ATT_Handle_Value_Indication* PDU.

RCS/SR/RCCP/BV-27-C [Downgrade from LESC Only Pairing]

- Test Purpose

  Verify that the IUT can downgrade from LESC Only pairing by executing the Downgrade from LESC Only procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.11

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Upgrade to LESC Only» and an 8-bit unsigned operand set to 0.

2. The RC server IUT responds with an *ATT_Write_Response.*
3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Upgrade to LESC Only» request opcode and the Result Code value of 0x01 (Success).
4. The Lower Tester sends an *ATT_Handle_Value_Confirmation.*

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDUs in the given order.

  If E2E-CRC is supported, the checksum is present in the *ATT_Handle_Value_Indication* PDU.

RCS/SR/RCCP/BV-28-C [Switch OOB Pairing On]

- Test Purpose

  Verify that the IUT can enable the out of band pairing mechanism by executing the Switch OOB Pairing procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.12

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Switch OOB Pairing» and an 8-bit unsigned operand set to 0xFF.
  2. The RC server IUT responds with an *ATT_Write_Response.*
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Switch OOB Pairing» request opcode and the Result Code value of 0x01 (Success).
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation.*

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDUs in the given order.

  If E2E-CRC is supported, the checksum is present in the *ATT_Handle_Value_Indication* PDU.

RCS/SR/RCCP/BV-29-C [Switch OOB Pairing Off]

- Test Purpose

  Verify that the IUT can disable the out of band pairing mechanism by executing the Switch OOB Pairing procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.12

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Switch OOB Pairing» and an 8-bit unsigned operand set to 0.
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Switch OOB Pairing» request opcode and the Result Code value of 0x01 (Success).
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDUs in the given order.

  If E2E-CRC is supported, the checksum is present in the *ATT_Handle_Value_Indication* PDU.

RCS/SR/RCCP/BV-30-C [Set Limited Access]

- Test Purpose

  Verify that the IUT can limit the access to the IUT by executing the Limited Access procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.13

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

- The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

- The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Limited Access» and an 8-bit unsigned operand set to 0xFF.
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Limited Access» request opcode and the Result Code value of 0x01 (Success).
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDUs in the given order.

  If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

- Notes

  The side effect from limiting the access is tested elsewhere.

RCS/SR/RCCP/BV-31-C [Reset Limited Access]

- Test Purpose

  Verify that the IUT can reset full access to the IUT by executing the Limited Access procedure.

- Reference

  [3] 3.3.1, 3.3.2, 3.3.2.13

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

    1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Limited Access» and an 8-bit unsigned operand set to 0.
    2. The RC server IUT responds with an *ATT_Write_Response*.
    3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Limited Access» request opcode and the Result Code value of 0x01 (Success).
    4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

    Pass verdict

    The IUT sends the requested PDUs in the given order.

    If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

- Notes

    The side effect from resetting to full access is tested elsewhere.

RCS/SR/RCCP/BI-05-C [Op Code not supported]

- Test Purpose

    Verify that the IUT behaves correctly when a Control Point procedure is invoked with an opcode that is not supported by the IUT.

- Reference

    [3] 3.3, 3.3.3

- Initial Condition

    - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

    - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

    - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

    - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

    1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an 8-bit opcode, which is not supported by the IUT, and no additional parameters.
    2. The RC server IUT responds with an *ATT_Write_Response*.
    3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure

Response» opcode, followed by the invalid request opcode, and the Result Code value of 0x02 (Op Code not supported).
4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

Pass verdict

The IUT sends the requested PDUs in the given order.

If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

RCS/SR/RCCP/BI-06-C [Invalid Operand]

- Test Purpose

Verify that the IUT behaves correctly when a Control Point procedure is invoked with an invalid operand.

- Reference

[3] 3.3, 3.3.3

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an 8-bit opcode of 0x0A (Upgrade to LESC Only) or 0x0B (Switch OOB Pairing) and an operand of 0xAA.
  2. The RC server IUT responds with an *ATT_Write_Response*.
  3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode, followed by the request opcode, and the Result Code value of 0x03 (Invalid Operand).
  4. The Lower Tester sends an *ATT_Handle_Value_Confirmation*.

- Expected Outcome

Pass verdict

The IUT sends the requested PDUs in the given order.

If the IUT supports E2E-CRC, the checksum sent in the *ATT_Handle_Value_Indication* PDU must be present.

