NXP Secure Smart Card Controller P61N1M3PVD/VD-1/VE-1

Security Target Lite

Rev. 2.11 — 10 January 2019
BSI-DSZ-CC-1051

Evaluation documentation
PUBLIC

Document information

Info	Content
Keywords	CC, Security Target Lite, P61N1M3PVD/VD-1/VE-1
Abstract	Security Target Lite of the NXP Secure Smart Card Controller P61N1M3PVD/VD-1/VE-1, which is developed and provided by NXP Semiconductors, Business Unit Identification according to the Common Criteria for Information Technology Security Evaluation Version 3.1 at Evaluation Assurance Level 6 augmented.

Revision history

Rev	Date	Description
0.1	20120604	Initial Revision, derived from P61N1M3 ST Document
1.0	20130111	evaluation draft
2.5	20140827	rework based on comments from HW evaluator
2.6	20170901	Conformance claim to PP0084 and Common Criteria v3.1 Release 5
2.7	20180305	Update regarding memory read/write operation in User Mode
2.8	20180323	Add the errata sheet as additional guidance documentation
2.9	20180913	Update memory access control policy to be consistent with MMU errata sheet
2.10	20181113	Add references to UGM and Errata Sheet in Logical Scope Further refinement to access control policy
2.11	20190110	Remove CBC mode from FCS.COP.1/TDES and FCS_COP.1/AES

Contact information

For additional information, please visit: http://www.nxp.com

For sales office addresses, please send an email to: salesaddresses@nxp.com

1. ST Introduction

This chapter is divided into the following sections: “ST Reference”, “TOE Reference”, “TOE Overview” and “TOE Description”.

1.1 ST Reference

“NXP Secure Smart Card Controller P61N1M3PVD/VD-1/VE-1, Security Target Lite, NXP Semiconductors, Business Unit Identification, Rev. 2.11, 10 January 2019”

1.2 TOE Reference

The TOE is named NXP Secure Smart Card Controller P61N1M3PVD/VD-1/VE-1 including IC Dedicated Software. The TOE is delivered as base type P61N1M3VD or as base type P61N1M3VD-1 or as base type P61N1M3VE-1, and each base type in one major configuration only. In short form the TOE is named P61N1M3PVD/VD-1/VE-1.

1.3 TOE Overview

1.3.1 Usage and major security functionality of the TOE

The TOE is the IC hardware platform P61N1M3PVD/VD-1/VE-1 with IC Dedicated Software and documentation describing instruction set and usage of the TOE. The TOE does not include a customer-specific Security IC Embedded Software.

The IC hardware is a microcontroller incorporating a central processing unit, memories accessible via a Memory Management Unit, cryptographic coprocessors, other security components and several electrical communication interfaces. The central processing unit supports a 32-/24-/16-/8-bit instruction set optimized for smartcard applications, which is a superset of the 80C51 family instruction set. The first and in some cases the second byte of an instruction are used for operation encoding. On-chip memories are ROM, Flash, EEPROM and RAM. The ROM is reserved for IC Dedicated Software. Flash and EEPROM can be used by the Security IC Embedded Software for code and data. They consist of high reliable memory cells, which guarantee data integrity. Flash and EEPROM are optimised for applications that require reliable non-volatile data storage for data and program code. Dedicated security functionality protects the contents of all memories.

The IC Dedicated Software comprises IC Dedicated Test Software for test purposes and IC Dedicated Support Software. The IC Dedicated Support Software consists of the Boot-ROM Software, which controls the boot process of the hardware platform, the Firmware Operating System and the Bootloader Software. The Firmware Operating System serves a hardware control interface for Flash and EEPROM, which is accessible to the Security IC Embedded Software. The Bootloader Software is not accessible to the Security IC Embedded Software.

CPU, memory and hardware management of P61N1M3PVD/VD-1/VE-1 support the Security IC Embedded Software to implement an operating system with multiple applications to one platform.

The security functionality of the TOE is designed to act as an integral part of a complete security system in order to strengthen the design as a whole. Several security mechanisms are completely implemented in and controlled by the TOE. Other security mechanisms allow for configuration by or even require support of the Security IC Embedded Software.

P61N1M3PVD/VD-1/VE-1 provides high security for smartcard applications and in particular for being used in the banking and finance market, in electronic commerce or in governmental applications. Hence P61N1M3PVD/VD-1/VE-1 maintains

- integrity, correctness and confidentiality of its security functionality,
- integrity and confidentiality of data and code stored to its memories,
- CPU modes with specific access to memories and hardware resources.

This is ensured by the construction of P61N1M3PVD/VD-1/VE-1 and its security functionality.

P61N1M3PVD/VD-1/VE-1 basically provides

- hardware to calculate Data Encryption Standard (Triple-DES) with up to three keys,
- hardware to calculate Advanced Encryption Standard (AES) with different key lengths,
- hardware support for large integer arithmetic operations like multiplication, addition and logical operations, which are suitable for public key cryptography and elliptic curve cryptography,
- a True Random Number Generator,
- a Memory Management Unit,
- hardware to calculate Cyclic Redundancy Check (CRC),
- an ISO/IEC 7816 compliant interface,
- a Serial Peripheral Interface (SPI),
- a Single Wire Protocol (SWP) interface with ETSI TS 102 613 protocol,
- an S^2C interface with ISO/IEC 14443 protocol.

In addition, several security mechanisms are implemented to ensure proper operation as well as integrity and confidentiality of stored data. For example, this includes security mechanisms for memory protection and security exceptions as well as sensors, which allow operation under specified conditions only.

Note: Large integer arithmetic operations are intended to be used for calculation of asymmetric cryptographic algorithms. Any asymmetric cryptographic algorithm utilising the support for large integer arithmetic operations has to be implemented to the Security IC Embedded Software. The support for large integer arithmetic operations does not provide security functionality like cryptography. The Security IC Embedded Software that implements an asymmetric cryptographic algorithm is not included in this Security Target, but the support for large integer arithmetic operations is a security relevant component of the TOE, which resists to the attacks mentioned in this Security Target and operates correctly as specified in the data sheet. The same scope is applied to the CRC calculation.

1.3.2 TOE type

NXP Secure Smart Card Controller P61N1M3PVD/VD-1/VE-1 is provided as an IC hardware platform with IC Dedicated Software for various operating systems and applications with high demands on security.

1.3.3 Required non-TOE hardware/software/firmware

None

1.4 TOE Description

1.4.1 Physical Scope of TOE

P61N1M3PVD/VD-1/VE-1 is manufactured in an advanced 90nm CMOS technology. A block diagram of the IC hardware is depicted in Fig 1.

