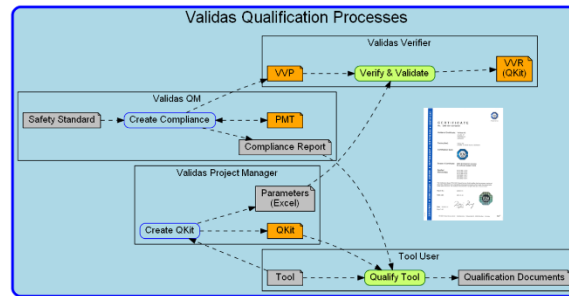


# Compliance Report for Perform Module Test



Version:	Template 0.5 / Document 1.0
Maximal Safety Level:	ASIL_D
Date:	2019-03-11
Status:	Generic / Generated / Reviewed / <b>Final</b>
Author:	Dr. Oscar Slotosch
File:	CR.docm
Size:	18 Pages



Template Revision History:

<b>Version</b>	<b>Date</b>	<b>Status</b>	<b>Author</b>	<b>Change</b>
0.1	2018-10-16	Generic	Slotosch	Created initial template
0.2	2018-10-30	Generic	Slotosch	Added Configuration and V&V, moved process to [PR]
0.3	2018-11-11	Generic	Slotosch	Added Glossary
0.4	2018-11-11	Generic	Slotosch	Added Maximal Safety Level
0.5	2019-01-18	Generic	Slotosch	Added Compliance Method Chapter

## Document Revision History

Version	Date	Status	Author	Change
0.1	2019-03-11	Generic	Slotosch	Created Generic Template 0.5
0.3	2019-03-11	Generated	oscar	Configured Process Parameters, Selected Requirements Claims and generated document using PMT
0.5	2019-03-11	Reviewed	Oscar Slotosch<RequirementsExpert>	Reviewed Requirements & Compliance
1.0	<finalization date>	Final	Oscar Slotosch	Finalized document

## Contents

<b>1</b>	<b>Scope of this Document</b>	<b>5</b>
<b>2</b>	<b>Compliance Method</b>	<b>6</b>
<b>3</b>	<b>Parameters of Perform Module Test</b>	<b>7</b>
3.1	Types in Perform Module Test	7
3.2	Project Parameter of Perform Module Test	8
3.3	Project Parameter of Perform Module Test	8
<b>4</b>	<b>Main Requirements (Claims) for Perform Module Test</b>	<b>9</b>
<b>5</b>	<b>Requirements for Perform Module Test</b>	<b>9</b>
5.1	Requirement: Structural Code Coverage [ISO-6.9.4.4]	9
<b>6</b>	<b>V&amp;V Checks for Perform Module Test</b>	<b>9</b>
6.1	Check for Path: ModuleTest/Artifacts/SafetyCase/CoverageReport	10
6.1.1	Check for Artifact: Code Coverage Report	10
6.2	Check for Path: ModuleTest/Artifacts/SafetyCase/TestReport	11
6.2.1	Check for Artifact: Test Report	11
6.3	Check for Path: ModuleTest/Artifacts/SafetyCase/Tests	12
6.3.1	Check for Artifact: Test Case	12
<b>7</b>	<b>Compliance for Perform Module Test</b>	<b>13</b>
7.1	Compliance with Structural Code Coverage [ISO-6.9.4.4]	13
<b>8</b>	<b>Glossary</b>	<b>16</b>
<b>9</b>	<b>References</b>	<b>18</b>

# 1 Scope of this Document

This document describes the compliance of the process Perform Module Test, see [PR] with the selected main requirements.

The document contains the following chapters:

- 1) This introduction
- 2) The **compliance method**: describing the used compliance method and processes
- 3) The **configuration** of the process, i.e. the types and process variables and their values
- 4) The **claims**, i.e. the main requirement that shall be satisfied from the process
- 5) The **requirements** all requirements contained in the main requirements
- 6) The **checks** that verify the product for compliance with requirements
- 7) The **compliance** argumentation for all requirements

The document concludes with a glossary in Section 8 and references in Section 9.

## 2 Compliance Method

The applied compliance method is model based. It bases on a parameterized model that is used to model the following things:

- Requirements from safety standard (or from somewhere else),
- Development process of the qualification kit (or something else),
- Compliance argumentation,
- Verification actions and
- Parameters.