RCS/SR/RCCP/BI-07-C [Client Characteristic Configuration Descriptor Improperly Configured]

- Test Purpose

  Verify that the IUT responds appropriately when a Client attempts to perform an RCCP procedure with a Client Characteristic Configuration descriptor that is improperly configured.

- Reference

  [3] 3.3, 3.3.3

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains a valid Control Point procedure request for a procedure that is supported by the IUT and with parameters that are known to be accepted by the IUT.
  2. The RC server IUT responds with an *ATT_Error_Response* with the request opcode set to «Write Request», the attribute handle set to the handle of the RC Control Point Characteristic Value declaration, and the error code set to «Client Characteristic Configuration Descriptor Improperly Configured».

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDU.

4.7   Reconnection Settings

4.7.1   Reconnection Settings

- Test Purpose

  This test group verifies that the IUT's internal state, that is changed by executing RC Control Point procedures, is reflected in the RC Settings characteristic.

- Reference

  [3] 3.2.1

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - The Lower Tester knows the handle of the RC Settings characteristic value declaration by executing RCS/SR/SGGIT/CHA/BV-03-C [Characteristic GGIT – RC Settings].

- Test Case Configuration

Test Case	Additional Initialization Step	Value Position	Expected Value
RCS/SR/RCSET/BV-01-C [Switch to LESC On]	Switch on LESC pairing by executing RCS/SR/RCCP/BV-26-C [Upgrade to LESC Only Pairing].	Octet 0; Bit 1	1
RCS/SR/RCSET/BV-02-C [Switch to LESC Off]	Switch off LESC pairing by executing RCS/SR/RCCP/BV-27-C [Downgrade from LESC Only Pairing].	Octet 0; Bit 1	0
RCS/SR/RCSET/BV-03-C [Use OOB Pairing On]	Switch on OOB pairing by executing RCS/SR/RCCP/BV-28-C [Switch OOB Pairing On].	Octet 0; Bit 2	1
RCS/SR/RCSET/BV-04-C [Use OOB Pairing Off]	Switch off OOB pairing by executing RCS/SR/RCCP/BV-29-C [Switch OOB Pairing Off].	Octet 0; Bit 2	0
RCS/SR/RCSET/BV-05-C [Limited Access On]	Limit the access to the IUT by executing RCS/SR/RCCP/BV-30-C [Set Limited Access].	Octet 0; Bits 5 and 6	1 and 1
RCS/SR/RCSET/BV-06-C [Limited Access Off]	Reset the access limitation to the IUT by executing RCS/SR/RCCP/BV-31-C [Reset Limited Access].	Octet 0; Bits 5 and 6	0 and 0
RCS/SR/RCSET/BV-07-C [Advertisement Configuration 1]	Set advertisement configuration by executing RCS/SR/RCCP/BV-22-C [Set Connectable Undirected Advertisement Configuration].	Octet 1; Bit 2 and 3	0 and 0
RCS/SR/RCSET/BV-08-C [Advertisement Configuration 2]	Set advertisement configuration by executing RCS/SR/RCCP/BV-23-C [Set Scannable Undirected Advertisement Configuration].	Octet 1; Bit 2 and 3	1 and 0
RCS/SR/RCSET/BV-09-C [Advertisement Configuration 3]	Set advertisement configuration by executing RCS/SR/RCCP/BV-24-C [Set Non-Connectable Undirected Advertisement Configuration].	Octet 1; Bit 2 and 3	0 and 1
RCS/SR/RCSET/BV-10-C [Advertisement Configuration 4]	Set advertisement configuration by executing RCS/SR/RCCP/BV-25-C [Set Connectable Directed Advertisement Configuration].	Octet 1; Bit 2 and 3	1 and 1
RCS/SR/RCSET/BV-11-C [Ready for Disconnect]	Do whatever is necessary for the IUT to be ready to be disconnected	Octet 0; Bit 4	1
RCS/SR/RCSET/BV-12-C [Not Ready for Disconnect]	Do whatever is necessary for the IUT to be not ready to be disconnected	Octet 0; Bit 4	0

Table 4.11: Reconnection Settings test cases

- Test Procedure

    1. Execute the additional initialization step described in Table 4.11.
    2. The Lower Tester sends an *ATT_Read_Request* with the attribute handle set to the handle of the RC Settings characteristic value declaration.
    3. The IUT replies with an *ATT_Read_Response.*

- Expected Outcome

  <u>Pass verdict</u>

  If the IUT supports E2E-CRC, the E2E-CRC field must be present.

