W77F32WWAW\W77F32WQ3W

Secure Flash Memory

Security Target

Document Revision History

VERSION	DATE	DESCRIPTION
I	30-Aug-2020	Lite version
H	31-Aug-2020	Certified version

Contents

1 SECURITY TARGET INTRODUCTION 6

1.1 SECURITY TARGET REFERENCE 6
1.2 TOE REFERENCE 6
1.3 TOE OVERVIEW 6
1.3.1 TOE Type 6
1.3.2 TOE Intended Usage 6
1.3.3 Non-TOE Hardware/Software/Firmware 7
1.4 TOE DESCRIPTION 7
1.4.1 Physical Scope 7
1.4.2 Logical Scope 8

2 CONFORMANCE CLAIM 9

2.1 CC CONFORMANCE CLAIM 9
2.2 PP CLAIM 9
2.3 PACKAGE CLAIM 9

3 SECURITY PROBLEM DEFINITION 10

3.1 ASSETS 10
3.1.1 TSF data 10
3.1.2 User data 10
3.2 USERS / SUBJECTS 10
3.3 THREATS 11
3.4 ORGANISATIONAL SECURITY POLICIES 12
3.5 ASSUMPTIONS 12

4 SECURITY OBJECTIVES 13

4.1 SECURITY OBJECTIVES FOR THE TOE 13
4.2 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT 14
4.3 SECURITY OBJECTIVES RATIONALE 15
4.3.1 Threats 15
4.3.2 Assumptions 15
4.3.3 SPD and Security Objectives 16

5 EXTENDED REQUIREMENTS 18

5.1 EXTENDED FAMILIES 18
5.1.1 Extended Family FMT_LIM - Limited Capabilities and Availability 18
5.1.2 Extended Family FDP_SDC – Stored Data Confidentiality 19

6 SECURITY REQUIREMENTS 21

6.1 SECURITY FUNCTIONAL REQUIREMENTS 21
6.1.1 Malfunctions 21
6.1.2 Abuse of Functionality 21
6.1.3 Physical Manipulation and Probing 22
6.1.4 Leakage 23
6.1.5 Secure Data Exchange 23
6.1.6 Protection of Binding Key 24
6.2 SECURITY ASSURANCE REQUIREMENTS 25
6.2.1 ADV Development Documents 25
6.2.2 AGD Guidance Documents 26
6.2.3 ALC Life-cycle Support 27
6.2.4 ASE Security Target Evaluation 28
6.2.5 ATE Tests 31

6.2.6 AVA Vulnerability Assessment 32
6.3 SECURITY REQUIREMENTS RATIONALE 33
6.3.1 Objectives 33
6.3.2 Rationale Tables of Security Objectives and SFRs 34
6.3.3 Dependencies 35
6.3.4 Rationale for the Security Assurance Requirements 37

7 TOE SUMMARY SPECIFICATION 38

7.1 TOE SUMMARY SPECIFICATION 38
7.2 SFRS AND TSS 40
7.2.1 SFRs and TSS - Rationale 40

8 ANNEX 42

8.1 GLOSSARY 42
8.2 ABBREVIATIONS 42
8.3 REFERENCES 43

Table of Tables

Table 1 TOE Identification 6
Table 2: TOE Physical Scope 7
Table 3 Threats and Security Objectives - Coverage 16
Table 4 Security Objectives and Threats - Coverage 16
Table 5 Security Objectives and OSPs - Coverage 17
Table 6 Assumptions and Security Objectives for the Operational Environment - Coverage 17
Table 7 Security Objectives for the Operational Environment and Assumptions - Coverage 17
Table 8 Security Objectives and SFRs - Coverage 34
Table 9 SFRs and Security Objectives 35
Table 10 SFRs Dependencies 36
Table 11 SARs Dependencies 37
Table 12 SFRs and TSS - Coverage 41
Table 13 TSS and SFRs - Coverage 41

1 Security Target Introduction

1.1 Security Target Reference

Title: Security Target of W77F32WWAW\W77F32WQ3W Data Secure Flash Memory

Version: I

Authors: Winbond Technology Ltd.

Evaluator: Applus

Certified by: CCN Organismo de Certificacion

1.2 TOE Reference

The Target of Evaluation is identified as below:

Commercial Name	Data Secure Flash Memory
Product Name	W77F32WWAW\W77F32WQ3W
Version	B
Guidance	Operational User Guidance [6] Preparative Procedure [7] Datasheet [5] SFI Library User Guide [8]

Table 1 TOE Identification

1.3 TOE Overview

1.3.1 TOE Type

The Target of Evaluation is a Memory Flash IC.

1.3.2 TOE Intended Usage

The TOE is dedicated to be embedded into devices that will embed secure applications. The TOE is dedicated to the secure storage of the code and application's data.

The security needs for the TOE consist in:

- Maintaining the integrity of the content of the memories and the confidentiality of the content of protected memory areas as required by HW the Memory Flash is built for;
- Providing a secure communication with the Host device that will embed the TOE in a secure HW product such as Security IC.

1.3.3 Non-TOE Hardware/Software/Firmware

For the present ST, the TOE is a pure storage hardware device.

The TOE does not comprise:

a) The Host device that will embed the TOE and will be needed to use the TOE in order to stimulate the TSF;

b) SPI Bus for the communication between the Host device and the TOE;

c) SFI Library.

1.4 TOE Description

1.4.1 Physical Scope

The TOE comprises all security functionality necessary to ensure the secure execution of the Memory Flash.

NO	TYPE	IDENTIFIER	VERSION	DELIVERY METHOD
FORM OF DELIVERY : KNOWN GOOD DIE FORM				
1	HW	IC Part number	W77F32WWAW	Via Courier
FORM OF DELIVERY : ASSEMBLED DEVICE IN QFN32 PACKAGE				
1	HW	IC Part number	W77F32WQ3W	Via Courier
FORM OF DELIVERY : ASSOCIATED IC DEDICATED DOCUMENTATION				
1	PDF	Operational User Guidance [6]	Version D	Mail
2	PDF	Preparative Procedure [7]	Version F	Mail
3	PDF	SFI Library User Guide [8]	Version D	Mail
4	PDF	Datasheet [5]	Version D	Mail

Table 2: TOE Physical Scope

1.4.1.1 TOE Physical Characteristics

The TOE physical characteristics are described as follows:

- Capacity: 4M-byte
- Space-efficient Packaging: QFN32
- 16-byte burst read
- Data Integrity Check
- Program 1 to 16 byte in a single command
- Erase Suspend & Resume
- Security sensors or detectors including power glitch detector and out-of-specified operating conditions (voltage, temperature, clock frequency).

1.4.1.2 TOE Architecture

The TOE consists of the following Hardware components:

- Auxiliary array contains the flash specific data;
- Flash array stores the User data and translates SPI commands into Flash operations;
- SFF (Secure Flash Front-end) which implements encrypted interface for Flash operation and supports Flash memories up to 4GB.

1.4.1.3 Interfaces of the TOE

- The physical interface of the TOE with the external environment is the entire surface of the Memory Flash module.
- The electrical interface of the TOE with the external environment is made of the chip's pads including the data pins for SPI bus:
  - Standard SPI: CLK, /CS, DI_IO0, DO_IO1
  - Quad SPI: CLK, /CS, DI_IO0, DO_IO1, IO2, IO3.

1.4.2 Logical Scope

The main security features of the TOE are described as follows:

- Secure separation between Test mode and User mode. More precisely:
  - The switch from User mode to Test mode can only be done after completely erasing the flash content;
  - The confidentiality and the integrity of the flash content are protected in both Test mode and User mode;
- The confidentiality and the integrity of the transmitted data from/to the Host device are protected by a secure channel;
- Confidentiality protection of the flash content by memory scrambling with diversified key;
- State machine protection to counter fault injection;
- Dual Flip-Flops to counter fault injection and leakage attacks;
- Failure counter to detect and react to tamper attempts.

The logical interface of the TOE is made of Flash commands.

2 Conformance Claim

2.1 CC Conformance Claim

This Security target claims to be conformant to the Common Criteria version 3.1 Release 5. Furthermore, it claims to be CC Part 2 extended and CC Part 3 conformant.

2.2 PP Claim

This Security Target does not claim conformance to any Protection Profile.

2.3 Package Claim

The assurance level for this Security Target is EAL2.