The key ideas of the compliance method are:

- 1) Every requirement is linked to two things:
  - a. an element in the process that implements it and
  - b. an verification step in the process that verified the corresponding artifact
- 2) Project parameters: Every project is different, even if it follows the same process. Those differences are modeled using parameters in the process. The parameters are instantiated with values during the project. In our qualification projects typical parameters are "TEST" or "FEATURE".

Parameters can be used for project management, e.g. qualifying 20 features and creating 100 tests, but parameters have to be used for verification and validation in order to ensure requirement (and standard) compliance.

The compliance method is therefore structured in two parts

- 1) The process specific part, that is the scheme for all projects
- 2) The project specific part, which is just an instantiation of the process by defining the parameters and performing V&V for all instances as pre-scribed within the process.

The process (part 1 only) can be assessed independently. For example Validas has a TÜV certification for the processes of Tool qualification (since 2018) and Library qualification (since 2019).

This compliance documentation is, together with the process report [PR] the basis for a process certification for Perform Module Test.

The compliance method is graphically shown in Figure 1. The compliance is achieved in the following steps:

- 1) Process Part. We use the Process Modeling Tool (PMT) for this:
  - a. Model or select requirements for the process.
  - b. Model or configure the process (based on existing processes), including the verification steps. More details on process modeling can be found in the introduction of [PR].
  - c. Argue the compliance of the process by providing for every requirement at least one implementing process and one verification action
  - d. Generate the process report [PR]
  - e. Generate this compliance report
  - f. Generate the verification and validation plan, see [VVP]
- 2) Project specific part (only the verification and validation, which is essential for the compliance). We use the V&V Tool VVT for this
  - a. Instantiate the Verification and Validation Plan (VVP) by
    - i. Assigning concrete names to stakeholders that perform V&V

- ii. Import the project parameters. For qualification projects based on Validas Tool Chain Analyzer this can be done by exporting the parameters from the TCA tool into an Excel table)
- b. Perform V&V by going through all checks for all instances of the parameters (this can be done using Excel Export & import of VVT)
- c. Check Completeness and generate the Verification and Validation Report (VVR) using VVT.

The safety plan is the description of the process including the compliance argumentation and the safety case is the safety case including the V&V report with the verification results.

