A blue text with a light shining on it

Description automatically generated

From Web Cryptik

**From Web Cryptik**

FIPS 140-3 Non-Proprietary Security Policy

Table of Contents

[1 General 5](#_Toc168901288)

[1.1 Overview 5](#_Toc168901289)

[1.2 Security Levels 5](#_Toc168901290)

[2 Cryptographic Module Specification 5](#_Toc168901291)

[2.1 Description 5](#_Toc168901292)

[2.2 Tested and Vendor Affirmed Module Version and Identification 6](#_Toc168901293)

[2.3 Excluded Components 6](#_Toc168901294)

[2.4 Modes of Operation 7](#_Toc168901295)

[2.5 Algorithms 7](#_Toc168901296)

[2.6 Security Function Implementations 8](#_Toc168901297)

[2.7 Algorithm Specific Information 8](#_Toc168901298)

[2.8 RBG and Entropy 8](#_Toc168901299)

[2.9 Key Generation 8](#_Toc168901300)

[2.10 Key Establishment 8](#_Toc168901301)

[2.11 Industry Protocols 8](#_Toc168901302)

[3 Cryptographic Module Interfaces 9](#_Toc168901303)

[3.1 Ports and Interfaces 9](#_Toc168901304)

[4 Roles, Services, and Authentication 9](#_Toc168901305)

[4.1 Authentication Methods 9](#_Toc168901306)

[4.2 Roles 9](#_Toc168901307)

[4.3 Approved Services 9](#_Toc168901308)

[4.4 Non-Approved Services 10](#_Toc168901309)

[4.5 External Software/Firmware Loaded 10](#_Toc168901310)

[5 Software/Firmware Security 10](#_Toc168901311)

[5.1 Integrity Techniques 10](#_Toc168901312)

[5.2 Initiate on Demand 10](#_Toc168901313)

[5.3 Open-Source Parameters [O] 10](#_Toc168901314)

[6 Operational Environment 10](#_Toc168901315)

[6.1 Operational Environment Type and Requirements 10](#_Toc168901316)

[7 Physical Security 10](#_Toc168901317)

[7.1 Mechanisms and Actions Required 10](#_Toc168901318)

[8 Non-Invasive Security 11](#_Toc168901319)

[9 Sensitive Security Parameters Management 11](#_Toc168901320)

[9.1 Storage Areas 11](#_Toc168901321)

[9.2 SSP Input-Output Methods 11](#_Toc168901322)

[9.3 SSP Zeroization Methods 11](#_Toc168901323)

[9.4 SSPs 12](#_Toc168901324)

[10 Self-Tests 12](#_Toc168901325)

[10.1 Pre-Operational Self-Tests 12](#_Toc168901326)

[10.2 Conditional Self-Tests 12](#_Toc168901327)

[10.3 Periodic Self-Test Information 13](#_Toc168901328)

[10.4 Error States 13](#_Toc168901329)

[10.5 Operator Initiation of Self-Tests 13](#_Toc168901330)

[11 Life-Cycle Assurance 13](#_Toc168901331)

[11.1 Installation, Initialization, and Startup Procedures 13](#_Toc168901332)

[11.2 Administrator Guidance 13](#_Toc168901333)

[11.3 Non-Administrator Guidance 14](#_Toc168901334)

[11.4 Design and Rules 14](#_Toc168901335)

[11.5 End of Life [O] 14](#_Toc168901336)

[12 Mitigation of Other Attacks 14](#_Toc168901337)

[12.1 Attack List [O] 14](#_Toc168901338)

[12.2 Mitigation Effectiveness [O] 14](#_Toc168901339)

[12.3 Guidance and Constraints [O] 14](#_Toc168901340)

List of Tables

**No table of figures entries found.**

List of Figures

[Figure 1: Block Diagram 7](#_Toc165464906)

# General

## Overview

This document defines the Non-Proprietary Security Policy for the *Caliptra Root of Trust for Measurement (RTM)* module, hereafter denoted the Module.

The Module is a limited operational environment under the FIPS 140-3 definitions. The Module includes a firmware load function. New firmware versions within the scope of this validation must be validated through the CMVP; any other firmware loaded into the Module is out of the scope of this validation and requires a separate FIPS 140-3 validation.

The Module meets FIPS 140-3 overall Level 1 requirements, with security levels as specified in Section 1.2.

## Security Levels

From Web Cryptik

The Module is validated to Level 1 overall with *Life-Cycle Assurance* (Section 7.11) Level 3 to allow the Module to be embedded into or bound with another validated module seeking Level 2 or 3 overall.

# Cryptographic Module Specification

## Description

Purpose and Use:

The Module is a dedicated security controller subsystem of the <vendor / module specific description>. The Module is a sub-chip subsystem (single-chip embodiment) that provides a hardware root of trust for measurement and identity.