3 Security Problem Definition

3.1 Assets

Assets include all data stored in the TOE (including executable code of the applications). They include:

- User data, that is typically stored in the "flash array" part of the memory chip;
- TSF data, that is relied upon for the enforcement of the TOE security functionality.
  - TSF data contains sensitive data stored in registers or in the auxiliary array of the memory chip;
  - The TOE does not include any software, however, the logic of the TOE security mechanisms is still part of the TSF data. This logic is hardcoded in SFF.

3.1.1 TSF data

TSF logic

The TSF logic is the functionality of the TSF, and is hardcoded in the SFF component.

The TSF logic is protected in terms of integrity and confidentiality.

Binding key (Kb)

A unique 256-bit key that is shared between the TOE and the Host.

This key is protected in terms of integrity and confidentiality.

Runtime data

The internal runtime data necessary for the execution of the SFF: session key, memory scrambling keys, Integrity Checking Engine register, stream-ciphering buffer, Failure counter, session counter, etc. All runtime data shall be protected in terms of integrity. All runtime data (except for the session counter) shall be protected in terms of confidentiality.

3.1.2 User data

User data corresponds to all data stored inside the memory flash (including executable code of the applications).

User Data

Mass data (including executable codes) stored in the "flash array" part of the memory chip. User data is protected in terms of integrity and confidentiality.

3.2 Users / Subjects

U.Host-Device

The host device communicates with the TOE through a SPI Bus.

3.3 Threats

T.Phys-Manipulation – Physical Manipulation

An attacker may physically modify the Memory Flash in order to:

- Modify *User Data* stored in the TOE;
- Modify *TSF Data* stored in the TOE;
- Modify or deactivate the security services of the TOE (provided by *TSF logic*);
- Modify the security mechanisms of the TOE (provided by *TSF logic*) to enable attacks disclosing or manipulating *User Data*.

T.Phys-Probing – Physical Probing

An attacker may perform physical probing of the TOE in order to disclose *User Data* and *TSF Data* while stored in Memory Flash.

T.Malfunction – Malfunction due to Environmental Stress

An attacker may cause a malfunction of *TSF logic* by applying environmental stress in order to deactivate or affect security mechanisms of the TOE. This enables attacks disclosing or manipulating *User Data*.

This may be achieved by operating the Memory Flash outside the normal operating conditions.

T.Abuse-Func – Abuse of Functionality

An attacker may use functions of the TOE which may not be used after TOE Delivery in order to:

- Disclose or manipulate *User Data* (user data or code stored in the TOE); or
- Enable an attack disclosing or manipulating *User Data*.

T.Leak-Inherent – Inherent Information Leakage

An attacker may exploit information which is leaked from the TOE during usage of the Memory Flash in order to disclose confidential *User Data*.

T.Leak-Forced – Forced Information Leakage

An attacker may exploit information which is leaked from the TOE during usage of the Memory Flash in order to disclose confidential *User Data* even if the information leakage is not inherent but caused by the attacker.

T.Abuse-Communication – Communication Probing and Manipulation

An attacker may probe and modify the communication between the TOE and **U.Host-Device** in order to manipulate *User/TSF Data* or disclose *User/TSF Data* read from the TOE.

T.Host-Forging – Forge the functionality of an authorized Host device

An attacker may access to the User data currently stored in the TOE by:

- Illegally establishing a secure channel with the TOE (e.g. by tampering the Binding key or by forging the secure channel without knowing the Binding key) in order to execute the Flash commands;
- Binding the TOE with another **U.Host-Device** in order to execute the Flash commands.

3.4 Organisational Security Policies

There are no Organisational Security Policies in this Security Target.

3.5 Assumptions

A.Secure-Channel – External protection during the secure channel

It is assumed that **U.Host-Device** supports the trusted communication channel with the TOE by protecting the confidentiality and the integrity of the transmitted data.

In particular, **U.Host-Device** is assumed to correctly protect the secure channel in order to prevent data modification, disclosure, insertion, deletion and replaying.

A.Binding-Process – Protection during Binding process

It is assumed that security procedures are used after delivery of the TOE by the TOE Manufacturer to maintain confidentiality and integrity of the TOE (to prevent any possible copy, modification, or unauthorized use).

This means that the binding process (i.e. generating a unique and random key Kb for **U.Host-Device** and the TOE) is assumed to be done in a secure environment where the communication between **U.Host-Device** and the TOE is protected.

Furthermore, **U.Host-Device** is assumed to provide a secure random source for generating a fresh Binding key (Kb) for the TOE.

4 Security Objectives

4.1 Security Objectives for the TOE

O.Phys-Probing – Protection against Physical Probing

The TOE must provide protection against disclosure/reconstruction of *User Data* and *TSF Data* while stored in the Flash.

This includes protection against:

- Measuring through galvanic contacts which is direct physical probing on the chips surface except on pads being bonded (using standard tools for measuring voltage and current); or
- Measuring not using galvanic contacts but other types of physical interaction between charges (using tools used in solid-state physics research and IC failure analysis) with a prior reverse-engineering to understand the design and its properties and functions.

The TOE must be designed and fabricated so that it requires a high combination of complex equipment, knowledge, skill, and time to be able to derive detailed design information or other information which could be used to compromise security through such a physical attack.

O.Malfunction – Protection against Malfunctions

The TOE must ensure its correct operation. The TOE must indicate and prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent malfunctions. Examples of environmental conditions are voltage, and clock frequency, temperature, or external energy fields.

O.Phys-Manipulation – Protection against Physical Manipulation

The TOE must provide protection against manipulation of *User Data* (the user data stored in the TOE) and *TSF data. i.e. The TOE must protect physically the integrity of the User Data and TSF data. This incl*udes protection against:

- Reverse-engineering (understanding the design and its properties and functions);
- Manipulation of the hardware and TSF data; and
- Undetected manipulation of User data (i.e. Flash array).

O.Abuse-Func – Protection against Abuse of Functionality

The TOE must prevent that functions of the TOE which may not be used after TOE Delivery can be abused in order to (i) disclose sensitive user data stored in the TOE., (ii) manipulate sensitive user data stored in the TOE.

O.Leak-Inherent – Protection against Inherent Information Leakage

The TOE must provide protection against disclosure of confidential data stored and processed in the TOE:

- By measurement and analysis of the shape and amplitude of signals (for example on the power, clock, or I/O lines); and
- By measurement and analysis of the time between events found by measuring signals (for instance on the power, clock, or I/O lines).

O.Leak-Forced – Protection against Forced Information Leakage

The TOE must be protected against disclosure of confidential data processed in the TOE (using methods as described under O.Leak-Inherent) even if the information leakage is not inherent but caused by the attacker:

- By forcing a malfunction (refer to "Protection against Malfunction due to Environmental Stress O.Malfunction"); and/or
- By a physical manipulation (refer to "Protection against Physical Manipulation - O.Phys-Manipulation").

If this is not the case, signals which normally do not contain significant information about secrets could become an information channel for a leakage attack.

O.Sec-Binding – Protection of Residual Information at Re-binding

This objective protects against the disclosure of the User data when the TOE is re-bound to another Host device.

This includes protection against:

- Integrity failure on Binding Key;
- Illegal modification on Binding Key;
- Illegal attempt to erase the Binding key.

O.Trusted-Path – Trusted Communication with Authorized Host

The TSF provides a trusted path only with authorized **U.Host-Device** (based on the shared Binding key), and protects the confidentiality and the integrity of the User/TSF data to be communicated with **U.Host-Device**.

4.2 Security Objectives for the Operational Environment

OE.Secure-Channel – Secure Communication with the TOE

The authorized **U.Host-Device** shall support the trusted communication channel with the TOE by protecting the confidentiality and the integrity of the transmitted data.

In particular, **U.Host-Device** shall correctly protect the secure channel in order to prevent data modification, disclosure, insertion, deletion and replaying.

OE.Binding-Process – Protection during Binding Process

Security procedures shall be used after the TOE delivery to maintain confidentiality and integrity of the TOE (to prevent any possible copy, modification, retention, theft or unauthorized use).

In addition, **U.Host-Device** shall provide a secure random source for generating a fresh Binding key (Kb) for the TOE.

4.3 Security Objectives Rationale

4.3.1 Threats

T.Phys-Manipulation. This threat is countered by the security objectives O.Phys-Manipulation. This objective ensures that the protection against manipulation of the user data is provided by the TOE.

T.Phys-Probing. This threat is countered by the security objectives O.Phys-Probing. This objective ensures that the protection against disclosure/reconstruction of User Data and TSF Data while stored in the Flash is provided by the TOE.

T.Malfunction. This threat is countered by the security objectives O.Malfunction. This objective ensures the correct operation of the TOE outside the normal operating conditions.