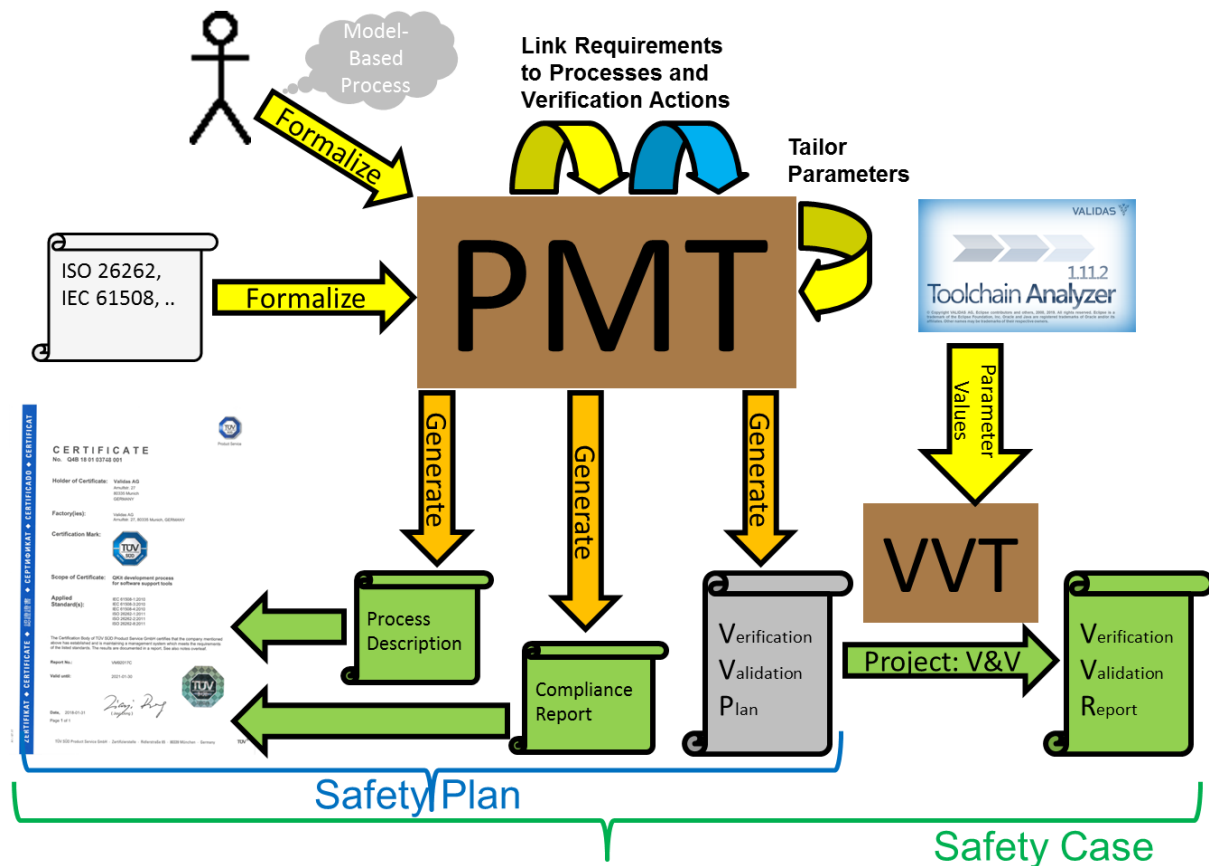


Figure 1: Compliance Method

### 3 Parameters of Perform Module Test

This section describes the 2 parameters and 2 types for Perform Module Test. The parameters are typed and this section describes also the used 2 types and their values.

#### 3.1 Types in Perform Module Test

This section describes the 2 types for 'Perform Module Test' and their possible values.

EnumType ISO_ASIL	
Name:	
ISO_ASIL	
Description:	
The type for all Automotive Safety Integrity Level (ASIL) as defined in ISO 26262.	

Enumeration Values:
<ul style="list-style-type: none"> <li>ASIL_A: Automotive Safety Integrity Level A</li> <li>ASIL_B: Automotive Safety Integrity Level B</li> <li>ASIL_C: Automotive Safety Integrity Level C</li> <li>ASIL_D: Automotive Safety Integrity Level D</li> </ul>

Table 1 EnumType ISO\_ASIL

Type String
<b>Name:</b>
String
<b>Description:</b>
The standard type of Strings with arbitrary values, usable for names, values,...

Table 2 Type String

### 3.2 Project Parameter of Perform Module Test

This section describes the 1 process variable for Perform Module Test.

ProcessVariable ISO_ASIL
<b>Name:</b>
ISO_ASIL
<b>Description:</b>
The current ASIL of the project.
<b>Type:</b>
ISO_ASIL
<b>Value:</b>
ASIL_D

Table 3 ProcessVariable ISO\_ASIL

### 3.3 Project Parameter of Perform Module Test

This section describes the 1 project parameter for Perform Module Test.

ProjectParameter MODULE
<b>Name:</b>
MODULE
<b>Description:</b>
The name of module that is currently tested.
<b>Comment:</b>
The name can vary for different modules.
<b>Type:</b>
String
<b>Value:</b>
To be defined in project

Table 4 ProjectParameter MODULE



## 4 Main Requirements (Claims) for Perform Module Test

This section describes the 1 main requirement, so called 'claim' for 'Perform Module Test'.  
The maximal safety level is ASIL\_D.

The main requirement is (see next section for the detailed description):

- Structural Code Coverage [ISO-6.9.4.4], see Table 5

Next section lists all required requirements, the following sections describe the compliance argumentations for all requirements.

## 5 Requirements for Perform Module Test

This section contains 1 requirement for Perform Module Test.

### 5.1 Requirement: Structural Code Coverage [ISO-6.9.4.4]

This section describes requirement Structural Code Coverage [ISO-6.9.4.4].