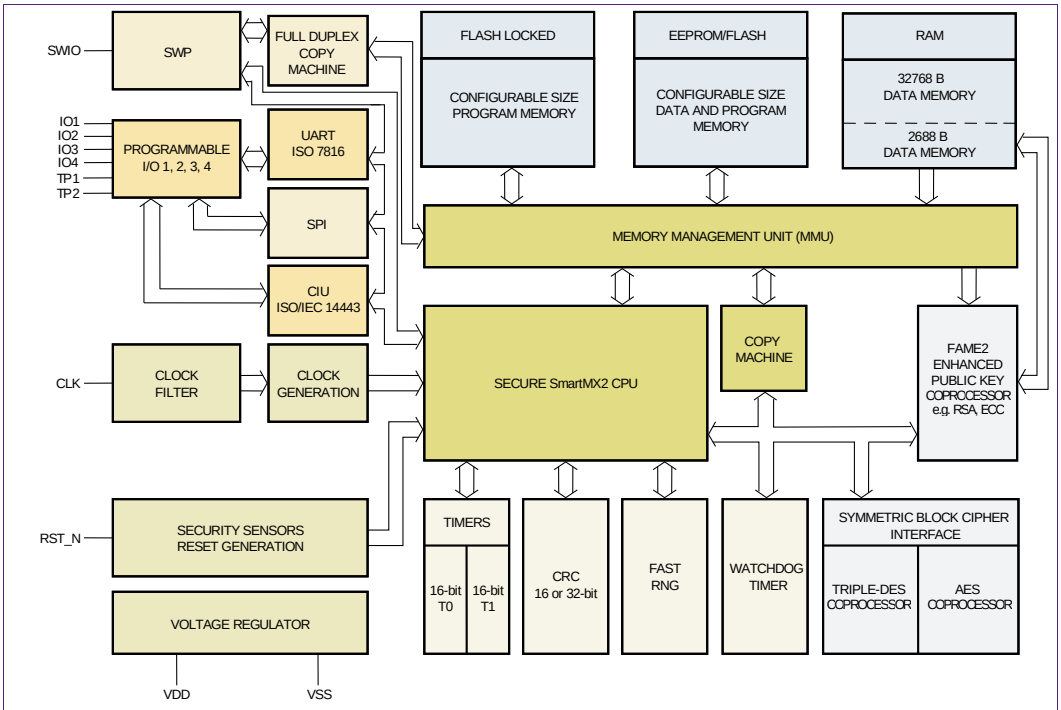

Fig 1. Block Diagram of P61N1M3PVD/VD-1/VE-1

P61N1M3PVD/VD-1/VE-1 consists of the IC hardware and IC Dedicated Software. The IC Dedicated Software is composed of IC Dedicated Test Software and IC Dedicated Support Software. The IC Dedicated Test Software contains the Test-ROM Software, the IC Dedicated Support Software is composed of the Boot-ROM Software, the Firmware Operating System and the Bootloader Software. All other software is called Security IC Embedded Software. The Security IC Embedded Software is not part of the TOE. All components of the TOE are listed in next section 1.4.1.1.

1.4.1.1 TOE components

Table 1. Components of the TOE in base type P61N1M3VD

Type	Name	TOE Identification	Form of delivery
IC hardware	P61N1M3VD	The IC hardware is base type P61N1M3VD and identified by its nameplate 9068B, which is located in the layout of the chip as described in [16].	wafer with dice acc. to 9068B_BE_20130604.gds2.gz
IC Dedicated Test Software	Test-ROM Software	The IC Dedicated Software is identified by its NXP Content Number (NCN) set Table 4, which can be read out by the Security IC Embedded Software as described in	ROM code on the IC acc. to 9068B_DA005_TESTROM_v1_btos_0Ev13_fos_9v30rc4.hex
IC Dedicated Support Software	Boot-ROM Software		
	Firmware Operating System		
	Bootloader Software		ROM code on the IC acc.

Type	Name	TOE Identification	Form of delivery
		[9].	to phBootloader_P61_Crc.hex
Document	SmartMX2 P61N1M3 Secure high-performance mobile controller, Product data sheet	The documents are provided by NXP.	PDF via DocStore
Document	P61N1M3 VD, NV Properties, data sheet addendum		PDF via DocStore
Document	Instruction Set for the SmartMX2 family, Secure smart card controller		PDF via DocStore
Document	Chip Health Mode (CHM) for P61N1M3, data sheet addendum		PDF via DocStore
Document	P61N1M3 Firmware interface specification, data sheet addendum		PDF via DocStore
Document	NXP Secure Smart Card Controller P61N1M3PVD/VD-1/VE-1 Information on Guidance and Operation		PDF via DocStore
Document	SmartMX2 family P61N1M3 VD/VE Wafer and delivery specification, data sheet addendum		PDF via DocStore
Document	Trust Provisioning – Trust Provisioning concept and security architecture[1]		PDF via DocStore
Document	Key Delivery Procedures for Trust Provisioning[1]		PDF via DocStore
Document	SmartMX2 family P61N1M3 Product errata sheet		PDF via DocStore

Table 2. Components of the TOE in base type P61N1M3VD-1

Type	Name	TOE Identification	Form of delivery
IC hardware	P61N1M3VD-1	The IC hardware is base type P61N1M3VD-1 and identified by its nameplate 9068B, which is located in the layout of the chip as described in [16].	wafer with dice acc. to 9068B_BE_20130604.gds2.gz
IC Dedicated Test Software	Test-ROM Software	The IC Dedicated Software is identified by its NXP Content Number (NCN) set in Table 5, which can be read out by the Security IC Embedded Software as described in [9].	ROM code on the IC acc. to 9068B_DA005_TESTROM_v1_btos_0Ev13_fos_9v30rc4.hex
IC Dedicated Support Software	Boot-ROM Software		
	Firmware Operating System		
	Bootloader Software		ROM code on the IC acc. to phBootloader_P61_Crc.hex

[1]This document is provided in case service Trust Provisioning is ordered, see [17].

Type	Name	TOE Identification	Form of delivery
Document	SmartMX2 P61N1M3 Secure high-performance mobile controller, Product data sheet	The documents are provided by NXP.	PDF via DocStore
Document	P61N1M3 VD, NV Properties, data sheet addendum		PDF via DocStore
Document	Instruction Set for the SmartMX2 family, Secure smart card controller		PDF via DocStore
Document	Chip Health Mode (CHM) for P61N1M3, data sheet addendum		PDF via DocStore
Document	P61N1M3 Firmware interface specification, data sheet addendum		PDF via DocStore
Document	NXP Secure Smart Card Controller P61N1M3PVD/VD-1/VE-1 Information on Guidance and Operation		PDF via DocStore
Document	SmartMX2 family P61N1M3 VD/VE Wafer and delivery specification, data sheet addendum		PDF via DocStore
Document	Trust Provisioning – Trust Provisioning concept and security architecture[2]		PDF via DocStore
Document	Key Delivery Procedures for Trust Provisioning[1]		PDF via DocStore
Document	SmartMX2 family P61N1M3 Product errata sheet		PDF via DocStore

Table 3. Components of the TOE in base type P61N1M3VE-1

Type	Name	TOE Identification	Form of delivery
IC hardware	P61N1M3VE-1	The IC hardware is base type P61N1M3VE-1 and identified by its nameplate 9068C, which is located in the layout of the chip as described in [16].	wafer with dice acc. to 9068C_20130808. gds2.gz
IC Dedicated Test Software	Test-ROM Software	The IC Dedicated Software is identified by its NXP Content Number (NCN) set in Table 5, which can be read out by the Security IC Embedded Software as described in [9].	ROM code on the IC acc. to 9068C_DA007_TESTROM_v1_btos_0Ev15_fos_9v3.hex
IC Dedicated Support Software	Boot-ROM Software		
	Firmware Operating System		
	Bootloader Software		ROM code on the IC acc. to phBootloader_P61_Crc.hex
Document	SmartMX2 P61N1M3 Secure	The documents are	PDF via DocStore

[2]This document is provided in case service Trust Provisioning is ordered, see [17].

Type	Name	TOE Identification	Form of delivery
	high-performance mobile controller, Product data sheet	provided by NXP.	
Document	P61N1M3 VE, NV Properties, data sheet addendum		PDF via DocStore
Document	Instruction Set for the SmartMX2 family, Secure smart card controller		PDF via DocStore
Document	Chip Health Mode (CHM) for P61N1M3, data sheet addendum		PDF via DocStore
Document	P61N1M3 Firmware interface specification, data sheet addendum		PDF via DocStore
Document	NXP Secure Smart Card Controller P61N1M3PVD/VD-1/VE-1 Information on Guidance and Operation		PDF via DocStore
Document	SmartMX2 family P61N1M3 VD/VE Wafer and delivery specification, data sheet addendum		PDF via DocStore
Document	Trust Provisioning – Trust Provisioning concept and security architecture[1]		PDF via DocStore
Document	Key Delivery Procedures for Trust Provisioning[1]		PDF via DocStore
Document	SmartMX2 family P61N1M3 Product errata sheet		PDF via DocStore

The base type P61N1M3VD is chosen by the customer via Order Entry Form [17] as detailed inTable 4. The P61N1M3VE has been superseded by the P61N1M3VE-1 and cannot be ordered.

Table 4. Evaluated base type P61N1M3VD

Name	Value	Description
Please specify NXP Content Number (NCN) Selection	• 65	for base type P61N1M3VD

The base types P61N1M3VD-1 and P61N1M3VE-1 are chosen by the customer via Order Entry Form [18] as detailed in Table 5.

Table 5. Evaluated base types P61N1M3VD-1 and P61N1M3VE-1

Name	Value	Description
Please specify NXP Content Number (NCN) Selection	• 67 • 84	for base type P61N1M3VD-1 for base type P61N1M3VE-1

1.4.2 Evaluated configurations

The customer selects logical and physical configuration options of each base type without modification of its physical scope described in section 1.4.1. Logical configuration options are structured in major configuration options according to section 1.4.2.1 and minor configuration options according to section 1.4.2.2. Physical configuration options are the package types as detailed in section 1.4.2.3.