T.Abuse-Func. This threat is countered by the security objectives O.Abuse-Func. This objective prevents that functions of the TOE which may not be used after TOE Delivery can be abused in order to manipulate/disclose sensitive user data stored in the TOE.

T.Leak-Inherent. This threat is countered by the security objectives O.Leak-Inherent. This objective ensures the protection against disclosure of confidential data stored and processed in the TOE.

T.Leak-Forced. This threat is countered by the security objectives O.Leak-Forced. This objective ensures the protection against disclosure of confidential data stored and processed in the TOE even if the information leakage is not inherent but caused by the attacker.

T.Abuse-Communication. This threat is countered by the security objective O.Trusted-Path. This objective protects the confidentiality and the integrity of the User/TSF data to be communicated with U.Host-Device.

T.Host-Forging. This threat is countered by the security objectives:

- O.Trusted-Path to protect the confidentiality and the integrity of the User data to be communicated with U.Host-Device;
- O.Sec-Binding to protect against the disclosure of the User data when the TOE is re-bound to another Host device.

4.3.2 Assumptions

A.Secure-Channel. Since OE.Secure-Channel requires the Host device to implement the protection assumed in A.Secure-Channel, the assumption is covered by this objective.

A.Binding-Process. Since OE.Binding-Process requires the Composite Product Manufacturer to implement those measures assumed in A.Binding-Process, the assumption is covered by this objective.

4.3.3 SPD and Security Objectives

THREATS	SECURITY OBJECTIVES	RATIONALE
T.Phys-Manipulation	O.Phys-Manipulation	Section 4.3.1
T.Phys-Probing	O.Phys-Probing	Section 4.3.1
T.Malfunction	O.Malfunction	Section 4.3.1
T.Abuse-Func	O.Abuse-Func	Section 4.3.1
T.Leak-Inherent	O.Leak-Inherent	Section 4.3.1
T.Leak-Forced	O.Leak-Forced	Section 4.3.1
T.Abuse-Communication	O.Trusted-Path	Section 4.3.1
T.Host-Forging	O.Trusted-Path, O.Sec-Binding	Section 4.3.1

Table 3 Threats and Security Objectives - Coverage

SECURITY OBJECTIVES	THREATS
O.Phys-Probing	T.Phys-Probing
O.Malfunction	T.Malfunction
O.Phys-Manipulation	T.Phys-Manipulation
O.Abuse-Func	T.Abuse-Func
O.Leak-Inherent	T.Leak-Inherent
O.Leak-Forced	T.Leak-Forced
O.Sec-Binding	T.Host-Forging
O.Trusted-Path	T.Abuse-Communication, T.Host-Forging
OE.Secure-Channel	
OE.Binding-Process	

Table 4 Security Objectives and Threats - Coverage

SECURITY OBJECTIVES
O.Phys-Probing
O.Malfunction
O.Phys-Manipulation
O.Abuse-Func
O.Leak-Inherent
O.Leak-Forced
O.Sec-Binding
O.Trusted-Path
OE.Secure-Channel
OE.Binding-Process

Table 5 Security Objectives and OSPs - Coverage

ASSUMPTIONS	SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT	RATIONALE
A.Secure-Channel	OE.Secure-Channel	Section 4.3.2
A.Binding-Process	OE.Binding-Process	Section 4.3.2

Table 6 Assumptions and Security Objectives for the Operational Environment - Coverage

SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT	ASSUMPTIONS
OE.Secure-Channel	A.Secure-Channel
OE.Binding-Process	A.Binding-Process

Table 7 Security Objectives for the Operational Environment and Assumptions - Coverage

5 Extended Requirements

5.1 Extended Families

5.1.1 Extended Family FMT_LIM - Limited Capabilities and Availability

5.1.1.1 Description

To define the IT security functional requirements of the TOE an additional family (FMT_LIM) of the Class FMT (Security Management) is defined here. This family describes the functional requirements for the Test Features of the TOE. The new functional requirements were defined in the class FMT because this class addresses the management of functions of the TSF. The examples of the technical mechanism used in the TOE (refer to Section 6.1) appropriate to address the specific issues of preventing the abuse of functions by limiting the capabilities of the functions and by limiting their availability.

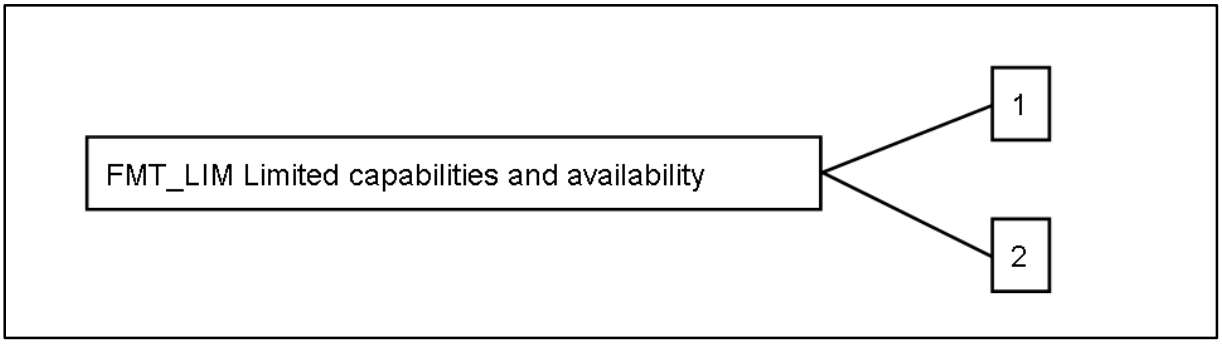
The family "Limited capabilities and availability (FMT_LIM)" is specified as follows.

FMT_LIM Limited Capabilities and Availability

Family Behavior:

This family defines requirements that limit the capabilities and availability of functions in a combined manner. Note that FDP_ACF restricts the access to functions whereas the component Limited Capability of this family requires the functions themselves to be designed in a specific manner.

Component Levelling:



FMT_LIM.1 Limited capabilities requires that the TSF is built to provide only the capabilities (perform action, gather information) necessary for its genuine purpose.

FMT_LIM.2 Limited availability requires that the TSF restrict the use of functions (refer to Limited capabilities (FMT_LIM.1)). This can be achieved, for instance, by removing or by disabling functions in a specific phase of the TOE's life-cycle.

Management: FMT_LIM.1, FMT_LIM.2

There are no management activities foreseen.

Audit: FMT_LIM.1, FMT_LIM.2

There are no actions defined to be auditable.

5.1.1.2 Extended Components

EXTENDED COMPONENT FMT_LIM.1

Description:

Limited capabilities require that the TSF is built to provide only the capabilities (perform action, gather information) necessary for its genuine purpose.

Hierarchical to: No other components.

Definition:

FMT_LIM.1 Limited Capabilities

FMT_LIM.1.1 The TSF shall be designed and implemented in a manner that limits its capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced [assignment: Limited capability policy].

Dependencies: (FMT_LIM.2)

EXTENDED COMPONENT FMT_LIM.2

Description:

Limited availability requires that the TSF restrict the use of functions (refer to Limited capabilities (FMT_LIM.1)). This can be achieved, for instance, by removing or by disabling functions in a specific phase of the TOE's life-cycle.

Hierarchical to: No other components.

Definition:

FMT_LIM.2 Limited Availability

FMT_LIM.2.1 The TSF shall be designed in a manner that limits its availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced [assignment: Limited availability policy].

Dependencies: (FMT_LIM.1).

Application Note:

The functional requirements FMT_LIM.1 and FMT_LIM.2 assume that there are two types of mechanisms (limitation of capabilities and limitation of availability) which together shall provide protection in order to enforce the same policy or two mutual supportive policies related to the same functionality. This allows, for example, that:

(i) The TSF is provided without restrictions in the product in its user environment but its capabilities are so limited that the policy is enforced or conversely;

(ii) The TSF is designed with high functionality but is removed or disabled in the product in its user environment.

5.1.2 Extended Family FDP_SDC – Stored Data Confidentiality

5.1.2.1 Description

To define the security functional requirements of the TOE an additional family (FDP_SDC.1) of the Class FDP (User data protection) is defined here.