Requirement Structural Code Coverage [ISO-6.9.4.4]	
<b>Name:</b>	Structural Code Coverage
<b>ID:</b>	ISO-6.9.4.4
<b>Description:</b>	To evaluate the completeness of verification and to provide evidence that the objectives for unit testing are adequately achieved, the coverage of requirements at the software unit level shall be determined and the structural coverage shall be measured in accordance with the metrics as listed in Table 9. If the achieved structural coverage is considered insufficient, either additional test cases shall be specified or a rationale based on other methods shall be provided.
<b>Recommended From:</b>	ASIL_A
<b>Recommended To:</b>	ASIL_D
<b>Implementing Process (Satisfying this claim):</b>	<ul style="list-style-type: none"><li>• Perform Module Test</li></ul>
<b>Sub-Requirement (not included in Required):</b>	<ul style="list-style-type: none"><li>• Table 9 [ISO-6.9.4.4-9]</li></ul>

Table 5 Requirement Structural Code Coverage [ISO-6.9.4.4]

This section contains 0 compliant requirement for Perform Module Test.

Skipping 1 compliant requirement (out of scope for this document, i.e. no compliance evidence):

Skipping Table 9.

## 6 V&V Checks for Perform Module Test

Every atomic check, i.e. a check without sub-checks contains a list of criteria (questions) that have to be answered to perform the check.

This section describes the 3 explicit/atomic V&V tasks for Perform Module Test. 1 implicit check is performed implicitly/automatically and does not need to be done explicitly (see process report for it and its argument).

The checks are grouped by the verified paths and artifacts and are performed based on the V&V model.

## 6.1 Check for Path: ModuleTest/Artifacts/SafetyCase/CoverageReport

This section describes V&V of the 1 artifact located at the path ModuleTest/Artifacts/SafetyCase/CoverageReport.

### 6.1.1 Check for Artifact: Code Coverage Report

Artifact Code Coverage Report	
<b>Name:</b>	Code Coverage Report
<b>Description:</b>	The report with the selected coverage (Statement, Branch, MCDC).
<b>Artifact Hierarchy:</b>	Code Coverage Report (in Safety Case)
<b>Path:</b>	ModuleTest/Artifacts/SafetyCase/CoverageReport
<b>Main StakeHolder (Owner):</b>	Tester
<b>Created By:</b>	<ul style="list-style-type: none"> <li>Execute Test</li> <li>Measure MCDC</li> </ul>
<b>Used By:</b>	<ul style="list-style-type: none"> <li>Analyze Code Coverage</li> <li>Analyze MCDC</li> </ul>

**Table 6 Artifact Code Coverage Report**

This section describes the 1 explicit/atomic check for Code Coverage Report (in Safety Case).

VerificationModule: Analyze MCDC	
<b>Name:</b>	Analyze MCDC
<b>Description:</b>	Analyze if MCDC is 100% or if there are explained reasons why the coverage is <100%.
<b>Qualified Name:</b>	Analyze Code Coverage.Analyze MCDC
<b>Relevant Parameter:</b>	<ul style="list-style-type: none"> <li>MODULE = &lt;to be defined&gt;, see Table 4</li> </ul>
<b>Variant Condition:</b>	<ul style="list-style-type: none"> <li>(ISO_ASIL == ASIL_D)</li> </ul>
<b>Criteria:</b>	<ul style="list-style-type: none"> <li>MCDCe OK [AnaCCR-MC-C1]:Is the MCDC 100% (or are reasonable explanations</li> </ul>

given in case it is less than 100%)?
<ul style="list-style-type: none"> <li>All Conditions Considered [AnaCCR-MC-C2]:Have all conditions of the module been considered / instrumented (or are some files not instrumented that belong to the module)?</li> </ul>
<b>Compliance:</b>
<ul style="list-style-type: none"> <li>6.9.4.4-9.1c Compliance with MCDC</li> </ul>
<b>Owner (Inherited):</b>
Tester
<b>Verifies:</b>
<ul style="list-style-type: none"> <li>Artifact: Code Coverage Report, see Table 6</li> </ul>

**Table 7 VerificationModule: Analyze MCDC**

## 6.2 Check for Path: ModuleTest/Artifacts/SafetyCase/TestReport

This sections describes V&V of the 1 artifact located at the path ModuleTest/Artifacts/SafetyCase/TestReport.

### 6.2.1 Check for Artifact: Test Report

Artifact Test Report	
<b>Name:</b>	Test Report
<b>Description:</b>	The test report created from test execution including the test result (PASS/FAIL).
<b>Artifact Hierarchy:</b>	Test Report (in Safety Case)
<b>Path:</b>	ModuleTest/Artifacts/SafetyCase/TestReport
<b>Main StakeHolder (Owner):</b>	Tester
<b>Created By:</b>	<ul style="list-style-type: none"> <li>Execute Test</li> <li>Run Test</li> </ul>
<b>Used By:</b>	<ul style="list-style-type: none"> <li>Analyze Test Results</li> </ul>

**Table 8 Artifact Test Report**

This section describes the 1 explicit/atomic check for Test Report (in Safety Case).