The Module’s formal name is < vendor / module specific name>. The Module is available in the configurations shown in Table 2.

The Module design corresponds to the Module security rules. Security rules enforced by the Module are described in the appropriate context of this document.

**Module Type**: From Web Cryptik

**Module Embodiment**: From Web Cryptik

**Module Characteristics**: From Web Cryptik

Cryptographic Boundary:

The Module complies with all FIPS 140-3 IG 2.3.B *Sub-Chip Cryptographic Subsystems* requirements. The cryptographic boundary is the set of components within the dashed red line of Figure 1, inclusive of all hardware and firmware components that comprise the Module as a sub-chip subsystem.

The Module boots from an internal ROM but requires a firmware container to be loaded into RAM: during the initialization period, the loaded firmware is verified with an approved authentication method in accordance with firmware load test requirements.

The hardware module interface (HMI) is defined at the sub-chip cryptographic subsystem boundary.

The pre-operational approved integrity test is performed over all firmware components within the cryptographic boundary.

Private and secret keys cross logical and physical boundaries only in the form of <TBD>, meeting FIPS 140-3 IG 9.5.A and D.G requirements.

A diagram of a block diagram

Description automatically generated

Figure 1: Block Diagram

Tested Operational Environment’s Physical Perimeter (TOEPP):

<Specify TOEPP>

The physical form of the Module is shown in Figure 2.

Figure 2: Module Physical Form

## Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

From Web Cryptik [O]

## Excluded Components

N/A for this module.

## Modes of Operation

Modes List and Description:

From Web Cryptik

The Module as defined above will always be in an Approved mode of operation. No configuration is necessary for the Module to operate and remain in the Approved mode. The Module design corresponds to the Module security rules (see Section 11.4).

The conditions for using the Module in the Approved mode of operation are:

1. Installation of the Module as described in Section 11.1 results in the settings described below, which are required for operation in the Approved mode:
   1. <TBD>.
2. The Module is a functional block integrated into an FPGA or SoC. The integrator is responsible for:
   1. <TBD>.

Mode Change Instructions and Status [O]:

<Text>

## Algorithms

Approved Algorithms:

From Web Cryptik

Vendor-Affirmed Algorithms:

From Web Cryptik

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

From Web Cryptik

Non-Approved, Not Allowed Algorithms:

N/A for this module.

## Security Function Implementations

From Web Cryptik

## Algorithm Specific Information

<Text>

## RBG and Entropy

N/A for this module.

## Key Generation

The Module:

* Does not provide symmetric key generation.
* Produces asymmetric keys in accordance with …
* Supports symmetric key derivation in accordance with SP 800-133r2 Section 6.2, using the approved and CAVP listed KDF algorithms.

## Key Establishment

<Text>

## Industry Protocols

N/A for this module.

# Cryptographic Module Interfaces

## Ports and Interfaces

From Web Cryptik

The Control Output interface is not applicable, as the module does not control other components.

# Roles, Services, and Authentication

## Authentication Methods

N/A for this module.

## Roles

From Web Cryptik

The Module supports the mandatory Cryptographic Officer (CO) operational role only (implicitly identified) and does not support a maintenance role or a bypass capability. The CO role is assumed by meeting the conditions of Section 11.2 of this document and in associated Guidance documentation.

## Approved Services

SSP Access indicators:

* G (Generate): The Module generates or derives the SSP.
* R (Read): The SSP is read from the Module (e.g., the SSP is output).
* W (Write): The SSP is updated, imported, or written to the Module.
* E (Execute): The Module uses the SSP in performing a cryptographic operation.
* Z (Zeroize): The Module zeroizes the SSP.

From Web Cryptik

*Command frame* refers to the mailbox input registers and the associated data <…>

*Result frame* refers to the mailbox output registers and the associated data <…>

Each *command frame* and *result frame* includes a simple checksumvalue,permitting the recipientto verify parameter integrity.

Services are only operational in the running state. Any attempts to access services in any other state will result in an error being returned. If the integrity test or any CAST fails, then any attempt to access any service will result in an error being returned. The Module conforms to FIPS 140-3 IG 2.4.C *Approved Security Service Indicator*, similar to example 2. Each service provides context sensitive status responses as described in Caliptra online documentation; generally, functions of return type int return the value 1 for success with other error codes as appropriate for the call.

The *<FIPS\_NAME\_VERSION>* service response provides the means to confirm the Table 2 Module name and version information. These parameters, along with the Module’s internal indicators of the security-check and conditional-errors settings, are used to confirm the Module is the validated Module operating in the Approved mode with only Approved security services.