1.4.2.1 Major configuration options

One major configuration is target of evaluation for each base type, which is denoted by name P61N1M3PVD for base type P61N1M3VD, P61N1M3PVD-1 for base type P61N1M3VD-1 and P61N1M3PVE-1 for base type P61N1M3VE-1. The major configuration is chosen by the customer via Order Entry Forms [17] and [18] as detailed in Table 6.

Table 6. Evaluated major configuration options

Name	Value	Description
Convergence Implementation Selection	• P	"P" is Plain version, no emulation (default)

The Order Entry Forms [17] and [18] are individual to each type name. The first seven characters in the name of a major configuration give the type name and therewith the Order Entry Form [17] resp. [18] belonging to.

Each major configuration is provided with several minor configuration options, which are defined in section 1.4.2.2.

1.4.2.2 Minor configuration options

Minor configurations are chosen by the customer via Order Entry Forms [17] and [18] as detailed in Table 7. The Order Entry Forms [17] and [18] identifies the minor configuration options, which are supported by a major configuration.

Table 7. Evaluated minor configuration options

Name	Value	Description
ROMinization: specifies the amount of Flash memory that shall be rominized (8k byte to 128k byte in multiples of 8k byte)	• Specific value	Determines the size of the ROMinized Flash area in multiples of 8 Kbytes.
ATR Test, acc. ISO/IEC 7816-3 allowed at NXP production test?	• NO • YES	ATR Test is done by NXP during production or not.
ROM[3] read instructions by Copy Machine allowed	• YES • NO	Read access by Copy Machine to ROM[3] is allowed or not.
NVM[4] read instructions by Copy Machine allowed	• YES • NO	Read access by Copy Machine to NVM[4] is allowed or not.
code execution from	• NO	Code execution from RAM is allowed or not.

[3] "ROM" refers to ROMinized Flash

[4] "NVM" refers to both, EEPROM and non-ROMinized Flash

RAM allowed	• YES	
Activation of “Card Disable” feature allowed	• NO • YES	If allowed, the TOE can be locked completely. Once activated by the Security IC Embedded Software, execution of the Security IC Embedded Software is inhibited after next reset.
EEPROM application content erase allowed	• NO • YES	Erase of application content of EEPROM enabled or not.
Inverse EEPROM Error Correction Attack Detection activated	• NO • YES	If activated, detection probability of fault injections to EEPROM can be increased.
Digital SWP operation via additional pads TP1 and TP2 (only if SWP interface is selected)	• NO • YES	If allowed, two additional pads are available for communication via SWP interface in dual pad configuration
CXRAM parity error configuration	• DISABLED • CONDITIONALLY ENABLED	Configuration of parity check on CXRAM read by Security IC Embedded Software
MIFARE crypto available in System Mode and User Mode	• NO • YES	MIFARE crypto hardware accessible in System Mode and User Mode or not.
FXRAM parity error configuration	• DISABLED • CONDITIONALLY ENABLED • ENABLED	Configuration of the parity check on FXRAM read by Security IC Embedded Software
EDATASCALE specification (EDATA size will be EDATASCALE * 16 bytes)	• 10h • Specific EDATASCALE = 0 up to E0h	Determines the size of the memory area available for the extended stack pointer in multiples of 16 bytes.
SWP or S2C interface	• SWP • S2C	Single Wire Protocol (SWP) interface or S^2C interface available.
Selection of reset value for UART CRC algorithm	• de-facto PC/SC • ISO/IEC 13239 / HDLC	Selection of CRC algorithm for ISO7816 enhanced protocol support.
Chip Health mode enabled	• YES • NO	If enabled, read-out of IC identification items and start of built-in self test functions is possible.
UID options	• single FNUID, four bytes (using the non-unique and revolving xFh range of ISO/IEC 14443) • double UID, seven bytes	Sets values of the contactless communication protocol parameters.
Contactless parameters with deviating protocol options	• ATQ0 value • ATQ1 value • SAK value • TA value	Sets values of the contactless communication protocol parameters.

See [9] for details on all minor configuration options listed in Table 7.

Please note that the optional MIFARE software is not in the scope of the evaluation

1.4.2.3 Evaluated package types

The commercial types are named according to the following format.

- P61N1M3*edd(d)*/9*Srnff(o)*

Cursive characters in the above format are replaced as described in Table 8 and Table 9 to retrieve a commercial type name. The commercial type name is composed of fixed symbols, which are detailed in Table 8, and variable entries, which are filled in according to the rules in Table 9.

Table 8. Fixed values definitions for commercial type name

Fixed symbol	Definition
P61N1M3	Type name acc. to Device Coding DC(1) in [9].
e	*e*=”P” for major configurations P61N1M3PVD¸ P61N1M3PVD-1 and P61N1M3PVE-1.
9	Identifier of Fab TSMC.
S	Silicon version code, which is also typed in the base type version that gives the ending in the name of each major configuration. *S*=”D” for base type P61N1M3VD with base type version “VD” and for base type P61N1M3VD-1 with base type version “VD-1”. *S*=”E” for base type P61N1M3VE-1 with base type version “VE-1”.
rn	NXP Content Number (NCN) acc. to [9], which identifies contents in ROM, IC-Dedicated-Software-Flash and Firmware-EEPROM for each base type. *rn*=“65” for base type P61N1M3VD, *rn*=“67” for base type P61N1M3VD-1, *rn*=“84” for base type P61N1M3VE-1.
(o)	*o*=”0” for major configurations P61N1M3PVD, P61N1M3PVD-1 and P61N1M3PVE-1.

Table 9. Variable definitions for commercial type name

Variable entry	Definition
dd(d)	Package type (alpha numeric, last character optional).
ff	FabKey Number (FKN) acc. to in [9], which identifies the contents in Application-Flash and Application-EEPROM at TOE Delivery.

Table 10 depicts the package types, which are supported in this Security Target for each major configuration of P61N1M3PVD/VD-1/VE-1. The two or three characters in each entry of the table stand for variable *dd(d)* in the commercial type name, and identify the package type. An empty cell means that the Security Target does not support the respective package type for the corresponding major configuration.

Table 10. Supported package types

P61N1M3PVD	Description
U*x*	Wafer not thinner than 50 µm and without wafer level chip scale package (WLCSP). The letter “*x*” in “U*x*” stands for a capital letter or a number, which identifies the wafer type, e.g. package type U15 stand for 150µm sawn wafer on film frame carrier.

Package types do not influence the security functionality of P61N1M3PVD/VD-1/VE-1. They solely define the pads of the die, which are connected to the package, as well as purpose and environment, in which the chip can be used. Note that the security of the TOE does not rely on which pad is connected or not. In case the TOE is delivered as wafer the customer can even connect all pads.

Security during development and production is ensured for all package types listed above, for details refer to section 1.4.4.

Information on how to order and identify P61N1M3PVD/VD-1/VE-1 and its major and minor configurations is described in [9].

1.4.3 Logical Scope of TOE

1.4.3.1 Hardware Description

The CPU of P61N1M3PVD/VD-1/VE-1 supports a 32-/24-/16-/8-bit instruction set and distinguishes five CPU modes, which are named in Table 11.

Table 11. CPU modes of the TOE

Super System Mode				
Boot Mode	Test Mode	Firmware Mode	System Mode	User Mode

Boot Mode, Test Mode and Firmware Mode are summarized under the term Super System Mode, which isn't a CPU mode on its own. When the CPU is in Super System Mode, it is always either in Boot Mode, Test Mode or Firmware Mode.

Boot Mode, Test Mode and Firmware Mode are not available to the Security IC Embedded Software. These three CPU modes are mapped one-to-one to three components of the IC Dedicated Software: The Boot-ROM Software is executed in Boot Mode, the Test-ROM Software is executed in Test Mode and the Firmware Operating System is executed in Firmware Mode. The Bootloader Software runs in System Mode, which is also available to the Security IC Embedded Software.