  The received value contains a valid length field and a settings field.

  From Table 4.11, the expected values show up at the value position.

RCS/SR/RCSET/BV-13-C [Ready for Disconnect Notification]

- Test Purpose

  Verify that the Ready for Disconnect bit will be notified by the IUT.

- Reference

  [3] 3.2.1, 3.2.2

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - Do whatever is necessary for the IUT to be not ready to be disconnected.

  - The Lower Tester knows the handle of the RC Settings characteristic value declaration by executing RCS/SR/SGGIT/CHA/BV-03-C [Characteristic GGIT – RC Settings].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. Do whatever is necessary for IUT to become ready to be disconnected.
  2. The IUT sends an ATT Handle Value Notification to the Lower Tester.

- Expected Outcome

  <u>Pass verdict</u>

  If the IUT supports E2E-CRC, the E2E-CRC field must be present.

  The received value contains a valid length field and a settings field.

  Bit 4 of the first octet in the settings field of the notified characteristic value is set to 1.

RCS/SR/RCSET/BV-14-C [Downgrade from LESC by Timer]

- Test Purpose

  Verify that the IUT implements a timer that resets the LESC Only-bit in the RC Settings characteristic.

- Reference

  [3] 3.2.1, 3.3.2.11

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - Set the LESC Only-bit in the IUT's RC Settings by executing RCS/SR/RCCP/BV-26-C [Upgrade to LESC Only Pairing].

  - The IUT's timeout (fallback timer) and accuracy for the tested timer is known to the Lower Tester from IXIT [6].

- Test Procedure

  1. Disconnect the Lower Tester from the IUT.
  2. Wait until the fallback-timer safely expires.
  3. Reestablish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1
  4. Read RCSET characteristic and verify that the LESC Only-bit was reset by the IUT, Expected Outcome

  Pass verdict

  Reconnection is possible with the security settings of the initial connection.

  The LESC Only-bit was reset.

RCS/SR/RCSET/BV-15-C [Switch OOB Pairing Off by Timer]

- Test Purpose

  Verify that the IUT implements a timer that resets the Use OOB Pairing-bit in the RC Settings characteristic.

- Reference

  [3] 3.2.1, 3.3.2.12

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - Set the Use OOB Pairing –bit in the IUT's RC Settings by executing RCS/SR/RCCP/BV-28-C [Switch OOB Pairing On].

  - The IUT's timeout and accuracy for the tested timer is known to the Lower Tester from IXIT [6].

- Test Procedure

  1. Disconnect the Lower Tester from the IUT.
  2. Wait until the timer safely expired.
  3. Verify that the Use OOB Pairing-bit was reset by the IUT, by executing RCS/SR/RCSET/BV-04-C [Use OOB Pairing Off] without the additional initialization step.

- Expected Outcome

Pass verdict

The test procedure in step 3 is executed successfully.

Bit 2 of the first octet in the Settings field, of the RC Settings characteristic, is set to 0.

4.8 Limited Access Behavior

RCS/SR/RCLAB/BV-01-C [Access Disabled RC Service – Settings Value]

- Test Purpose

Verify that an IUT that is disabled, signals the limited access to connected clients.

- Reference

[3] 3.2.1

- Initial Condition

  - Establish a connection between the Lower Tester 2 and the IUT using the Preamble in Section 4.2.1.

  - The Lower Tester 2 limits the access to the IUT by executing RCS/SR/RCCP/BV-30-C [Set Limited Access] or by any other means and disconnect.

- Test Procedure

  1. Establish a connection between the Lower Tester 1 and the IUT using the Preamble in Section 4.2.1. The Lower Tester 1 is not the client who executed the Limited Access procedure on the IUT.
  2. The Lower Tester 1 acquires the handle of the RC Settings characteristic value declaration by executing RCS/SR/SGGIT/CHA/BV-03-C [Characteristic GGIT – RC Settings].
  3. The Lower Tester 1 sends an *ATT_Read_Request* with the attribute handle set to the handle of the RC Settings characteristic value declaration.
  4. The IUT replies with an *ATT_Read_Response*.

- Expected Outcome

Pass verdict

If the IUT supports E2E-CRC, the E2E-CRC field must be present.

The received value contains a valid length field and a settings field.

Bit 5 of the first octet in the settings field of the read characteristic value is set to 1.