VerificationModule: Analyze Test Results	
<b>Name:</b>	Analyze Test Results
<b>Description:</b>	Checks the test results (There should not be surprises and all expected tests should be OK). It also contributes to the coveral sumary statement on the safety of the tested module(s).
<b>Comment:</b>	

Analysis of test results is done as part of the process (and it is required for unit tests), but it is not required for coverage (6.9.4.4)
<b>Qualified Name:</b>
Analyze Test Results
<b>Relevant Parameter:</b>
<ul style="list-style-type: none"> <li>MODULE = &lt;to be defined&gt;, see Table 4</li> </ul>
<b>Criteria:</b>
<ul style="list-style-type: none"> <li>Test Results Complete [AnaTR-C1]:Are the test results complete, i.e. have all required tests been executed?</li> <li>Test Results OK [AnaTR-C2]:Are the tests results OK, i.e. PASS or only failing/error for known bugs?</li> </ul>
<b>Owner (Inherited):</b>
Tester
<b>Verifies:</b>
<ul style="list-style-type: none"> <li>VerificationModule: Analyze Test Results, see Table 9</li> <li>Artifact: Test Report, see Table 8</li> </ul>
<b>VerifiedBy:</b>
<ul style="list-style-type: none"> <li>Analyze Test Results, see Table 9</li> </ul>

Table 9 VerificationModule: Analyze Test Results

### 6.3 Check for Path: ModuleTest/Artifacts/SafetyCase/Tests

This sections describes V&V of the 1 artifact located at the path  
ModuleTest/Artifacts/SafetyCase/Tests.

#### 6.3.1 Check for Artifact: Test Case

Artifact Test Case
<b>Name:</b>
Test Case
<b>Description:</b>
A module test case (implementation).
<b>Artifact Hierarchy:</b>
Test Case (in Safety Case)
<b>Path:</b>
ModuleTest/Artifacts/SafetyCase/Tests
<b>Main StakeHolder (Owner):</b>
Tester
<b>Created By:</b>
<ul style="list-style-type: none"> <li>Generate Tests</li> <li>Specify Test</li> <li>Validate Tests</li> </ul>
<b>Used By:</b>
<ul style="list-style-type: none"> <li>Execute Test</li> <li>Measure MCDC</li> <li>Run Test</li> <li>Validate Tests</li> </ul>

**Table 10 Artifact Test Case**

This section describes the 1 explicit/atomic check for Test Case (in Safety Case).

VerificationModule: Validate Tests	
<b>Name:</b>	Validate Tests
<b>Description:</b>	Tests can be validated in many ways, typically by compiling them, eventually by testing against a standard implementations from open source or reviewing them. Validas.
<b>Qualified Name:</b>	Specify Test.Validate Tests
<b>Relevant Parameter:</b>	<ul style="list-style-type: none"><li>• MODULE = &lt;to be defined&gt;, see Table 4</li></ul>
<b>Criteria:</b>	<ul style="list-style-type: none"><li>• Test Complete [VT-C1]:Does the test cover the requirement completly?</li><li>• Test Effective [VT-C2]:Can the test detect dviations (failures/errors) in case they would occur, e.g. by using some statements like "assert"?</li></ul>
<b>Compliances:</b>	<ul style="list-style-type: none"><li>• 6.9.4.4-9.1b Compliance with Branch Coverage</li><li>• 6.9.4.4-9.1c Compliance with MCDC</li></ul>
<b>Owner (Inherited):</b>	Tester
<b>Verifies:</b>	<ul style="list-style-type: none"><li>• Artifact: Test Case, see Table 10</li></ul>