During phase 3 IC Manufacturing acc. to the Security IC product life cycle in the PP [6], start-up and reset of P61N1M3PVD/VD-1/VE-1 always complete with Test Mode and execution of the Test-ROM Software. Test Mode and Test-ROM Software are permanently disabled and Bootloader Software is irreversibly terminated before TOE Delivery acc. to the Security IC product life cycle [6]. Then start-up and reset of P61N1M3PVD/VD-1/VE-1 always end up in System Mode and execution of the Security IC Embedded Software.

System Mode and User Mode are available to the Security IC Embedded Software. P61N1M3PVD/VD-1/VE-1 serves the Security IC Embedded Software with hardware resources, which are controlled via Special Function Registers. Such hardware resources are CPU, Memory Management Unit, interrupt system, timers, watchdog timer, Non Volatile Counter, random number generator, AES, DES, CRC and Fame2 coprocessors, a Copy Machine and several electrical interfaces supported by a dedicated Full Duplex Copy Machine. System Mode has full access to all Special Function Registers that control these hardware resources, whereas their access is very limited in User Mode by default. Software running in System Mode can explicitly grant and deny access in User Mode to a subset of these Special Function Registers that interface to hardware components. Access in Firmware Mode to this subset of Special Function Registers is granted and denied in the same way so that System Mode is in control of these hardware components.

P61N1M3PVD/VD-1/VE-1 provides two types of interrupts, which are (a) exception interrupts, called "exceptions" in the following, and (b) event interrupts, called "interrupts" in the following. Exceptions and interrupts each force a jump to a specific fixed vector address. Any exception and interrupt can therewith be processed by a specific part of the Security IC Embedded Software.

In addition, P61N1M3PVD/VD-1/VE-1 provides eight firmware vectors (FVEC) and 32 system call vectors (SVEC). These vectors have to be explicitly called by the Security IC

Embedded Software. A jump to a firmware vector forces Firmware Mode and starts execution of the Firmware Operating System, a jump to a system call vector forces System Mode.

P61N1M3PVD/VD-1/VE-1 incorporates 184 KB ROM, 1216 KB Flash, 128 KB EEPROM and 34.625 KB RAM. Access to all memories is controlled by the Memory Management Unit, which partitions each memory as follows. The whole ROM and Flash partitions of 32 KB overall are reserved for the IC Dedicated Software. The IC Dedicated Software also allocates an EEPROM partition of 0.75k bytes in major configuration P61N1M3PVD/VD-1/VE-1. In addition, a RAM partition of 512 bytes in major configuration P61N1M3PVD/VD-1/VE-1 is reserved for the IC Dedicated Software. Remaining Flash, EEPROM and RAM is named Application-Flash, Application-EEPROM and Application-RAM, which is available to the Security IC Embedded Software. System Mode has full access to these memory partitions, whereas User Mode has no access by default. Software running in System Mode can define address areas in these memory partitions and explicitly grant and deny access in User Mode to each of these areas. The TOE Manufacturer EEPROM area of 768 bytes is located in Application-EEPROM and subject to separate access control. Only parts of it can be accessed by the Security IC Embedded Software.

Test Mode has full access to Application-Flash, Application-EEPROM and Application-RAM, but Boot Mode and Firmware Mode have no access to read, write or execute these memory partitions. However, the Security IC Embedded Software must call the Firmware Operating System to modify contents in Application-Flash, Application-EEPROM and TOE Manufacturer EEPROM area. Prior to that software running in System Mode must explicitly grant access to Firmware Mode for data integrity checking during erase/programming of Application-Flash and Application-EEPROM.

Address mapping to memory partitions and access to memory partitions and Special Function Registers is controlled by the hardware based on CPU modes so that IC Dedicated Software running in Super System Mode is separated from software running in System Mode or User Mode. In User Mode, access to Application-Flash, Application-EEPROM, Application-RAM and Special Function Registers that interface to hardware components is further controlled based on memory segmentation so that different applications running in User Mode can be separated with an appropriate memory management scheme defined by the Security IC Embedded Software.

The Application-RAM is further split in two parts. These are 31.5 Kbytes in major configuration P61N1M3PVD/VD-1/VE-1 for general purpose CXRAM and 2.625 Kbytes for FXRAM. Both parts are accessible to the CPU via the Memory Management Unit, but FXRAM is also accessible to the Fame2 coprocessor directly without access controlled by the Memory Management Unit. Thus, software which has access to the Fame2 coprocessor implicitly has access to FXRAM.

The Triple-DES coprocessor supports single DES and Triple-DES operations. Only Triple-DES is in the scope of this Security Target, in 2-key or 3-key operation with two/three 56-bit keys (112-/168-bit). The AES coprocessor supports AES operation with three different key lengths of 128, 192 or 256 bits. The Fame2 coprocessor supplies basic arithmetic functions to support implementation of asymmetric cryptographic algorithms by the Security IC Embedded Software. The random number generator provides true random numbers without pseudo random calculation. The CRC coprocessor supports CRC generation polynomials CRC-16 and CRC-32. The Copy Machine provides direct transfer of data between Special Function Registers and memories without interaction of the CPU. The Full Duplex Copy Machine is restricted to such direct data transfer between SWP

interface and XDATA. XDATA is a virtual memory address range, which is mapped to physical memory addresses of CXRAM in all CPU modes except for User Mode, in which mapping of virtual to physical addresses is defined by the Security IC Embedded Software based on memory segmentation.

Note: The Triple-DES and AES coprocessors only support encryption and decryption in ECB mode. The Security IC Embedded Software is responsible to implement appropriate padding mechanisms if neccesary.

The watchdog timer can be used by the Security IC Embedded Software to abort irregular program executions by a time-out mechanism.

P61N1M3PVD/VD-1/VE-1 operates with a single external power supply of 1.8 V, 3 V or 5 V nominal. The external clock frequency is used to synchronize ISO/IEC 7816 compliant communication and is limited to 10 MHz nominal. CPU and coprocessors always run without external clocks, their speed can be aligned to different clocks as selected by the Security IC Embedded Software.

P61N1M3PVD/VD-1/VE-1 also provides two power-save modes with reduced activity. These are named idle and power-down.

P61N1M3PVD/VD-1/VE-1 protects code and data against physical tampering while being stored to and/or processed in the device. Memory encryption is applied to all memories. ROM and RAM are equipped with integrity checks. Flash and EEPROM are protected by error corrections. Beyond that, dedicated security mechanisms verify code and data read accesses for fault injections. The IC is provided with active and passive shielding. Sensors for light are spread in the IC and sensors on temperature, voltages and frequencies are implemented. In addition, P61N1M3PVD/VD-1/VE-1 controls integrity of security relevant Special Function Registers and CPU registers.

P61N1M3PVD/VD-1/VE-1 provides security functionality, which acts independent of the Security IC Embedded Software, but some of its security functionality must be supported by security functionality of the Security IC Embedded Software. Interaction of such security functionalities is of particular importance for the overall system security.

1.4.3.2 Software Description

Operating system and applications for P61N1M3PVD/VD-1/VE-1 are developed by the customer and are part of the Security IC Embedded Software. The Security IC Embedded Software is stored to Flash and/or EEPROM and is not part of the TOE. The Security IC Embedded Software defines the operational usage of P61N1M3PVD/VD-1/VE-1.

The IC Dedicated Software is developed by NXP and stored to ROM, Flash and EEPROM outside of Application-Flash and Application-EEPROM. It is composed of the Test-ROM Software, the Boot-ROM Software, the Firmware Operating System and the Bootloader Software.

The Test-ROM Software is used by the TOE Manufacturer during production test. It includes the test operating system, test functions for the various blocks of the circuitry and shutdown functions to ensure that security relevant test functions cannot be executed illegally after IC manufacturing acc. to the Security IC product life cycle in the PP [6]. The Test-ROM Software is disabled before TOE Delivery acc. to the Security IC product life cycle in the PP [6].

The Boot-ROM Software is executed during start-up or reset of P61N1M3PVD/VD-1/VE-1, i.e. each time the device powers up or resets. It sets up P61N1M3PVD/VD-1/VE-1 and its configuration. In case minor configuration option “Chip Health mode” is set to “YES” the Boot-ROM Software serves an APDU interface, which supports the Composite Product

Manufacturer with functional testing and identification of P61N1M3PVD/VD-1/VE-1. This “Chip Health mode” is limited for security reasons to 63 executions.