Bit 6 of the first octet in the settings field of the read characteristic value is set to 0.

RCS/SR/RCLAB/BV-02-C [Access Disable RC Service – Executing RCCP Procedure]

- Test Purpose

Verify that an IUT with disabled RC service will reply with an error message, when a RCCP procedure is invoked.

- Reference

[3] 3.2.2, 3.3.2.13

- Initial Condition

  - Establish a connection between the Lower Tester 2 and the IUT using the Preamble in Section 4.2.1.

  - The Lower Tester 2 limits the access to the IUT by executing RCS/SR/RCCP/BV-30-C [Set Limited Access] or by any other means and disconnect.

- Test Procedure

  1. Establish a connection between the Lower Tester 1 and the IUT using the Preamble in Section 4.2.1. The Lower Tester 1 is not the client who executed the Limited Access procedure on the IUT.
  2. The Lower Tester 1 acquires the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].
  3. The Lower Tester 1 sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to one octet containing the «Limited Access» opcode, followed by an octet with a value of 0x00.
  4. The IUT replies with an *ATT_Error_Response*.

- Expected Outcome

  Pass verdict

  The *ATT_Error_Response* contains a request attribute opcode set to 0x12, a request handle set to the handle of the RC Control Point Characteristic Value declaration, and an error code set to 0x08 (Insufficient Authorization).

4.9   End to End – Cyclic redundancy check

RCS/SR/E2ECRC/BV-01-C [Correct CRC Algorithm]

- Test Purpose

  Verify that the IUT uses the correct CRC.

- Reference

  [3] 3.4

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

    1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Activate Stored Settings» and an operand octet set to 0. The 16-bit checksum is appended to the value.
    2. The RC server IUT responds with an *ATT_Write_Response*.

    Depending on the differences between the actual communication parameters and the values of the stored parameters-set 0, the test has two different outcomes:

    3. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Activate Stored Settings» request opcode and the Result Code value of 0x01 (Success). The 16-bit checksum is appended to the characteristic value.

    Or

    4. The IUT sends an *ATT_Handle_Value_Indication* PDU with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration and the value set to the «Procedure Response» opcode and followed by the «Activate Stored Settings» request opcode and the Result Code value of 0x09 (Proposal Accepted). The 16-bit checksum is appended to the characteristic value.

    In both cases, the last step is:

    5. The Lower Tester sends an *ATT_Handle_Value_Confirmation.*

- Expected Outcome

    <u>Pass verdict</u>

    The IUT responds with one *ATT_Handle_Value_Indication* PDU with the handle of the RC Control Point Characteristic Value declaration.

    The indication contains either the following octets as value:

    0x0E 0x03 0x01 0xC9 0x12

    or the value:

    0x0E 0x03 0x09 0x81 0x9E

RCS/SR/E2ECRC/BI-01-C [Verify E2E-CRC]

- Test Purpose

    Verify that the IUT verifies the E2E-CRC sent by a Lower Tester and rejects requests with invalid calculated E2E-CRC.

- Reference

    [3] 1.6

- Initial Condition

    - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

    - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

- The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

- The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Activate Stored Settings» and an operand octet set to 0. A wrong 16-bit checksum is appended to the value.
  2. The RC server IUT responds with an *ATT_Error_Response* PDU containing the handle of the RC Control Point Characteristic and the «Invalid CRC» error code.

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDU.

RCS/SR/E2ECRC/BI-02-C [Verify E2E-CRC existence]

- Test Purpose

  Verify that the IUT verifies the existence of a required checksum.

- Reference

  [3] 1.6

- Initial Condition

  - Establish a connection between the Lower Tester and IUT using the Preamble in Section 4.2.1.

  - Configure the IUT for notifications and indications using the Preamble in Section 4.2.2.

  - The Lower Tester knows the handle of the RC Control Point Characteristic Value declaration by executing RCS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – Reconnection Configuration Control Point].

  - The Lower Tester has discovered the IUT's supported features using the test cases in Section 4.5.

- Test Procedure

  1. The Lower Tester sends an *ATT_Write_Request* with the attribute handle set to the handle of the RC Control Point Characteristic Value declaration. The value contains an opcode set to «Activate Stored Settings» and an operand octet set to 0. No checksum is appended to the value.
  2. The RC server IUT responds with an *ATT_Error_Response* PDU containing the handle of the RC Control Point Characteristic and the «Missing CRC» error code.