## Non-Approved Services

From Web Cryptik

## External Software/Firmware Loaded

<Text>

# Software/Firmware Security

## Integrity Techniques

The Module uses ECDSA signature verification (P-384, SHA2-384) as the firmware integrity method.

## Initiate on Demand

The operator can initiate the integrity test on demand by invoking the FIPS\_RUN\_PO\_TEST service.

## Open-Source Parameters [O]

<Text>

# Operational Environment

## Operational Environment Type and Requirements

**Type of Operational Environment**: From Web Cryptik

# Physical Security

## Mechanisms and Actions Required

From Web Cryptik

The Module is a single-chip embodiment that meets commercial-grade specifications for power, temperature, reliability, and shock/vibration. The Module is packaged in standard integrated circuit packaging that provides protection from probing and direct visual observation of circuit detail in the visible spectrum, as well as passivation.

# Non-Invasive Security

N/A for this module.

# Sensitive Security Parameters Management

## Storage Areas

From Web Cryptik

## SSP Input-Output Methods

From Web Cryptik

## SSP Zeroization Methods

From Web Cryptik

Keys used for CASTs and the temporary value used in the integrity test are not SSPs; however, the latter is deleted after use as required by AS05.10.

Zeroization Methods and Associated Rationales

*Zeroing on Object Destruction*

<Description>

*Hardware Register Zeroing (Power Cycle)*

<Description>

*Fuse SSP Revocation and Zeroing*

Key manifest PK and owner PK hashes permit 4 variations, each of which may be revoked in turn. Revocation status is maintained by a 4-bit one-hot revocation mask in OTP.

The revocation of a cryptographic key or an associated hash results in associated Fuse storage being zeroed.

Since by default, unprogrammed Fuse bits are read as ‘0b’, zeroed in this context requires that all zero Fuse bits in a field be programmed to ‘1b’; Fuse bits already programmed to ‘1b’ must never be attempted to be programmed to ‘1b’.

Zeroing a Fuse field is summarized as:

1. Update the associated revoked bit implemented in Fuse.
2. Read the current Fuse field that requires zeroization.
3. XOR the read field with a value of all 1s that is equivalent in length to the read field.
4. Program the Fuse field with the XORed value.

## SSPs

From Web Cryptik

From Web Cryptik

SSP Establishment During Device Manufacture or Provisioning

<G1>: Generated within the module on chip.

<IE1>: Written to OTP during device manufacture.

# Self-Tests

## Pre-Operational Self-Tests

From Web Cryptik

## Conditional Self-Tests

From Web Cryptik

All cryptographic algorithm self-tests (CASTs) must complete successfully prior to any other use of cryptography by the Module.

## Periodic Self-Test Information

From Web Cryptik

From Web Cryptik

<Text>

## Error States

From Web Cryptik

If one of the CASTs fails, the Module enters the <TBD> state. The error state is persistent, and only <TBD> services are available. All attempts to use the Module’s services result in the return of an error code (<specific error code here>). To recover from an error state, the Module must be power-cycled or reset.

## Operator Initiation of Self-Tests

The <TBD> function (inclusive of firmware integrity verification) can be called on demand, fulfilling AS05.11.

# Life-Cycle Assurance

## Installation, Initialization, and Startup Procedures

The Module is based on the open-source Caliptra RTL and Firmware. The Module is provided to vendors who integrate it into their product, typically in a manufacturing environment, and is not provided directly to US or Canadian Federal agencies. Adherence to the instructions in this document maintains security throughout the distribution, build, installation and configuration processes. Tamper is detected via the use of <TBD>. Additional Guidance inclusive of all information required per [ISO19790] Section 7.11.9 is provided by the vendor to the integrator.

## Administrator Guidance

<Text>

## Non-Administrator Guidance

N/A for this module.

## Design and Rules

The inherent properties of the Module are:

1. The Module supports only the Cryptographic Officer role, identified implicitly.
2. The Module does not support a maintenance interface or role.
3. The Module does not support authentication.
4. Power up self-tests do not require any operator action.
5. No additional interface or service is implemented by the Module which would provide access to CSPs.
6. Data output is inhibited during self-tests, zeroization, and error states.
7. The Module does not support manual key entry.
8. The Module does not output plaintext CSPs or intermediate key values.
9. Status information does not contain CSPs or sensitive data that if misused could lead to a compromise of the Module.

## End of Life [O]

<Text>

# Mitigation of Other Attacks

## Attack List [O]

The Module implements mitigations for <TBD, e.g., constant-time Implementations>. Constant-time implementations protect cryptographic implementations in the Module against timing analysis since such attacks exploit differences in execution time depending on the cryptographic operation, and constant-time implementations ensure that the variations in execution time cannot be traced back to the key, CSP or secret data.

## Mitigation Effectiveness [O]

<Text>

## Guidance and Constraints [O]

<Text>