The Firmware Operating System serves an FVEC interface, which is accessible to the Security IC Embedded Software. This interface provides erase and/or programming of Application-Flash and Application-EEPROM and also provides write access to the TOE manufacturer EEPROM area. This is mandatory to be used by the Security IC Embedded Software.

The Bootloader Software is implemented for purpose of tooling for development of Security IC Embedded Software and is not accessible to the Security IC Embedded Software. The Bootloader Software is terminated before TOE Delivery acc. to the Security IC product life cycle [6].

P61N1M3PVD/VD-1/VE-1 is always delivered with Firmware Operating System and Bootloader Software. The Firmware Operating System is in the scope of this Security Target wrt its hardware control interface. The Bootloader Software is only executed by the developer in secure environment and afterwards permanently blocked and not available for the customer. Hence, only the blocking of the Bootloader Software is in the scope of this evaluation.

1.4.3.3 Documentation

Document “SmartMX2 P61N1M3 Secure high-performance mobile controller, Product data sheet” [9] describes how to operate the hardware and its electrical interfaces. It also explains functionality and use of the hardware as well as access to its Special Function Registers by the Security IC Embedded Software. In particular, the CPU and its instruction set are introduced there, and the instruction set is further detailed in “Instruction Set for the SmartMX2 family, Secure smart card controller” [12]. Functionality and use of the Firmware Operating System as well as access to its FVEC interface by the Security IC Embedded Software are documented in “P61N1M3 Firmware interface specification, data sheet addendum” [14].

Proper use of minor configuration option “Chip Health mode” set to “YES” by the Composite Product Manufacturer is detailed in “Chip Health Mode (CHM) for P61N1M3, data sheet addendum” [13]. This includes technical functionality, electrical operation of the hardware and the APDU interface, which is served by the Boot-ROM Software.

The “NXP Secure Smart Card Controller P61N1M3PVD/VD-1/VE-1 Information on Guidance and Operation” [15] completes the data sheet and its addenda with requirements and recommendations on how to use and add to the security functionality of P61N1M3PVD/VD-1/VE-1 for its use as an evaluated product and in a way that supports the overall security of a composite product.

Customer orders of P61N1M3PVD/VD-1/VE-1 are specified in “Order Entry Form P61N1M3PVD/E” [17] and “Order Entry Form P61N1M3PVD-1/E-1, online document” [18]. Details on how to order P61N1M3PVD/VD-1/VE-1 in an evaluated configuration and package type are provided in the data sheet “SmartMX2 P61N1M3 Secure high-performance mobile controller, Product data sheet” [9].

Physical dimensions of wafer and die as well as delivery process and physical identification of P61N1M3PVD/VD-1/VE-1 are described in “SmartMX2 family P61N1M3 VD/VE Wafer and delivery specification, data sheet addendum” [16].

The product errata sheet “SmartMX2 family P61N1M3 Product errata sheet, NXP Semiconductors, Revision 1.2, Document Number 474012, 13 November 2018” [21] amends the data sheet with respect to the operation guidance of the MMU component.

The whole documentation shall be used by the developer to develop the Security IC Embedded Software.

1.4.4 Security during Development and Production

The Security IC product life cycle is scheduled in phases [6]. Phase 2 IC Development and phase 3 IC Manufacturing of this life cycle are part of this Security Target as well as phase 4 IC Packaging depending on TOE Delivery of P61N1M3PVD, which is either at the end of phase 3 or at the end of phase 4 as determined by the package type.

The development environment of P61N1M3PVD/VD-1/VE-1 always ranges from phase 2 IC Development to TOE Delivery. All other phases are part of the operational environment.

Appropriate to Application Note 3 in the PP [6] P61N1M3PVD/VD-1/VE-1 supports authentic delivery via FabKey, see [9], [16] and via minor configuration option “Chip Health mode”, see [9], [13].

In phase 2 IC Development of P61N1M3PVD/VD-1/VE-1 only those people have access to sensitive data of P61N1M3PVD/VD-1/VE-1, who are involved in the development project. In phase 3 IC Manufacturing NXP serves the Composite Product Manufacturer with storage of its contents to Application-Flash and Application-EEPROM of P61N1M3PVD/VD-1/VE-1. Different people are responsible for design data of NXP and data of the Composite Product Manufacturer.

In phase 3 IC Manufacturing of P61N1M3PVD/VD-1/VE-1 dice are produced and tested on wafers. Non-functional dice on a wafer are marked on a wafer map, which is provided to the Composite Product Manufacturer in electronic form.

In phase 4 IC Packaging of P61N1M3PVD/VD-1/VE-1 dice are embedded into modules, inlays or packages. This is not valid for P61N1M3PVD/VD-1/VE-1 since it is delivered as dice on wafers only.

The availability of major configurations of P61N1M3PVD/VD-1/VE-1 in package types is detailed in section 1.4.2.3.

Delivery processes between all involved sites provide accountability and traceability of the dice.

1.4.5 TOE Intended Usage

The operational environment of P61N1M3PVD/VD-1/VE-1 includes phase 1 IC Embedded Software Development and also ranges from TOE Delivery to phase 7 Operational Usage of the Security IC product life cycle [6]. These phases are not part of the TOE construction process in the sense of this Security Target. Information on these phases is included just to describe how P61N1M3PVD/VD-1/VE-1 is used after its construction. Nevertheless such security functionality of P61N1M3PVD/VD-1/VE-1, that is independent of the Security IC Embedded Software, is active at TOE Delivery and cannot be disabled by the Security IC Embedded Software.

The operational environment must be a controlled environment, except for phase 7 Operational Usage, which is an uncontrolled enviroment.

Phase 7 Operational Usage is the end-consumer environment of P61N1M3PVD/VD-1/VE-1. In this phase P61N1M3PVD/VD-1/VE-1 is in use by the end-consumer as part of a composite product. Its intended usage is defined by the Security IC Embedded Software.

P61N1M3PVD/VD-1/VE-1 is developed for most high-end safeguarded applications. It can be used to provide authorized conditional access in a wide range of applications.

Examples are identity cards, passports, banking cards, Pay-TV, portable communication SIM cards, health cards and transportation cards. The end-user environment covers a wide spectrum of very different functions, thus making it difficult to monitor and avoid abuse of P61N1M3PVD/VD-1/VE-1. In phase 7 Operational Usage P61N1M3PVD/VD-1/VE-1 is intended to be used in an insecure environment, which does not protect against threats.

P61N1M3PVD/VD-1/VE-1 may be assigned to a single end-consumer application only, but it has the capability to support multiple applications with different providers and several users. In this context, P61N1M3PVD/VD-1/VE-1 might store and process secrets of different providers and/or several users, which must be separated from each other. P61N1M3PVD/VD-1/VE-1 then must meet its security requirements for each single application. Secret data must be used as input for calculation of authentication data, of signatures and encryption of data and keys.

1.4.6 Interface of the TOE

The electrical interface of P61N1M3PVD/VD-1/VE-1 are the pads, which connect power supply, ground, reset input, clock input as well as communication pads I/O1, I/O2, I/O3 and I/O4, communication pad SWIO and also communication pads TP1 and TP2 in case minor configuration option “Access to additional general purpose I/O pads TP1 and TP2 allowed in System Mode” is set to “YES”.

Communication with the TOE can be established as follows.

- ISO/IEC 7816 compliant interface by use of ISO/IEC 7816 UART,
- I/O interface by use Special Function Registers,
- Serial Peripheral Interface (SPI),
- ETSI TS 102 613 compliant SWP interface,
- SWP interface in dual pad configuration by use of ETSI TS 102 613 protocol,
- S^2C interface by use of ISO/IEC 14443 protocol.

P61N1M3PVD/VD-1/VE-1 provides minor configuration options “Selection of interface configuration” and “Access to additional general purpose I/O pads TP1 and TP2 allowed in System Mode” to configure availability of communication interfaces resp. pads.

The logical interface of P61N1M3PVD/VD-1/VE-1 is composed of the following.

- The instruction set interface to the CPU acc. to [12], which is accessible via the Security IC Embedded Software.
- The Special Function Registers interface acc. to [9], which is accessible via the Security IC Embedded Software.
- The FVEC interface to the Firmware Operating System acc. to [14], which is accessible via the Security IC Embedded Software.
- The APDU interface to the Boot-ROM Software acc. to [13], which is accessible via the ISO/IEC 7816 compliant interface. This interface is available in case minor configuration option “Chip Health mode enabled” is set to “YES”.