- Expected Outcome

  Pass verdict

  The IUT sends the requested PDU.

5 Test case mapping

The Test Case Mapping Table (TCMT) maps test cases to specific requirements in the ICS. The IUT will be 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 the Reconnection Configuration Service [4].

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

Test Case(s): The applicable test case identifiers, 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)
RCS 3/1	Reconnection Configuration Service	RCS/SR/SGGIT/SER/BV-01-C
RCS 1/1 AND NOT RCS 5a/1	RC Feature Characteristic	RCS/SR/SGGIT/CHA/BV-02-C
RCS 1/1 AND RCS 5a/2	RC Feature characteristic indication	RCS/SR/SGGIT/CHA/BV-06-C RCS/SR/SGGIT/ISFC/BV-07-C
(RCS 6/10 OR RCS 4/14 OR RCS 4/15 OR RCS 4/18) AND NOT RCS 4/3	RC Settings Characteristic	RCS/SR/SGGIT/CHA/BV-03-C
RCS 4/3	RC Settings Characteristic with notify property	RCS/SR/SGGIT/CHA/BV-04-C RCS/SR/CON/BV-01-C
RCS 5/4	RC Control Point Characteristic	RCS/SR/SGGIT/CHA/BV-05-C RCS/SR/CON/BV-02-C RCS/SR/RCCP/BI-05-C RCS/SR/RCCP/BI-07-C
RCS 6/11 OR RCS 6/12	RC Control Point Characteristic, Invalid Operand	RCS/SR/RCCP/BI-06-C
RCS 4/1	E2E-CRC supported	RCS/SR/RCFEA/BV-01-C
RCS 4/1 AND RCS 6/4	Verify E2E-CRC	RCS/SR/E2ECRC/BV-01-C RCS/SR/E2ECRC/BI-01-C RCS/SR/E2ECRC/BI-02-C
RCS 1/1 AND NOT RCS 4/1	E2E-CRC not supported	RCS/SR/RCFEA/BV-02-C
RCS 4/2	Enable Disconnect Supported	RCS/SR/RCFEA/BV-03-C
RCS 1/1 AND NOT RCS 4/2	Enable Disconnect not Supported	RCS/SR/RCFEA/BV-04-C

Item	Feature	Test Case(s)
RCS 4/3	Ready for Disconnect Supported	RCS/SR/RCFEA/BV-05-C RCS/SR/RCSET/BV-11-C RCS/SR/RCSET/BV-12-C RCS/SR/RCSET/BV-13-C
RCS 1/1 AND NOT RCS 4/3	Ready for Disconnect not Supported	RCS/SR/RCFEA/BV-06-C
RCS 4/16	Propose Reconnection Timeout Supported	RCS/SR/RCFEA/BV-07-C
RCS 1/1 AND NOT RCS 4/16	Propose Reconnection Timeout not Supported	RCS/SR/RCFEA/BV-08-C
RCS 4/7	Propose Connection Interval Supported	RCS/SR/RCFEA/BV-09-C
RCS 1/1 AND NOT RCS 4/7	Propose Connection Interval not Supported	RCS/SR/RCFEA/BV-10-C
RCS 4/8	Propose Peripheral Latency Supported	RCS/SR/RCFEA/BV-11-C
RCS 1/1 AND NOT RCS 4/8	Propose Peripheral Latency not Supported	RCS/SR/RCFEA/BV-12-C
RCS 4/9	Propose Supervision Timeout Supported	RCS/SR/RCFEA/BV-13-C
RCS 1/1 AND NOT RCS 4/9	Propose Supervision Timeout not Supported	RCS/SR/RCFEA/BV-14-C
RCS 4/4	Propose Advertisement Interval Supported	RCS/SR/RCFEA/BV-15-C RCS/SR/RCCP/BV-09-C
RCS 1/1 AND NOT RCS 4/4	Propose Advertisement Interval not Supported	RCS/SR/RCFEA/BV-16-C
RCS 4/5	Propose Advertisement Count Supported	RCS/SR/RCFEA/BV-17-C RCS/SR/RCCP/BV-10-C
RCS 1/1 AND NOT RCS 4/5	Propose Advertisement Count not Supported	RCS/SR/RCFEA/BV-18-C
RCS 4/6	Propose Advertisement Repetition Supported	RCS/SR/RCFEA/BV-19-C RCS/SR/RCCP/BV-11-C
RCS 1/1 AND NOT RCS 4/6	Propose Advertisement Repetition not Supported	RCS/SR/RCFEA/BV-20-C
RCS 4/10	Advertisement Configuration 1 Supported	RCS/SR/RCFEA/BV-21-C RCS/SR/RCCP/BV-22-C RCS/SR/RCSET/BV-07-C
RCS 1/1 AND NOT RCS 4/10	Advertisement Configuration 1 not Supported	RCS/SR/RCFEA/BV-22-C
RCS 4/11	Advertisement Configuration 2 Supported	RCS/SR/RCFEA/BV-23-C RCS/SR/RCCP/BV-23-C RCS/SR/RCSET/BV-08-C
RCS 1/1 AND NOT RCS 4/11	Advertisement Configuration 2 not Supported	RCS/SR/RCFEA/BV-24-C
RCS 4/12	Advertisement Configuration 3 Supported	RCS/SR/RCFEA/BV-25-C RCS/SR/RCCP/BV-24-C RCS/SR/RCSET/BV-09-C