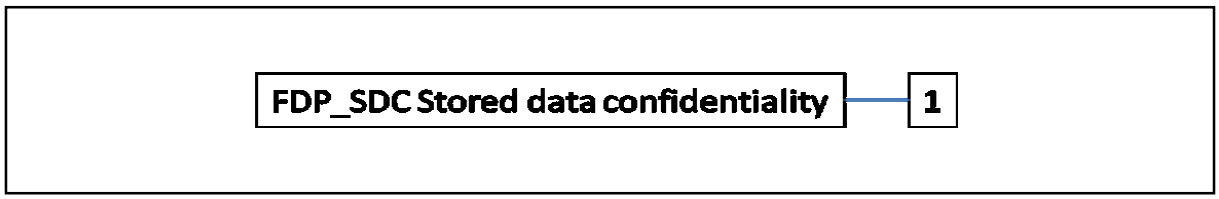
The family "Stored data confidentiality (FDP_SDC)" is specified as follows.

FDP_SDC STORED DATA CONFIDENTIALITY

Family Behavior:

This family provides requirements that address protection of user data confidentiality while these data are stored within memory areas protected by the TSF. The TSF provides access to the data in the memory through the specified interfaces only and prevents compromise of their information bypassing these interfaces.

Component Levelling:



FDP_SDC.1 Requires the TOE to protect the confidentiality of information of the user data in specified memory areas.

Management: FDP_SDC.1

There are no management activities foreseen.

Audit: FDP_SDC.1.

There are no actions defined to be auditable.

5.1.2.2 Extended Components

EXTENDED COMPONENT FDP_SDC.1

Description:

Requires the TOE to protect the confidentiality of information of the user data in specified memory areas.

Hierarchical to: No other components.

Definition:

FDP_SDC.1 Stored Data Confidentiality

FDP_SDC.1.1 The TSF shall ensure the confidentiality of the information of the user data while it is stored in the **[assignment: memory areas]**.

Dependencies: No dependencies.

6 Security Requirements

6.1 Security Functional Requirements

6.1.1 Malfunctions

FRU_FLT.2 Limited Fault Tolerance

FRU_FLT.2.1 The TSF shall ensure the operation of all the TOE's capabilities when the following failures occur: [assignment: *list of type of failures*].

The TSF shall ensure the operation of all the TOE's capabilities when the following failures occur: **exposure to operating conditions which are not detected according to the requirement Failure with preservation of secure state (FPT_FLS.1/Detectors)**.

Application Note:

The term "failure" above means "circumstances". The TOE prevents failures for the "circumstance" defined above.

FPT_FLS.1/Detectors Failure with Preservation of Secure State

FPT_FLS.1.1/Detectors The TSF shall preserve a secure state when the following types of failures occur: [assignment: *list of types of failures in the TSF*].

The TSF shall preserve a secure state when the following types of failures occur:

- **Out-of-specified range voltage**
- **Out-of-specified range temperature**
- **Out-of specified range clock frequency**
- **Power glitch.**

Application Note:

The term "failure" above means "circumstances". The TOE prevents failures for the "circumstance" defined above.

The secure state is maintained by TSF's detectors. The TSF's detectors monitor the failures. If a failure happens, the TSF disturbs the cryptographic computations, interrupts data interchange and inform **U.Host-Device**.

6.1.2 Abuse of Functionality

FMT_LIM.1 Limited Capabilities

FMT_LIM.1.1 The TSF shall be designed and implemented in a manner that limits its capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced [assignment: *Limited capability policy*].

The TSF shall be designed and implemented in a manner that limits its capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced **Deploying**

Test Features after TOE Delivery does not allow user data to be disclosed or manipulated, TSF data to be disclosed or manipulated, and no substantial information about construction of TSF to be gathered which may enable other attacks.

Application Note:

In the Test mode, the following restrictions are enforced by the TSF:

- The Binding Key (Kb) cannot be read out by the Flash commands;
- The Binding key cannot be erased unless a complete erase has been done after the last reset;
- The read and write commands do not read and write effective values of the flash array.

FMT_LIM.2 Limited Availability

FMT_LIM.2.1 The TSF shall be designed in a manner that limits its availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced [assignment: Limited availability policy].

The TSF shall be designed in a manner that limits its availability so that in conjunction with "Limited capabilities (FMT_LIM.1)" the following policy is enforced **Deploying Test Features after TOE Delivery does not allow user data to be disclosed or manipulated, TSF data to be disclosed or manipulated, and no substantial information about construction of TSF to be gathered which may enable other attacks**.

Application Note:

The switch from User mode to Test mode is allowed after TOE delivery but after the flash array is completely erased.

6.1.3 Physical Manipulation and Probing

FDP_SDC.1 Stored Data Confidentiality

FDP_SDC.1.1 The TSF shall ensure the confidentiality of the information of the user data while it is stored in the [**assignment: memory areas**].

The TSF shall ensure the confidentiality of the information of the user data while it is stored in the **Flash array**.

FPT_PHP.3 Resistance to Physical Attack

FPT_PHP.3.1 The TSF shall resist [assignment: physical tampering scenarios] to the [assignment: list of TSF devices/elements] by responding automatically such that the SFRs are always enforced.

The TSF shall resist **physical manipulation and physical probing** to the **TSF** by responding automatically such that the SFRs are always enforced.

Application Note:

The TSF will implement appropriate mechanisms to continuously counter physical manipulation and physical probing. Due to the nature of these attacks (especially manipulation) the TSF can by no means detect attacks on all of its elements. Therefore, permanent protection against these attacks is required ensuring that security functional requirements are enforced. Hence, "automatic response" means here (i) assuming that there might be an attack at any time and (ii) countermeasures are provided at any time.

6.1.4 Leakage

FDP_ITT.1 Basic Internal Transfer Protection

FDP_ITT.1.1 The TSF shall enforce the [assignment: access control SFP(s) and/or information flow control SFP(s)] to prevent the [selection: disclosure, modification, loss of use] of user data when it is transmitted between physically-separated parts of the TOE.

The TSF shall enforce the **Data Processing Policy** to prevent the **disclosure** of user data when it is transmitted between physically-separated parts of the TOE.

Application Note:

The Flash array and the SFF are seen as physically-separated parts of the TOE.

FPT_ITT.1 Basic Internal TSF Data Transfer Protection

FPT_ITT.1.1 The TSF shall protect TSF data from [selection: disclosure, modification] when it is transmitted between separate parts of the TOE.

The TSF shall protect TSF data from **disclosure** when it is transmitted between separate parts of the TOE.

Application Note:

The Flash array and the SFF are seen as physically-separated parts of the TOE.

FDP_IFC.1 Subset Information Flow Control

FDP_IFC.1.1 The TSF shall enforce the [assignment: information flow control SFP] on [assignment: list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP].

The TSF shall enforce the **Data Processing Policy** on **User data that is processed or transferred by the TOE or by U.Host-Device**.

Application Note:

The following Security Function Policy (SFP) Data Processing Policy is defined for the requirement "Subset information flow control (FDP_IFC.1)"

"User data and TSF data shall not be accessible from the TOE except when the U.Host-Device decides to communicate the User data via an external interface".

6.1.5 Secure Data Exchange

FDP_UCT.1 Basic Data Exchange Confidentiality

FDP_UCT.1.1 The TSF shall enforce the [assignment: access control SFP(s) and/or information flow control SFP(s)] to [selection: transmit, receive] user data in a manner protected from unauthorized disclosure.

The TSF shall enforce the **Data Processing Policy** to **receive and transmit** user data in a manner protected from unauthorized disclosure.

FDP_UIT.1 Data Exchange Integrity

FDP_UIT.1.1 The TSF shall enforce the [assignment: access control SFP(s) and/or information flow control SFP(s)]to [selection: receive] user data in a manner protected from [selection: modification, deletion, insertion, replay] errors.

The TSF shall enforce the **Data Processing Policy** to **receive** user data in a manner protected from **replay, modification, deletion and insertion** errors.

FDP_UIT.1.2 The TSF shall be able to determine on receipt of user data, whether [selection: modification, deletion, insertion, replay] has occurred.

The TSF shall be able to determine on receipt of user data, whether **replay, modification, deletion and insertion** has occurred.

FTP_TRP.1 Trusted Path

FTP_TRP.1.1 The TSF shall provide a communication path between itself and [selection: remote, local] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from [selection: modification, disclosure, [assignment: other types of integrity or confidentiality violation]].

The TSF shall provide a communication path between itself and **remote** users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from **modification and disclosure**.

FTP_TRP.1.2 The TSF shall permit [selection: the TSF, local users, remote users] to initiate communication via the trusted path.

The TSF shall permit **remote users** to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path for [selection: initial user authentication, [assignment: other services for which trusted path is required]].

The TSF shall require the use of the trusted path for **any access to User data stored in the Flash array**.