The chip surface must be considered as an interface of P61N1M3PVD/VD-1/VE-1. This interface could be exposed to environmental stress or physically manipulated by an attacker.

Note: P61N1M3PVD/VD-1/VE-1 does not operate without power supply. The reset input must be high in case a valid clock signal is applied to the clock input. A valid clock signal must be applied to the clock input in case the CPU is configured to run at

this clock. In addition, a communication interface must be available, which is defined and controlled by the Security IC Embedded Software based on the electrical behaviour provided by P61N1M3PVD/VD-1/VE-1.

2. Conformance Claims

This chapter is divided into the following sections: “CC Conformance Claim”, “Package claim”, “This Security Target claims conformance to assurance package EAL6 augmented. The augmentation to EAL6 is ALC_FLR.1. In addition, this Security Target is augmented by ASE_TSS.2, which is chosen to include architectural information on the security functionality of the TOE.

Note: The Protection Profile (PP) [6] to which this Security Target claims conformance, see section 2.3, requires assurance level EAL4 augmented. All changes needed for EAL6 augmented are described in the relevant sections of this Security Target.

Furthermore, this Security Target claims conformance to the following packages from Protection Profile (PP) [6]:

- Package “Package 1: Loader dedicated for usage in secured environment only” conformant (see section 7.3.1 of the PP),
- Package “TDES” conformant (see section 7.4.1 of the PP),
- Package “AES” conformant (see section 7.4.2 of the PP).

The level of evaluation and the functionality of P61N1M3PVD/VD-1/VE-1 are chosen in order to allow the confirmation that P61N1M3PVD/VD-1/VE-1 is suitable for use within devices compliant with the German Digital Signature Law.

Security IC Protection Profile claim” and “Conformance Claim Rationale”.

2.1 CC Conformance Claim

This Security Target claims to be conformant to version 3.1 of Common Criteria for Information Technology Security Evaluation according to

- “Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model, Version 3.1, Revision 5, April 2017, CCMB-2017-04-001” [1],
- “Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002” [2],
- “Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-003” [3].

The following methodology is used for evaluation.

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

This Security Target claims to be CC Part 2 extended and CC Part 3 conformant. The extended Security Functional Requirements are defined in chapter 5.

2.2 Package claim

This Security Target claims conformance to assurance package EAL6 augmented. The augmentation to EAL6 is ALC_FLR.1. In addition, this Security Target is augmented by ASE_TSS.2, which is chosen to include architectural information on the security functionality of the TOE.

Note: The Protection Profile (PP) [6] to which this Security Target claims conformance, see section 2.3, requires assurance level EAL4 augmented. All changes needed for EAL6 augmented are described in the relevant sections of this Security Target.

Furthermore, this Security Target claims conformance to the following packages from Protection Profile (PP) [6]:

- Package “Package 1: Loader dedicated for usage in secured environment only” conformant (see section 7.3.1 of the PP),
- Package “TDES” conformant (see section 7.4.1 of the PP),
- Package “AES” conformant (see section 7.4.2 of the PP).

The level of evaluation and the functionality of P61N1M3PVD/VD-1/VE-1 are chosen in order to allow the confirmation that P61N1M3PVD/VD-1/VE-1 is suitable for use within devices compliant with the German Digital Signature Law.

2.3 Security IC Protection Profile claim

This Security Target claims strict conformance to the Protection Profile (PP) “Security IC Platform Protection Profile PP-0084 “Security IC Platform Protection Profile with Augmentation Packages”, Version 1.0, 2014-01-13, BSI-CC-PP-0084-2014, available at https://www.bsi.bund.de” [6].

Since the Security Target claims conformance to this PP [6] its concepts are used in the same sense. For definition of terms refer to the PP [6]. These terms also apply to this Security Target.

P61N1M3PVD/VD-1/VE-1 provides additional functionality, which is not covered in the PP [6]. In accordance with Application Note 5 of the PP [6] this additional functionality is added using policy “P.Add-Components”, for details see section 3.3 of this Security Target.

2.4 Conformance Claim Rationale

As stated in section 2.3 this Security Target claims strict conformance to the Protection Profile (PP) [6].

P61N1M3PVD/VD-1/VE-1 is defined in section 1.3.2 of this Security Target is a smartcard controller. This is consistent with the TOE definition of a Security IC in section 1.2.2 of the PP [6].

All sections of this Security Target, in which security problem, security objectives and security requirements are defined, clearly state which of these items are taken from the PP [6] and which ones are added in this Security Target. This is not repeated here. Moreover, all additionally stated items in this Security Target do not contradict the items included from the PP [6], (see the respective sections in this document). The operations done for the Security Functional Requirements taken from the PP [6] are also clearly indicated.

The evaluation assurance level claimed for this target (EAL6+) is shown in section 6.2 to include respectively exceed the requirements claimed by the PP [6] (EAL4+).

These considerations show that the Security Target correctly claims strict conformance to the PP [6].

3. Security Problem Definition

This Security Target claims conformance to the Protection Profile (PP) [6]. Assets, threats, assumptions and organizational security policies are taken from the PP [6]. This chapter lists these assets, threats, assumptions and organisational security policies, and describes extensions to these elements in detail.

The chapter is divided into the following sections: “Description of Assets”, “Threats”, “Organisational Security Policies” and “Assumptions”.

3.1 Description of Assets

All assets, which are defined in section 3.1 of the PP [6], are related to standard functionality. They are applied in this Security Target. These assets are:

- integrity and confidentiality of User Data stored and in operation,
- integrity and confidentiality of Security IC Embedded Software, stored and in operation,
- correct operation of the Security Services provided by the TOE for the Security IC Embedded Software,
- quality of random numbers.

To be able to protect these assets the TOE shall protect its security functionality. Therefore, critical information on the TOE shall be protected. Critical information includes:

- logical design data, physical design data, IC Dedicated Software, configuration data,
- initialization data and pre-personalization data, specific development aids, data related to test and characterization, material for software development support, photo masks.

Note that the keys for cryptographic calculations using security services of the TOE are treated as User Data.

3.2 Threats

All threats, which are defined in section 3.2 of the PP [6], are valid for this Security Target. These threats are listed in Table 12.

Table 12. Threats defined in the PP [6]

Name	Title
T.Leak-Inherent	Inherent Information Leakage
T.Phys-Probing	Physical Probing
T.Malfunction	Malfunction due to Environmental Stress
T.Phys-Manipulation	Physical Manipulation
T.Leak-Forced	Forced Information Leakage
T.Abuse-Func	Abuse of Functionality
T.RND	Deficiency of Random Numbers

In compliance with Application Note 4 in the PP [6] the TOE provides additional functionality to protect against threats that appear when the TOE is used for multiple applications.

The TOE provides the Security IC Embedded Software running in System Mode with control of access to memories and hardware components by different applications running in User Mode. In this context, User Data of different applications is stored to such memory

and processed by such hardware components. The Security IC Embedded Software controls all these User Data. Any access to User Data assigned to one application by another application contradicts separation between different applications and is considered as a threat.

The TOE shall avert threat T.Unauthorised-Access as specified below.

T.Unauthorised-Access **Unauthorised Memory or Hardware Access**

Adverse action: An attacker may try to read, modify or execute code or data stored to restricted memory areas. An attacker may try to access or operate restricted hardware resources by executing code that accidentally or deliberately accesses these restricted hardware resources.

Any code executed or data used in Boot Mode, Firmware Mode, System Mode or User Mode may accidentally or deliberately access code or User Data of other applications. Any code executed or data used in Boot Mode, Firmware Mode, System Mode or User Mode may accidentally or deliberately access hardware resources that are restricted to other applications.

Threat agent: Attacker having high attack potential and access to the TOE.

Asset: Code executed by and data belonging to the IC Dedicated Support Software running in Super System Mode as well as code executed by and data belonging to the Security IC Embedded Software.

Restrictions of access to memories and hardware resources, which are available to the Security IC Embedded Software, must be defined and implemented by the security policy of the Security IC Embedded Software based on the specific application context.

The threats defined in this Securty Target are summarized in Table 13.