Item	Feature	Test Case(s)
RCS 1/1 AND NOT RCS 4/12	Advertisement Configuration 3 not Supported	RCS/SR/RCFEA/BV-26-C
RCS 4/13	Advertisement Configuration 4 Supported	RCS/SR/RCFEA/BV-27-C RCS/SR/RCCP/BV-25-C RCS/SR/RCSET/BV-10-C
RCS 1/1 AND NOT RCS 4/13	Advertisement Configuration 4 not Supported	RCS/SR/RCFEA/BV-28-C
RCS 4/14	Upgrade to LESC Pairing Supported	RCS/SR/RCFEA/BV-29-C
RCS 1/1 AND NOT RCS 4/14	Upgrade to LESC Pairing not Supported	RCS/SR/RCFEA/BV-30-C
RCS 4/15	Switch to OOB Pairing Supported	RCS/SR/RCFEA/BV-31-C
RCS 1/1 AND NOT RCS 4/15	Switch to OOB Pairing not Supported	RCS/SR/RCFEA/BV-32-C
RCS 4/17	Use of Filter Accept List Supported	RCS/SR/RCFEA/BV-33-C
RCS 1/1 AND NOT RCS 4/17	Use of Filter Accept List not Supported	RCS/SR/RCFEA/BV-34-C
RCS 4/18	Limited Access Supported	RCS/SR/RCFEA/BV-35-C
RCS 1/1 AND NOT RCS 4/18	Limited Access not Supported	RCS/SR/RCFEA/BV-36-C
RCS 1/1	Other Feature Flags	RCS/SR/RCFEA/BV-37-C RCS/SR/RCFEA/BV-38-C
RCS 6/1	Enable Disconnect Procedure	RCS/SR/RCCP/BV-01-C RCS/SR/RCCP/BI-01-C
RCS 6/2	Get Actual Communication Parameters	RCS/SR/RCCP/BV-02-C
RCS 4/16	Propose Reconnection Timeout Settings	RCS/SR/RCCP/BV-03-C RCS/SR/RCCP/BV-04-C
RCS 4/7	Propose Connection Interval Settings	RCS/SR/RCCP/BV-05-C RCS/SR/RCCP/BI-02-C
RCS 4/8	Propose Peripheral Latency Settings	RCS/SR/RCCP/BV-06-C
RCS 4/9	Propose Supervision Timeout Settings	RCS/SR/RCCP/BV-07-C
RCS 4/7 OR RCS 4/8 OR RCS 4/9	Propose Connection Parameter Settings	RCS/SR/RCCP/BV-08-C
RCS 4/4 OR RCS 4/5 OR RCS 4/6	Propose Advertisement Parameter Settings	RCS/SR/RCCP/BV-12-C
RCS 6/4	Activate Stored Settings	RCS/SR/RCCP/BV-13-C RCS/SR/RCCP/BV-14-C RCS/SR/RCCP/BI-03-C
RCS 6/5	Get Max Values	RCS/SR/RCCP/BV-15-C
RCS 6/6	Get Min Values	RCS/SR/RCCP/BV-16-C