6.1.6 Protection of Binding Key

FPT_FLS.1/Binding_Key Failure with Preservation of Secure State

FPT_FLS.1.1/Binding_Key The TSF shall preserve a secure state when the following types of failures occur: [assignment: list of types of failures in the TSF].

The TSF shall preserve a secure state when the following types of failures occur: **integrity failure on Binding Key**.

Application Note:

The secure state is defined as follows:

- If the Binding key is illegally modified, then the TOE is locked;
- If the Binding key is erased, then the TOE User data (stored in the Flash array) is also erased.

FDP_RIP.1 Subset Residual Information Protection

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection: allocation of the resource to, deallocation of the resource from] the following objects: [assignment: list of objects].

Refinement: The TSF shall ensure that any previous information content of the Flash array is made unavailable upon the **allocation of the resource to and deallocation of the resource from** the following objects: **the Binding key (Kb)**.

Application Note:

- "Object Allocation" means that a new Binding key is set in order to replace the current Binding key.
- "Object Deallocation" means that the current Binding key is erased from the TSF (more precisely, from the auxiliary array).

6.2 Security Assurance Requirements

The Evaluation Assurance Level is EAL2.

6.2.1 ADV Development Documents

6.2.1.1 ADV_ARC Security Architecture

ADV_ARC.1 Security Architecture Description

ADV_ARC.1.1D The developer shall design and implement the TOE so that the security features of the TSF cannot be bypassed.

ADV_ARC.1.2D The developer shall design and implement the TSF so that it is able to protect itself from tampering by untrusted active entities.

ADV_ARC.1.3D The developer shall provide a security architecture description of the TSF.

ADV_ARC.1.1C The security architecture description shall be at a level of detail commensurate with the description of the SFR-enforcing abstractions described in the TOE design document.

ADV_ARC.1.2C The security architecture description shall describe the security domains maintained by the TSF consistently with the SFRs.

ADV_ARC.1.3C The security architecture description shall describe how the TSF initialization process is secure.

ADV_ARC.1.4C The security architecture description shall demonstrate that the TSF protects itself from tampering.

ADV_ARC.1.5C The security architecture description shall demonstrate that the TSF prevents bypass of the SFR-enforcing functionality.

ADV_ARC.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.1.2 ADV_FSP Functional Specification

ADV_FSP.2 Security-enforcing Functional Specification

ADV_FSP.2.1D The developer shall provide a functional specification.

ADV_FSP.2.2D The developer shall provide a tracing from the functional specification to the SFRs.

ADV_FSP.2.1C The functional specification shall completely represent the TSF.

ADV_FSP.2.2C The functional specification shall describe the purpose and method of use for all TSFI.

ADV_FSP.2.3C The functional specification shall describe all parameters associated with each

TSFI.

ADV_FSP.2.4C For each SFR-enforcing TSFI, the functional specification shall describe the SFR-enforcing actions associated with the TSFI.

ADV_FSP.2.5C For each SFR-enforcing TSFI, the functional specification shall describe direct error messages resulting from processing associated with the SFR-enforcing actions.

ADV_FSP.2.6C The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification..

ADV_FSP.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.2.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

6.2.1.3 ADV_TDS TOE Design

ADV_TDS.1 Basic Design

ADV_TDS.1.1D The developer shall provide the design of the TOE.

ADV_TDS.1.2D The developer shall provide a mapping from the TSFI of the functional specification to the lowest level of decomposition available in the TOE design.

ADV_TDS.1.1C The design shall describe the structure of the TOE in terms of subsystems.

ADV_TDS.1.2C The design shall identify all subsystems of the TSF.

ADV_TDS.1.3C The design shall describe the behavior of each SFR-supporting or SFR-non-interfering TSF subsystem in sufficient detail to determine that it is not SFR-enforcing.

ADV_TDS.1.4C The design shall summarise the SFR-enforcing behavior of the SFR-enforcing subsystems.

ADV_TDS.1.5C The design shall provide a description of the interactions among SFR-enforcing subsystems of the TSF, and between the SFR-enforcing subsystems of the TSF and other subsystems of the TSF.

ADV_TDS.1.6C The mapping shall demonstrate that all TSFIs trace to the behavior described in the TOE design that they invoke..

ADV_TDS.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence..

ADV_TDS.1.2E The evaluator shall determine that the design is an accurate and complete instantiation of all security functional requirements.

6.2.2 AGD Guidance Documents

6.2.2.1 AGD_OPE Operational User Guidance

AGD_OPE.1 Operational User Guidance

AGD_OPE.1.1D The developer shall provide operational user guidance.

AGD_OPE.1.1C The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2C The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3C The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4C The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5C The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AGD_OPE.1.6C The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7C The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.2.2 AGD_PRE Preparative Procedures

AGD_PRE.1 Preparative Procedures

AGD_PRE.1.1D The developer shall provide the TOE including its preparative procedures.

AGD_PRE.1.1C The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2C The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

AGD_PRE.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2E The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

6.2.3 ALC Life-cycle Support

6.2.3.1 ALC_CMC CM Capabilities

ALC_CMC.2 Use of a CM System

ALC_CMC.2.1D The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.2.2D The developer shall provide the CM documentation.

ALC_CMC.2.3D The developer shall use a CM system.

ALC_CMC.2.1C The TOE shall be labelled with its unique reference.

ALC_CMC.2.2C The CM documentation shall describe the method used to uniquely identify the configuration items.

ALC_CMC.2.3C The CM system shall uniquely identify all configuration items.

ALC_CMC.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.3.2 ALC_CMS CM Scope

ALC_CMS.2 Parts of the TOE CM Coverage

ALC_CMS.2.1D The developer shall provide a configuration list for the TOE.

ALC_CMS.2.1C The configuration list shall include the following: the TOE itself; the evaluation evidence required by the SARs; and the parts that comprise the TOE.

ALC_CMS.2.2C The configuration list shall uniquely identify the configuration items.

ALC_CMS.2.3C For each TSF relevant configuration item, the configuration list shall indicate the developer of the item.

ALC_CMS.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.3.3 ALC_DEL Delivery

ALC_DEL.1 Delivery Procedures

ALC_DEL.1.1D The developer shall document and provide procedures for delivery of the TOE or parts of it to the consumer.

ALC_DEL.1.2D The developer shall use the delivery procedures.

ALC_DEL.1.1C The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to the consumer.

ALC_DEL.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.4 ASE Security Target Evaluation

6.2.4.1 ASE_CCL Conformance Claims

ASE_CCL.1 Conformance Claims

ASE_CCL.1.1D The developer shall provide a conformance claim.

ASE_CCL.1.2D The developer shall provide a conformance claim rationale.

ASE_CCL.1.1C The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance.

ASE_CCL.1.2C The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended.

ASE_CCL.1.3C The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended.

ASE_CCL.1.4C The CC conformance claim shall be consistent with the extended components definition.

ASE_CCL.1.5C The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance.

ASE_CCL.1.6C The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package-augmented.

ASE_CCL.1.7C The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed.

ASE_CCL.1.8C The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed.

ASE_CCL.1.9C The conformance claim rationale shall demonstrate that the statement of security objectives is consistent with the statement of security objectives in the PPs for which conformance is being claimed.

ASE_CCL.1.10C The conformance claim rationale shall demonstrate that the statement of security requirements is consistent with the statement of security requirements in the PPs for which conformance is being claimed.

ASE_CCL.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.4.2 ASE_ECD Extended Components Definition

ASE_ECD.1 Extended Components Definition

ASE_ECD.1.1D The developer shall provide a statement of security requirements.

ASE_ECD.1.2D The developer shall provide an extended components definition.

ASE_ECD.1.1C The statement of security requirements shall identify all extended security requirements.

ASE_ECD.1.2C The extended components definition shall define an extended component for each extended security requirement.

ASE_ECD.1.3C The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.

ASE_ECD.1.4C The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.

ASE_ECD.1.5C The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated.

ASE_ECD.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ASE_ECD.1.2E The evaluator shall confirm that no extended component can be clearly expressed using existing components.

6.2.4.3 ASE_INT ST Introduction

ASE_INT.1 ST Introduction

ASE_INT.1.1D The developer shall provide an ST introduction.

ASE_INT.1.1C The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.

ASE_INT.1.2C The ST reference shall uniquely identify the ST.

ASE_INT.1.3C The TOE reference shall identify the TOE.

ASE_INT.1.4C The TOE overview shall summarise the usage and major security features of the TOE.

ASE_INT.1.5C The TOE overview shall identify the TOE type.

ASE_INT.1.6C The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.

ASE_INT.1.7C The TOE description shall describe the physical scope of the TOE.