Table 13. Additional threats defined in this ST

Name	Title
T.Unauthorised-Access	Unauthorised Memory or Hardware Access

3.3 Organisational Security Policies

All security policies, which are defined in section 3.3 of the PP [6], are valid for this Security Target. These security policies are listed in Table 14.

Table 14. Security policies defined in the PP [6]

Name	Title
P.Process-TOE	Protection during TOE Development and Production
P.Lim_Block_Loader	Limiting and Blocking the Loader Functionality
P.Crypto-Service	Cryptographic services of the TOE

In compliance with Application Note 5 in the PP [6], this Security Target defines two additional security policies as detailed below.

The TOE provides specific security functionality, which can be used by the Security IC Embedded Software. This specific security functionality is not derived from threats identified for the TOE. Instead, the Security IC Embedded Software decides how to use

this security functionality to protect from threats for the composite product. Thus, security policy P.Add-Components is defined as follows.

P.Add-Components **Additional Specific Security Components**

The TOE shall provide the following additional security functionality to the Security IC Embedded Software:

- Integrity support of data stored to Flash/EEPROM.

The security policies defined in this Securty Target are summarized in Table 15.

Table 15. Additional security policies defined in this ST

Name	Title
P.Add-Components	Additional Specific Security Components

3.4 Assumptions

All assumptions, which are defined in section 3.4 of the PP [6], are valid for this Security Target. These assumptions are listed inTable 16.

Table 16. Assumptions defined in the PP [6]

Name	Title
A.Process-Sec-IC	Protection during Packaging, Finishing and Personalisation
A.Resp-Appl	Treatment of User Data of the Composite TOE

In compliance with Application Notes 6 and 7 in PP [6], this Security Target defines three additional assumptions as follows.

A.Check-Init **Check of initialization data by the Security IC Embedded Software**

The Security IC Embedded Software must provide a function to check initialization data. Such data is defined by the Composite Product Manufacturer and injected by the TOE Manufacturer into the non-volatile memory to provide the ability to identify and trace the TOE.

The following additional assumption considers specialized encryption hardware of the TOE.

The developer of the Security IC Embedded Software must ensure appropriate usage of key-dependent functions as defined below during phase 1 IC Embedded Software Development of the Security IC product life cycle [6].

A.Key-Function **Usage of Key-dependent Functions**

Key-dependent functions (if any) shall be implemented to the Security IC Embedded Software in a way that they are not susceptible to leakage attacks (as described under T.Leak-Inherent and T.Leak-Forced).

Note that here, the routines which may compromise keys when being executed, are part of the Security IC Embedded Software. In contrast to this, threats T.Leak-Inherent and T.Leak-Forced address (i) the cryptographic routines, which are part of the TOE and (ii) the processing of User Data including cryptographic keys.

The assumptions policies defined in this Securty Target are summarized in Table 17.

Table 17. Additional assumptions defined in this ST

Name	Title
A.Check-Init	Check of initialization data by the Security IC Embedded SW
A.Key-Function	Usage of Key-dependent Functions

4. Security Objectives

This chapter contains the following sections: “Security Objectives for the TOE”, “Security Objectives for the Security IC Embedded Software development Environment”, “Security Objectives for the Operational Environment” and “Security Objectives Rationale”.

4.1 Security Objectives for the TOE

All security objectives for the TOE, which are defined in the PP [6], are applied to this Security Target. These security objectives are listed in Table 18.

Table 18. Security objectives for the TOE defined in the PP [6]

Name	Title
O.Leak-Inherent	Protection against Inherent Information Leakage
O.Phys-Probing	Protection against Physical Probing
O.Malfunction	Protection against Malfunctions
O.Phys-Manipulation	Protection against Physical Manipulation
O.Leak-Forced	Protection against Forced Information Leakage
O.Abuse-Func	Protection against Abuse of Functionality
O.Identification	TOE Identification
O.RND	Random Numbers
O.Cap_Avail_Loader	Capability and availability of the Loader
O.TDES	Cryptographic service Triple-DES
O.AES	Cryptographic service AES

In compliance with Application Notes 9 and 10 in the PP [6], additional security objectives for the TOE are defined below based on additional functionality provided by the TOE.

O.NVM_INTEGRITY **Integrity support of data stored to EEPROM and Flash**

The TOE shall provide re-trimming of EEPROM memory to support integrity of contents stored to. The TOE shall provide detection of wear-out failures in EEPROM and Flash memories to support integrity of contents stored to.

O.FM_FW **Firmware Mode Firewall**

The TOE shall provide separation between the Firmware Operating System and the Security IC Embedded Software. This separation shall comprise code execution and data access.

O.MEM_ACCESS **Area based Memory Access Control**

The TOE shall control access of CPU instructions and copy machines to memory areas depending on memory partitioning and based on CPU modes Boot Mode, Test Mode, Firmware Mode, System Mode and User Mode. In User Mode, the TOE shall control access also based on memory segments, which are configured in System Mode when implementing a memory management scheme. This control shall be individual to each memory segment and consider different access rights.

O.SFR_ACCESS	**Special Function Register Access Control** The TOE shall control access of CPU instructions and copy machines to Special Function Registers depending on the purpose of the register and based on CPU modes. The TOE shall provide System Mode with the ability to configure access rights for User Mode and Firmware Mode to Special Function Registers that interface to hardware components. In User Mode, the TOE shall control access to these Special Function Registers also based on memory segments, which are configured in System Mode when implementing a memory management scheme. This control shall be individual to each memory segment.

4.2 Security Objectives for the Security IC Embedded Software development Environment

The security objective for the Security IC Embedded Software development Environment, which is defined in the PP [6] section 4.2, is applied to this Security Target. This security objective is listed in Table 19.

Table 19. Security objectives for the Security IC Embedded Software development environment defined in the PP [6]

Security objective	Description	Applied to phase
OE.Resp-Appl	Treatment of User Data	Phase 1

Clarification of OE.Resp-Appl Treatment of User Data

By definition cipher or plain text data and cryptographic keys are user data of the Composite TOE. The Security IC Embedded Software shall treat these data appropriately, use proper secret keys only (chosen from a large key space) as input for the cryptographic services of the TOE and use keys and functions appropriately in order to ensure the strength of cryptographic operation. This means that keys are treated as confidential as soon as they are generated. Keys must be unique with a very high probability, as well as cryptographically strong. For example, if asymmetric algorithms are used, it must be ensured that it is not possible to derive the private key from a related public key using the attacks defined in this Security Target. If keys are imported into the TOE and/or derived from other keys, quality and confidentiality must be maintained. This implies that appropriate key management has to be realised in the environment.

In case the Security IC Embedded Software operates multiple applications on the TOE, OE.Resp-Appl must also be met. The Security IC Embedded Software must not disclose security relevant user data of the Composite TOE of one application to another application when processed in or stored to the TOE.

4.3 Security Objectives for the Operational Environment

In addition to the security objectives for the operational environment as required by CC Part 1 [1] all security objectives for the operational environment, which are defined in the PP [6] in the sections 4.3 and 7.3.1 due to the chosen package “Loader 1”, are applied to this Security Target. These security objectives are listed in Table 20.

Table 20. Security objectives for the operational environment, taken from the PP [6]

Security objective	Description	Applied to phase
OE.Process-Sec-IC	Protection during composite product	From TOE Delivery up to

Security objective	Description	Applied to phase
	manufacturing	the end of phase 6
OE.Lim_Block_Loader	Limitation of capability and blocking the Loader	This objective is a special requirement of the German scheme. The loader is only used up to Phase 3 by the developer and afterwards permanently blocked before delivery to the customer.

The following additional security objectives for the operational environment are defined in this Security Target.

The following security objective for the operational environment derives from security policy A.Check-Init. The TOE provides specific functionality that requires the TOE Manufacturer to implement measures for unique identification of the TOE. Security objective OE.Check-Init is defined to allow for such a TOE specific implementation.

OE.Check-Init **Check of initialisation data by the Security IC Embedded Software**

To ensure the receipt of the correct TOE, the Security IC Embedded Software shall check a sufficient part of the pre-personalisation data. This shall include at least the FabKey Data that is agreed between the customer and the TOE Manufacturer.