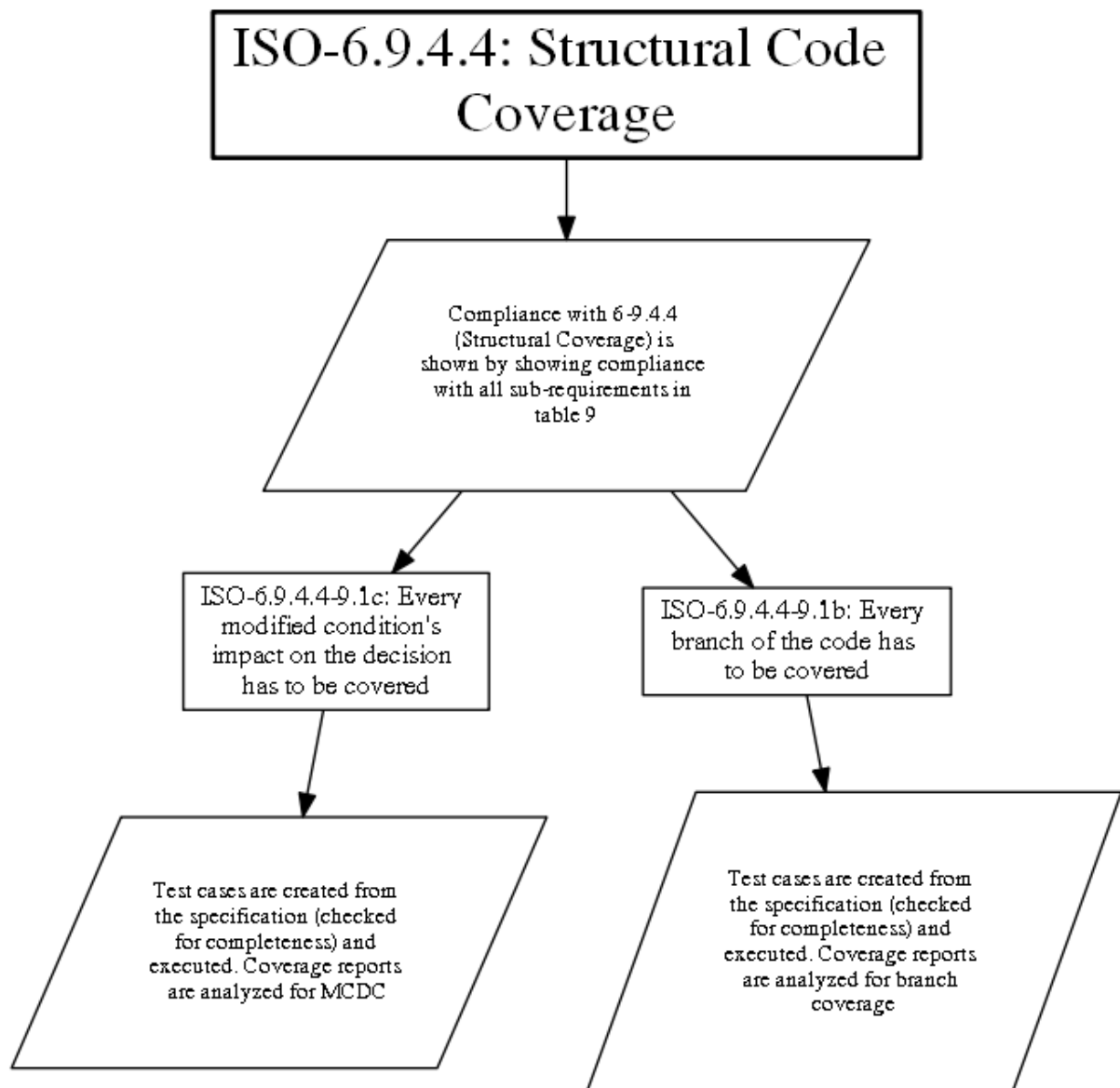
**Table 11 VerificationModule: Validate Tests**

## 7 Compliance for Perform Module Test

This section describes the compliance for Perform Module Test.

### 7.1 Compliance with Structural Code Coverage [ISO-6.9.4.4]

This section describes compliance with requirement Structural Code Coverage [ISO-6.9.4.4]. An overview on the argument is depicted in the following figure:



#### Compliance 6.9.4.4 Compliance with Structural Coverage

##### Name:

6.9.4.4 Compliance with Structural Coverage

##### Description:

Compliance with 6-9.4.4 (Structural Coverage) is shown by showing compliance with all sub-requirements in table 9.