ASE_INT.1.8C The TOE description shall describe the logical scope of the TOE.

ASE_INT.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ASE_INT.1.2E The evaluator shall confirm that the TOE reference, the TOE overview, and the TOE description are consistent with each other.

6.2.4.4 ASE_OBJ Security Objectives

ASE_OBJ.2 Security Objectives

ASE_OBJ.2.1D The developer shall provide a statement of security objectives.

ASE_OBJ.2.2D The developer shall provide a security objectives rationale.

ASE_OBJ.2.1C The statement of security objectives shall describe the security objectives for the TOE and the security objectives for the operational environment.

ASE_OBJ.2.2C The security objectives rationale shall trace each security objective for the TOE back to threats countered by that security objective and OSPs enforced by that security objective.

ASE_OBJ.2.3C The security objectives rationale shall trace each security objective for the operational environment back to threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective.

ASE_OBJ.2.4C The security objectives rationale shall demonstrate that the security objectives counter all threats.

ASE_OBJ.2.5C The security objectives rationale shall demonstrate that the security objectives enforce all OSPs.

ASE_OBJ.2.6C The security objectives rationale shall demonstrate that the security objectives for the operational environment uphold all assumptions.

ASE_OBJ.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.4.5 ASE_REQ Security Requirements

ASE_REQ.2 Derived Security Requirements

ASE_REQ.2.1D The developer shall provide a statement of security requirements.

ASE_REQ.2.2D The developer shall provide a security requirements rationale.

ASE_REQ.2.1C The statement of security requirements shall describe the SFRs and the SARs.

ASE_REQ.2.2C All subjects, objects, operations, security attributes, external entities and other terms that are used in the SFRs and the SARs shall be defined.

ASE_REQ.2.3C The statement of security requirements shall identify all operations on the security requirements.

ASE_REQ.2.4C All operations shall be performed correctly.

ASE_REQ.2.5C Each dependency of the security requirements shall either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied.

ASE_REQ.2.6C The security requirements rationale shall trace each SFR back to the security objectives for the TOE.

ASE_REQ.2.7C The security requirements rationale shall demonstrate that the SFRs meet all security objectives for the TOE.

ASE_REQ.2.8C The security requirements rationale shall explain why the SARs were chosen.

ASE_REQ.2.9C The statement of security requirements shall be internally consistent.

ASE_REQ.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.4.6 ASE_SPD Security Problem Definition

ASE_SPD.1 Security Problem Definition

ASE_APD.1.1D The developer shall provide a security problem definition.

ASE_SPD.1.1C The security problem definition shall describe the threats.

ASE_SPD.1.2C All threats shall be described in terms of a threat agent, an asset, and an adverse action.

ASE_SPD.1.3C The security problem definition shall describe the OSPs.

ASE_SPD.1.4C The security problem definition shall describe the assumptions about the operational environment of the TOE.

ASE_SPD.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.4.7 ASE_TSS TOE Summary Specification

ASE_TSS.1 TOE Summary Specification

ASE_TSS.1.1D The developer shall provide a TOE summary specification.

ASE_TSS.1.1C The TOE summary specification shall describe how the TOE meets each SFR.

ASE_TSS.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ASE_TSS.1.2E The evaluator shall confirm that the TOE summary specification is consistent with the TOE overview and the TOE description.

6.2.5 ATE Tests

6.2.5.1 ATE_COV Coverage

ATE_COV.1 Evidence of Coverage

ATE_COV.1.1D The developer shall provide evidence of the test coverage.

ATE_COV.1.1C The analysis of the test coverage shall demonstrate the correspondence between the tests in the test documentation and the TSFIs in the functional specification.

ATE_COV.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.5.2 ATE_FUN Functional Tests

ATE_FUN.1 Functional Testing

ATE_FUN.1.1D The developer shall test the TSF and document the results.

ATE_FUN.1.2D The developer shall provide test documentation.

ATE_FUN.1.1C The test documentation shall consist of test plans, expected test results and actual test results.

ATE_FUN.1.2C The test plans shall identify the tests to be performed and describe the scenarios for performing each test. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.3C The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.4C The actual test results shall be consistent with the expected test results.

ATE_FUN.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.5.3 ATE_IND Independent Testing

ATE_IND.2 Independent Testing – Sample

ATE_IND.2.1D The developer shall provide the TOE for testing.

ATE_IND.2.1C The TOE shall be suitable for testing.

ATE_IND.2.2C The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

ATE_IND.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.2.2E The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

ATE_IND.2.3E The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

6.2.6 AVA Vulnerability Assessment

6.2.6.1 AVA_VAN Vulnerability Analysis

AVA_VAN.2 Vulnerability analysis

AVA_VAN.2.1D The developer shall provide the TOE for testing**.**

AVA_VAN.2.1C The TOE shall be suitable for testing**.**

AVA_VAN.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.2.2E The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.2.3E The evaluator shall perform an independent vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design and security architecture description to identify potential vulnerabilities in the TOE.

AVA_VAN.2.4E The evaluator shall conduct penetration testing based on the identified potential vulnerabilities to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6.3 Security Requirements Rationale

6.3.1 Objectives

6.3.1.1 Security Objectives for the TOE

O.Phys-Probing. The SFR FDP_SDC.1 requires the TSF to protect the confidentiality of the user data stored in specified memory areas and prevent its compromise by physical attacks bypassing the specified interfaces for memory access. The scenario of physical probing as described for this objective is explicitly included in the assignment chosen for the physical tampering scenarios in FPT_PHP.3. Therefore, it is clear that this security functional requirement supports the objective.

O.Malfunction. The definition of this objective shows that it covers a situation, where malfunction of the TOE might be caused by the operating conditions of the TOE (while direct manipulation of the TOE is covered O.Phys-Manipulation). There are two possibilities in this situation: Either the operating conditions are inside the tolerated range or at least one of them is outside of this range. The second case is covered by FPT_FLS.1/Detectors, because it states that a secure state is preserved in this case. The first case is covered by FRU_FLT.2 because it states that the TOE operates correctly under normal (tolerated) conditions.

O.Phys-Manipulation. The scenario of physical manipulation as described for this objective is explicitly included in the assignment chosen for the physical tampering scenarios in FPT_PHP.3. Therefore, it is clear that this security functional requirement supports the objective.

O.Abuse-Func. This objective states that abuse of functions (especially provided by the IC Dedicated Test Software, for instance in order to read secret data) must not be possible when TOE is used by the final user. There are two possibilities to achieve this: (i) They cannot be used by an attacker (i. e. its availability is limited) or (ii) using them would not be of relevant use for an attacker (i. e. its capabilities are limited) since the functions are designed in a specific way. The first possibility is specified by FMT_LIM.2 and the second one by FMT_LIM.1. Since these requirements are combined to support the policy, which is suitable to fulfil O.Abuse-Func, both security functional requirements together are suitable to meet the objective. Other security functional requirements (FPT_ITT.1, FDP_ITT.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1/Detectors and FDP_IFC.1) which prevent attackers from circumventing the functions implementing these two security functional requirements (for instance by manipulating the hardware) also support the objective. The relevant objectives are O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced.

O.Leak-Inherent. The refinements of the security functional requirements FPT_ITT.1 and FDP_ITT.1 together with the policy statement in FDP_IFC.1 explicitly require the prevention of disclosure of secret data (TSF data as well as user data) when while being processed. This includes that attackers cannot reveal such data by measurements of emanations, power

consumption or other behavior of the TOE while data is processed by TOE parts.

O.Leak-Forced. This objective is directed against attacks, where an attacker wants to force an information leakage, which would not occur under normal conditions. In order to achieve this the attacker has to combine a first attack step, which modifies the behavior of the TOE (either by exposing it to extreme operating conditions or by directly manipulating it) with a second attack step measuring and analysing some output produced by the TOE. The first step is prevented by the same mechanisms which support O.Malfunction (FPT_FLS.1/Detectors, FRU_FLT.2) and O.Phys-Manipulation (FPT_PHP.3), respectively. The requirements covering O.Leak-Inherent (FPT_ITT.1, FDP_ITT.1, FDP_IFC.1) also support O.Leak-Forced because they prevent the attacker from being successful if he tries the second step directly.

O.Sec-Binding. The security functional requirement FDP_RIP.1 ensures that the User data is erased before the Host device is changed.

O.Trusted-Path The security functional requirement FTP_TRP.1 contribute in this protection because it only establishes a trusted path between the TSF and authorized **U.Host-Device** for the communication purpose.

The security functional requirement FPT_FLS.1/Binding_Key protects the Binding key against the tampering.