Item	Feature	Test Case(s)
RCS 6/7	Get Stored Values	RCS/SR/RCCP/BV-17-C RCS/SR/RCCP/BV-18-C RCS/SR/RCCP/BI-04-C
RCS 6/8	Set Filter Accept List Timer	RCS/SR/RCCP/BV-19-C RCS/SR/RCCP/BV-20-C
RCS 6/9	Get Filter Accept List Timer	RCS/SR/RCCP/BV-21-C
RCS 6/11	Switch LESC Pairing	RCS/SR/RCCP/BV-26-C RCS/SR/RCCP/BV-27-C RCS/SR/RCSET/BV-01-C RCS/SR/RCSET/BV-02-C
RCS 6/12	Switch OOB Pairing	RCS/SR/RCCP/BV-28-C RCS/SR/RCCP/BV-29-C RCS/SR/RCSET/BV-03-C RCS/SR/RCSET/BV-04-C
RCS 6/13	Limited Access	RCS/SR/RCCP/BV-30-C RCS/SR/RCCP/BV-31-C RCS/SR/RCSET/BV-05-C RCS/SR/RCSET/BV-06-C RCS/SR/RCLAB/BV-01-C RCS/SR/RCLAB/BV-02-C
RCS 9/2	Upgrade to LESC Fallback Timer	RCS/SR/RCSET/BV-14-C
RCS 9/1	Switch OOB Pairing Fallback Timer	RCS/SR/RCSET/BV-15-C

Table 5.1: Test case mapping

6 Revision history and acknowledgments

Revision History

Publication Number	Revision Number	Date	Comments
	D0.9.0r00	2016-04-29	Initial draft
	D0.9.0r01	2016-07-13	Incorporated changes based on modification of Service
	D0.9.0r02	2016-08-30	Incorporated first feedback from BTI and other changes
	D0.90r03	2016-09-09	Incorporated missing test cases
	D0.9.0r04	2016-10-24	Incorporated feedback from BTI
	D0.9.0r05	2016-07-11	Incorporated feedback from BTI
	D1.0.0r00	2017-05-12	Transition to 1.0 and incorporation of changes based on IOP
	D1.0.0r01	2017-08-25	Incorporated feedback from Med WG
	D1.0.0r02	2017-08-30	Incorporated feedback from Med WG
	D1.0.0r03	2017-10-13	Updated TCMT per updated RCS.ICS and Incorporated feedback from BTI
	D1.0.0r04	2017-10-20	Incorporated feedback from BTI and editorial changes
	D1.0.0r05	2017-11-01	Per BTI's feedback updated MSCs to Visio file format, renumbered BV, and BI test cases, and incorporated editorial changes
	D1.0.0r05	2017-11-24	Approved by BTI.
0	1.0.0	2017-12-05	Adopted by the Bluetooth SIG Board of Directors.
	1.0.1r00	2018-03-16	TSE 10391 (rating 1): Made editorial revisions to Section 1 (Scope). Revised initial condition and added steps 1 and 2 and revised step 3 in test procedure in RCS/SR/RCLAB/BV-01-C and 02-C. Also revised pass verdict in RCS/SR/RCLAB/BV-01-C.
1	1.0.1	2018-06-27	Approved by BTI. Prepared for TCRL 2018-1 publication.
	p2r00–r06	2021-02-21 – 2021-12-20	TSE 16859 (rating 4): Added new test group ISFC. New test cases added RCS/SR/SGGIT/CHA/BV-06-C and RCS/SR/SGGIT/ISFC/BV-07-C (E16252). Added TCMT for the new test cases. Updated Section 4.2.2. TSE 18079 (rating 2): Converted the following test cases into GGIT tests: RCS/SR/SD/BV-01-C and -02-C, RCS/SR/DEC/BV-01-C – -04-C, and RCS/SR/DES/BV-01-C and -02-C. The new GGIT converted TCIDs are: RCS/SR/SGGIT/SER/BV-01-C and RCS/SR/SGGIT/CHA/BV-02-C – -05-C. Updated TCMT for the new GGIT tests and tests impacted by the GGIT conversion. TSE 18080 (rating 1): Updated inappropriate terms "slave" with "Peripheral" and "White List" with "Filter Accept List" in test cases: RCS/SR/RCFEA/BV-11-C, -12-C, -33-C, and -34-C, and RCS/SR/RCCP/BV-06-C and -19-C – -21-C and updated the TCMT (E15809).

Publication Number	Revision Number	Date	Comments
			Performed editorial work, including updating to the latest TS template and aligning the copyright page with v2 of the DNMD.
2	p2	2022-01-25	Approved by BTI on 2022-01-06. Prepared for TCRL 2021-2 publication.