##### Requirement:

- Structural Code Coverage [ISO-6.9.4.4], see Table 5

##### Sub-Compliances:

- 6.9.4.4-9.1b Compliance with Branch Coverage
- 6.9.4.4-9.1c Compliance with MCDC

**Table 12 Compliance 6.9.4.4 Compliance with Structural Coverage**

This section contains 0 compliance for Perform Module Test.

Skipping 1 compliance (out of scope for this document, i.e. no compliance evidence):

Skipping Table 9.



## 8 Glossary

The following abbreviations are used in the document. More information on the concepts & processes can be found in [QMeth].

- AOC: Anomalous Operating Condition
- Artifact: Element exchanged between processes
- CR: Compliance Report<sup>1</sup>
- CT: Construction Task (during QKit creation)
- KB: Known Bug
- LCR: Library Classification Report
- LQP: Library Qualification Plan
- LQR: Library Qualification Report
- LSM: Library Safety Manual
- LTG: Library Test Generator
- PCCP: (Development) Process Compliance Check Plan
- PCCR: (Development) Process Compliance Check Report
- PMT: Process Modeling Tool
- Process Module: modular tasks in the process
- PT: Preparation Task (before QKit creation)
- Role: see Stakeholder
- QKit: Qualification Kit
- QP: Qualification Plan (general), can be LQP or TQP
- QR: Qualification Report (general), can be LQR or TQR
- QST: Qualification Support Tool
- SEOOOC: Safety Element Out Of Context according to [ISO26262]
- SM: Safety Manual (general), can be LSM or TSM
- Stakeholder: abstract person taking over responsibilities in the process
- SWC: Software Component, e.g. a library<sup>2</sup>
- TAU: Test Automation Unit
- TCA: Tool Chain Analyzer
- TD: Tool Detection (part of TCL computation according to [ISO26262])
- TCL: Tool Confidence Level (according to [ISO26262])
- TCR: Tool Classification Report
- TI: Tool Impact (part of TCL computation according to [ISO26262])
- TQL: Tool Qualification Level (according to [DO330])
- TQP: Tool Qualification Plan
- TQR: Tool Qualification Report
- TSM: Tool Safety Manual
- V&V: Verification and Validation
- Verification Module: special form of Process module used to verify an artifact in the process

---

<sup>1</sup> Do not confuse with Classification Reports LCR and TCR.

<sup>2</sup> Note that libraries can be both changes and unchanged software components.



- VVP: Verification and Validation Plan
- VVR: Verification and Validation Report
- VVT: Verification and Validation Tool
- VT: Verification task (after QKit creation)

## 9 References

[DO330] RTCA. DO-330: Software Tool Qualification Considerations 1st Edition 2011-12-13.

[DO178C] RTCA. DO-178C: Software Considerations in Airborne Systems and Equipment Certification, 2011-12-13.

[EN50128] BS EN 50128:2011, Railway applications — Communication, signaling and processing systems — Software for railway control and protection systems, BSI Standards Publication

[FDA2002] General Principles of Software Validation; Final Guidance for Industry and FDA Staff, Jan 2002, from <http://www.fda.gov/downloads/MedicalDevices/DeviceRegulationandGuidance/GuidanceDocuments/ucm085371.pdf>

[FDA\_OTS] Guidance for Industry, FDA Reviewers and Compliance on Off-The-Shelf Software Use in Medical Devices, Center for Devices and Radiological Health (CDRH), from <http://www.fda.gov/downloads/MedicalDevices/DeviceRegulationandGuidance/GuidanceDocuments/ucm073779.pdf>

[IEC61508] International Electrotechnical Commission, IEC 61508, Functional safety of electrical/electronic/programmable electronic safety-related systems, Edition 2.0, Apr 2010.

[IEC62304] International Electrotechnical Commission, IEC 62304, Medical device software – Software life cycle processes

[ISO26262] International Organization for Standardization, ISO 26262 Road Vehicles – Functional safety – 2nd Edition, 2018-12.

[PR] Process Report for Perform Module Test, generated by PMT

[QMeth] Validas Qualification Method, White Paper, Version 1.7, see <Documentation>/QualificationMethodology.pdf.

[VVP] Verification and Validation Plan (Model) for Perform Module Test, generated by PMT

[VVR] Project specific Verification and Validation Report for <Product\_Name>, to be created by performing and documenting [VVP]