The security functional requirements FDP_UCT.1 and FDP_UIT.1 protect against the modification (integrity) and the disclosure (confidentiality) of the User data communication between the TSF and **U.Host-Device**.

6.3.2 Rationale Tables of Security Objectives and SFRs

SECURITY OBJECTIVES	SECURITY FUNCTIONAL REQUIREMENTS	RATIONALE
O.Phys-Probing	FPT_PHP.3, FDP_SDC.1	Section 6.3.1
O.Malfunction	FRU_FLT.2, FPT_FLS.1/Detectors	Section 6.3.1
O.Phys-Manipulation	FPT_PHP.3	Section 6.3.1
O.Abuse-Func	FDP_ITT.1, FPT_ITT.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1/Detectors, FMT_LIM.1, FMT_LIM.2, FDP_IFC.1	Section 6.3.1
O.Leak-Inherent	FDP_ITT.1, FPT_ITT.1, FDP_IFC.1	Section 6.3.1
O.Leak-Forced	FDP_ITT.1, FPT_ITT.1, FRU_FLT.2, FPT_FLS.1/Detectors, FPT_PHP.3, FDP_IFC.1	Section 6.3.1
O.Sec-Binding	FDP_RIP.1	Section 6.3.1
O.Trusted-Path	FDP_UCT.1, FDP_UIT.1, FPT_FLS.1/Binding_Key, FTP_TRP.1	Section 6.3.1

Table 8 Security Objectives and SFRs - Coverage

SECURITY FUNCTIONAL REQUIREMENTS	SECURITY OBJECTIVES
FRU_FLT.2	O.Malfunction, O.Abuse-Func, O.Leak-Forced
FPT_FLS.1/Detectors	O.Malfunction, O.Abuse-Func, O.Leak-Forced

SECURITY FUNCTIONAL REQUIREMENTS	SECURITY OBJECTIVES
FMT_LIM.1	O.Abuse-Func
FMT_LIM.2	O.Abuse-Func
FDP_SDC.1	O.Phys-Probing
FPT_PHP.3	O.Phys-Probing, O.Phys-Manipulation, O.Abuse-Func, O.Leak-Forced
FDP_ITT.1	O.Abuse-Func, O.Leak-Inherent, O.Leak-Forced
FPT_ITT.1	O.Abuse-Func, O.Leak-Inherent, O.Leak-Forced
FDP_IFC.1	O.Abuse-Func, O.Leak-Inherent, O.Leak-Forced
FDP_UCT.1	O.Trusted-Path
FDP_UIT.1	O.Trusted-Path
FTP_TRP.1	O.Trusted-Path
FPT_FLS.1/Binding_Key	O.Trusted-Path
FDP_RIP.1	O.Sec-Binding

Table 9 SFRs and Security Objectives

6.3.3 Dependencies

6.3.3.1 SFRs Dependencies

REQUIREMENTS	CC DEPENDENCIES	SATISFIED DEPENDENCIES
FRU_FLT.2	(FPT_FLS.1)	FPT_FLS.1/Detectors
FPT_FLS.1/Detectors	No Dependencies	
FMT_LIM.1	(FMT_LIM.2)	FMT_LIM.2
FMT_LIM.2	(FMT_LIM.1)	FMT_LIM.1
FDP_SDC.1	No Dependencies	
FPT_PHP.3	No Dependencies	
FDP_ITT.1	(FDP_ACC.1 or FDP_IFC.1)	FDP_IFC.1
FPT_ITT.1	No Dependencies	
FDP_IFC.1	(FDP_IFF.1)	
FDP_UCT.1	(FDP_ACC.1 or FDP_IFC.1) and (FTP_ITC.1 or FTP_TRP.1)	FDP_IFC.1, FTP_TRP.1
FDP_UIT.1	(FDP_ACC.1 or FDP_IFC.1) and (FTP_ITC.1 or FTP_TRP.1)	FDP_IFC.1, FTP_TRP.1
FTP_TRP.1	No Dependencies	
FPT_FLS.1/Binding_Key	No Dependencies	

REQUIREMENTS	CC DEPENDENCIES	SATISFIED DEPENDENCIES
FDP_RIP.1	No Dependencies	

Table 10 SFRs Dependencies

Rationale for the Exclusion of Dependencies

The dependency FDP_IFF.1 of FDP_IFC.1 is discarded. Part 2 of the Common Criteria defines the dependency of FDP_IFC.1 (information flow control policy statement) on FDP_IFF.1 (Simple security attributes). The specification of FDP_IFF.1 would not capture the nature of the security functional requirement nor add any detail.

As stated in the Data Processing Policy referred to in FDP_IFC.1, there are no attributes necessary. The security functional requirement for the TOE is sufficiently described using FDP_ITT.1 and its Data Processing Policy (FDP_IFC.1).

6.3.3.2 SARs Dependencies

REQUIREMENTS	CC DEPENDENCIES	SATISFIED DEPENDENCIES
ADV_ARC.1	(ADV_FSP.1) and (ADV_TDS.1)	ADV_FSP.2, ADV_TDS.1
ADV_FSP.2	ADV_TDS.1	ADV_TDS.1
ADV_TDS.1	(ADV_FSP.2)	ADV_FSP.2
AGD_OPE.1	(ADV_FSP.1)	ADV_FSP.2
AGD_PRE.1	No Dependencies	
ALC_CMC.2	ALC_CMS.1	ALC_CMS.2
ALC_CMS.2	No Dependencies	
ALC_DEL.1	No Dependencies	
ASE_CCL.1	(ASE_ECD.1) and (ASE_INT.1) and (ASE_REQ.1)	ASE_ECD.1, ASE_INT.1, ASE_REQ.2
ASE_ECD.1	No Dependencies	
ASE_INT.1	No Dependencies	
ASE_OBJ.2	(ASE_SPD.1)	ASE_SPD.1
ASE_REQ.2	(ASE_ECD.1) and (ASE_OBJ.2)	ASE_ECD.1, ASE_OBJ.2
ASE_SPD.1	No Dependencies	
ASE_TSS.1	(ADV_FSP.1) and (ASE_INT.1) and (ASE_REQ.1)	ADV_FSP.2, ASE_INT.1, ASE_REQ.2
ATE_COV.1	(ADV_FSP.2) and (ATE_FUN.1)	ADV_FSP.2, ATE_FUN.1
ATE_FUN.1	(ATE_COV.1)	ATE_COV.1
ATE_IND.2	(ADV_FSP.2) and (AGD_OPE.1) and (AGD_PRE.1) and (ATE_COV.1) and (ATE_FUN.1)	ADV_FSP.2, AGD_OPE.1, AGD_PRE.1, ATE_COV.1, ATE_FUN.1
AVA_VAN.2	(ADV_ARC.1) and (ADV_FSP.2) and (ADV_TDS.1) and (AGD_OPE.1) and (AGD_PRE.1)	ADV_ARC.1, ADV_FSP.2, ADV_TDS.1, AGD_OPE.1, AGD_PRE.1,

Table 11 SARs Dependencies

6.3.4 Rationale for the Security Assurance Requirements

These SARs have been chosen to meet the market needs of a Secure Flash with resistance to attacks performed by an attacker possessing Basic attack potential.

7 TOE Summary Specification

This Chapter describes the TSF security functionality by a set of security features and justifies how the SFR defined in Chapter 6 are enforced by those features.

7.1 TOE Summary Specification

SF.SEC-COM – Secure Communication

SF.SEC-COM protects the confidentiality and the integrity of the communication between the TOE and **U.Host-Device** against probing, Man-in-the-Middle, hammering and replay attacks. In particular:

- A fresh session key is used for each session;
- There is an encryption key produced for each transaction and this key depends on the Session Key. A unique Transaction Counter is used to prevent replay attacks too;
- In order to avoid key repetition, the TOE implements counters like a non-volatile Session Counter and a Transaction Counter;
- Session and transaction counters are also used to protect against replaying.

SF.SEC-COM is devised to enable in-place execution of the code stored in the TOE. For this purpose, each data-word sent by TOE is separately encrypted by applying a cascade of a SHA based stream ciphering operation that cryptographically maps input bits to output bits.

SF.PHY-PRO – Physical Protection

SF.PHY-PRO protects the TOE against physical manipulation (including the TOE probing). SF.PHY-PRO includes the following security mechanisms:

- Failure counter: this counter is incremented after each tamper-detection and the TOE is locked if the counter reaches a pre-defined value;
- Dual flip-flops: A difference in the state of two joint flip-flops indicates a fault and raises the Fault Injection Alarm output signal. This mechanism is designed to detect perturbation attacks like Laser or Electro-Magnetic fault injections;
- Clock-tree protection: The 0-1 pattern spreads in a dedicated shift register with every clock pulse provided all clock signals are active. This mechanism is designed to ensure that the clock-tree is intact;
- State machine monitoring: The TOE implements Tamper Detectors that detects abnormal conditions and reports a fault state;
- The SHA Module implements a mode with better protection against leakage attacks. This mode is used for sensitive calculations in Session Setup to prevent leakage of the Binding Key.

Note: Integrity of the flash content is not achieved by the implementation of a TOE security functionality specifically oriented for this feature. Instead, the integrity of the stored data is ensured indirectly by the physical protection mechanisms and the technology inherent properties.

SF.PHY-PRO also protects the TOE against the inherent or intentional leak of the TOE operations by the following security mechanisms:

- Advanced stream cipher using long linear feedback shift registers: the calculations are protected against timing and power consumption leak;
- Anti-Leakage measures for the hash functions that are used for stream-ciphering and MAC digest: masking input data and undisclosure of intermediate output values;
- Session setup: the logic is protected against timing and power consumption leak.

SF.OPE-MODE – Control of Operating Modes

SF.OPE-MODE ensures that the User Data is not disclosed or manipulated via the features available in the TEST mode.

In particular, the Flash array is completely erased before switching to TEST mode. Furthermore, the access to User data is also restricted in the Test mode. More precisely:

- The Binding Key (Kb) cannot be read out by the Flash commands;
- The Binding key cannot be erased unless a complete erase has been done after the last reset;
- The read and write commands do not read and write effective values of the Flash array.

SF.OPE-COND – Control of Operating Conditions

SF.OPE-COND detects the abnormal operation conditions (voltage, temperature, clock frequency, power glitch) using the corresponding sensors.

If an abnormal operation condition happens, then SF.OPE-COND disturbs the cryptographic computations, interrupts data interchange and inform **U.Host-Device**.

SF.SEC-MEM-CONF – Storage Confidentiality

SF.SEC-MEM-CONF protects the confidentiality of the User Data stored in the flash array by a memory scrambling mechanism that is based on diversified keys. Both the addresses and the memory content are scrambled but by a key that is unique for each instance of the TOE.

SF.KEY-PRO – Protection of Binding Key

SF.KEY-PRO protects the User data against disclosure by manipulating the binding key. In particular, the Flash array is completely erased before:

- A new Binding key is set; or
- The current Binding key is erased.

Furthermore, the current Binding key is stored in the Auxiliary array and cannot be read out by the Flash commands. The integrity of the Binding key is protected by a digest value: if an illegal modification is detected on the Binding key, then the TOE is locked and can only be unlocked in Test mode (and the Flash array has been erased).

SF.SEC-AUTH – Secure Authentication

SF.SEC-AUTH ensures that only an authorized Host device (i.e. a Host device that knows the Binding key Kb) can open a secure channel to communicate with the TOE.

More precisely, SF.SEC-AUTH provides a mutual authentication between the Host device and the TOE by verifying that both of them share the same Binding key. A failed authentication increases the Failure counter: if this counter reaches a pre-defined value, then the TOE is locked.

7.2 SFRs and TSS

7.2.1 SFRs and TSS - Rationale

7.2.1.1 TOE Summary Specification

SF.SEC-COM enforces the FDP_UCT.1 and FDP_UIT.1 because the User Data is protected while being transmitted to **U.Host-Device**. SF.SEC-COM enforces the FDP_IFC.1 in particular the user data is protected in terms of confidentiality when being transferred by the TOE to **U.Host-Device**. Moreover, the user data is protected in terms of integrity during the communication between the TOE and **U.Host-Device**.

SF.PHY-PRO enforces the TOE resistance against physical attacks (FPT_PHP.3). SF.PHY-PRO contributes to the confidentiality protection of the User data stored in the TOE (FDP_SDC.1); the cryptographic services are also protected against the physical attacks. SF.PHY-PRO protects against some attacks on the cryptographic services used for the transmission of the User data (FPT_ITT.1, FDP_ITT.1 and FDP_IFC.1).

SF.OPE-MODE enforces the restriction of the TSF capabilities and availability during the deployment of the test features after the TOE delivery (respectively FMT_LIM.1 and FMT_LIM.2).

SF.OPE-COND enforces the TOE fault-tolerance and fail-secure (respectively FRU_FLT.2 and FPT_FLS.1/Detectors).

SF.SEC-MEM-CONF By definition, SF.SEC-MEM-CONF enforces FDP_SDC.1. SF.SEC-MEM-CONF also enforces the FDP_IFC.1 in particular the User data and TSF data are protected in terms of confidentiality when being stored, processed or transferred between two TOE components (SFF and Flash array).

SF.KEY-PRO enforces FDP_RIP.1 because it erases the Flash content before a new Binding key is set or the current Binding key is erased. SF.KEY-PRO also detects the failure and put the TOE in a secure state (i.e. locked state) due to an illegal modification of the current Binding key. In other words, SF.BIND-KEY-PRO enforces FPT_FLS.1/Binding_Key.

SF.SEC-AUTH enforces the FTP_TRP.1 because only an authorized **U.Host-Device** can open a trusted channel with the TOE.

ASSOCIATION TABLES OF SFRS AND TSSECURITY FUNCTIONAL REQUIREMENTS	TOE SUMMARY SPECIFICATION
FRU_FLT.2	SF.OPE-COND
FPT_FLS.1/Detectors	SF.OPE-COND
FMT_LIM.1	SF.OPE-MODE
FMT_LIM.2	SF.OPE-MODE
FDP_SDC.1	SF.PHY-PRO, SF.SEC-MEM-CONF
FPT_PHP.3	SF.PHY-PRO
FDP_ITT.1	SF.PHY-PRO
FPT_ITT.1	SF.PHY-PRO
FDP_IFC.1	SF.SEC-MEM-CONF, SF.SEC-COM, SF.PHY-PRO
FDP_UCT.1	SF.SEC-COM
FDP_UIT.1	SF.SEC-COM
FTP_TRP.1	SF.SEC-AUTH
FPT_FLS.1/Binding_Key	SF.KEY-PRO
FDP_RIP.1	SF.KEY-PRO

Table 12 SFRs and TSS - Coverage

TOE SUMMARY SPECIFICATION	SECURITY FUNCTIONAL REQUIREMENTS
SF.SEC-COM	FDP_IFC.1, FDP_UCT.1, FDP_UIT.1
SF.PHY-PRO	FDP_SDC.1, FPT_PHP.3, FDP_ITT.1, FPT_ITT.1, FDP_IFC.1
SF.OPE-MODE	FMT_LIM.1, FMT_LIM.2
SF.OPE-COND	FRU_FLT.2, FPT_FLS.1/Detectors
SF.SEC-MEM-CONF	FDP_SDC.1, FDP_IFC.1
SF.KEY-PRO	FPT_FLS.1/Binding_Key, FDP_RIP.1
SF.SEC-AUTH	FTP_TRP.1

Table 13 TSS and SFRs - Coverage

8 Annex

8.1 Glossary

SFI – Secure Flash Interface is the SPI interface on the Host device (i.e. SPI Master).

SFF – Secure Flash Front-end is the SPI interface on the memory chip (i.e. SPI Slave).

SPI – Serial Peripheral Interface is a synchronous serial data link, a *de facto* standard, that operates in full duplex mode.

8.2 Abbreviations

CC	Common Criteria
EAL	Evaluation Assurance Level
IT	Information Technology
PP	Protection Profile
SFI	Secure Flash Interface
SFF	Secure Flash Front-end
SPI	Serial Peripheral Interface
ST	Security Target
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Functionality
TSFI	TSF Interface
TSP	TOE Security Policy

8.3 References

[1] Common Criteria, *Part 1: Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model,* Version 3.1, Revision 5, April 2017, CCMB-2017-04-001

[2] Common Criteria, *Part 2: Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components*, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002

[3] Common Criteria, *Part 3: Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components*, Version 3.1, Revision 5, April 2017, CCMB-2017-03-003

[4] *Common Methodology for Information Technology Security Evaluation, Evaluation Methodology*, Version 3.1, Revision 5, April 2017, CCMB-2017-04-004

[5] Winbond Technology Ltd., W77F32W Secure Flash Datasheet

[6] Winbond Technology Ltd., W77F32W Operational User Guide

[7] Winbond Technology Ltd., W77F32W Preparative User Guide

[8] Winbond Technology Ltd., SFI Library User Guide