4.4 Security Objectives Rationale

Section 4.4 in the PP [6] provides a rationale how the threats, organisational security policies and assumptions are addressed by the security objectives defined in the PP [6]. Table 21 summarizes this.

Table 21. Security objectives versus treats, policies, assumptions as defined in the PP [6]

Threat, policy, assumption	Security objective	Applied to phase
T.Leak-Inherent	O.Leak-Inherent	
T.Phys-Probing	O.Phys-Probing	
T.Malfunction	O.Malfunction	
T.Phys-Manipulation	O.Phys-Manipulation	
T.Leak-Forced	O.Leak-Forced	
T.Abuse-Func	O.Abuse-Func	
T.RND	O.RND	
P.Crypto-Service	O.TDES O.AES	
P.Process-TOE	O.Identification	Phases 2 – 3
P.Lim_Block_Loader	O.Cap_Avail_Loader OE.Lim_Block_Loader	Phase 3
A.Resp-Appl	OE.Resp-Appl	Phase 1
A.Process-Sec-IC	OE.Process-Sec-IC	Phases 4 – 6

Table 22 summarizes how threats, organisational security policies and assumptions are addressed by the security objectives wrt. those items defined in the Security Target. All these items are in line with those in the PP [6].

Table 22. Security objectives versus threats, policies, assumptions defined in this ST

Threat, policy, assumption	Security objective	Applied to phase
T.Unauthorised-Access	O.FM_FW O.MEM_ACCESS O.SFR_ACCESS	
P.Add-Components	O.NVM_INTEGRITY	
A.Check-Init	OE.Check-Init	Phases 1 and 4 - 6
A.Key-Function	OE.Resp-Appl	Phase 1

The rationale for all items defined in the Security Target is given below.

The justification related to threat T.Unauthorised-Access is as follows:

According to security objectives O.FM_FW, O.MEM_ACCESS and O.SFR_ACCESS the TOE must enforce memory partitioning with address mapping and control of access to memories and Special Function Registers in Firmware Mode, System Mode and User Mode and must enforce a memory management scheme in User Mode so that access to memories and Special Function Registers is under control. Access rights in Firmware Mode and User Mode must be explicitly granted by Security IC Embedded Software running in System Mode. Thus, security violations caused by accidental or deliberate access to restricted data, code and shared hardware resources can be prevented. Threat T.Unauthorised-Access is therewith covered by these security objectives.

The justification related to security policy P.Add-Components is as follows:

Security objective O.NVM_INTEGRITY requires the TOE to implement exactly the specific security functionality, which is defined in P.Add-Components, so that the security policy is covered by the security objective.

However, security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced must also be achieved for this specific security functionality defined in P.Add-Components, since the related threats also apply to this security functionality. For multiple applications, Security IC Embedded Software running in System Mode controls different applications running in User Mode. In this context, it is wanted that most of the specific security functionality cannot be influenced or used in User Mode.

The justification related to assumption A.Check-Init is as follows: Security objective OE.Check-Init requires the Security IC Embedded Software to implement a function assumed in assumption A.Check-Init, so that the assumption is covered by the security objective.

The justification related to the assumption A.Key-Function is as follows:

The definition of security objective OE.Resp-Appl in the PP [6] is further clarified in this Security Target: By definition cipher or plain text data and cryptographic keys are user data of the Composite TOE. So, the Security IC Embedded Software will protect such data if required and use keys and functions appropriately in order to ensure the strength of cryptographic operation. Quality and confidentiality must be maintained for keys that are imported and/or derived from other keys. This implies that appropriate key management has to be implemented in the environment. In addition, the treatment of user data of the Composite TOE comprises the implementation of a multi-application

operating system that does not disclose security relevant user data of one application to another one. These measures make sure that the assumption A.Key-Function is still covered by the security objective OE.Resp-Appl although additional functions are being supported according to P.Add-Components.

The justification of policies and assumptions, which are defined in this Securtiy Target, show that they do not contradict to the rationales given in the PP [6] for threats, policies and assumptions defined there.

5. Extended Components Definition

This Security Target does not define extended components.

All extended security functional requirements, which are defined in chapter 5 and 7.5 of the Protection Profile (PP) [6], are included in this Security Target. In this context, the security functional requirements for random number generation used for cryptographic purposes are refined according to [8].

6. Security Requirements

This chapter defines the security requirements that shall be met by the TOE. These security requirements are composed of the security functional requirements and the security assurance requirements that the TOE must meet in order to achieve its security objectives. This chapter is divided into sections “Security Functional Requirements”, “Security Assurance Requirements” and “Security Requirements Rationale”.

CC allows several operations to be performed on security requirements (on the component level); refinement, selection, assignment, and iteration are defined in section 8.1 of CC Part 1 [1]. These operations are used in the PP [6] and in this Security Target, respectively.

The **refinement** operation is used to add details to requirements, and thus, further intensifies a requirement. Refinements of security requirements are denoted in such a way that added words are in bold text and changed words are crossed out.

The **selection** operation is used to select one or more options provided by the PP [6] or CC in stating a requirement. Selections having been made are denoted as italic text.

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments having been made are denoted as italic text.

The **iteration** operation is used when a component is repeated with varying operations. It is denoted by showing brackets “[*iteration indicator*]” and the *iteration indicator* within the brackets.

For the sake of a better readability, the iteration operation may also be applied to some single components (being not repeated) in order to indicate belonging of such SFRs to same functional cluster. In such a case, the iteration operation is applied to only one single component.

Whenever an element in the PP [6] contains an operation that is left uncompleted, the Security Target has to complete that operation.

6.1 Security Functional Requirements

All Security Functional Requirements (SFRs) of the TOE are presented in the following sections to support a better understanding of the combination of the PP [6] and this Security Target.

6.1.1 SFRs of the Protection Profile

All SFRs, which are defined in the PP [6], are summarized in Table 23. Some of these SFRs are defined in CC Part 2 [2] and eventually subject to refinement, selection, assignment and/or iteration operation in the PP [6]. Others are newly defined in the PP [6]. These are indicated in the third column of the table.

Table 23. SFRs defined in the PP [6]

SFR	Title	Defined in
FRU_FLT.2	Limited fault tolerance	CC, Part 2
FPT_FLS.1	Failure with preservation of secure state	CC, Part 2
FMT_LIM.1	Limited capabilities	PP, section 5.2
FMT_LIM.2	Limited availability	PP, section 5.2
FAU_SAS.1	Audit storage	PP, section 5.3

SFR	Title	Defined in
FPT_PHP.3	Resistance to physical attack	CC, Part 2
FDP_ITT.1	Basic internal transfer protection	CC, Part 2
FPT_ITT.1	Basic internal TSF data transfer protection	CC, Part 2
FDP_IFC.1	Subset information flow control	CC, Part 2
FCS_RNG.1	Random number generation	PP, section 5.1
FDP_SDI.2	Stored data integrity and monitoring	CC, Part 2
FDP_SDC.1	Stored data confidentiality	PP, section 5.4
FMT_LIM.1/Loader	Limited Capabilities – Loader	PP, section 7.3.1
FMT_LIM.2/Loader	Limited Availability – Loader	PP, section 7.3.1
FCS_COP.1/TDES	Cryptographic operation – TDES	PP, section 7.4.1
FCS_CKM.4/TDES	Cryptographic key destruction –TDES	PP, section 7.4.1
FCS_COP.1/AES	Cryptographic operation – AES	PP, section 7.4.2
FCS_CKM.4/AES	Cryptographic key destruction - AES	PP, section 7.4.2

The operations made in PP [6] fully apply for the TOE and are not highlighted explicitly. Only the operations made in this Security Target are further denoted in blue.

The SFRs FRU_FLT.2, FPT_FLS.1, FMT_LIM.1, FMT_LIM.2, FDP_IFC.1 and FPT_PHP.3 are completely defined in PP [6] and are not repeated here.

FAU_SAS.1

The SFR FAU_SAS.1 is defined in the PP [6], §163 and specified below. Iteration [HW] is done here to prepare for other iterations that address any future major configurations of the TOE.

FAU_SAS.1[HW]	Audit storage
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FAU_SAS.1.1[HW]	The TSF shall provide the test process before TOE Delivery with the capability to store *Initialisation Data and/or Pre-personalisation Data*[5] in the *Flash or/and EEPROM*[6].

FDP_ITT.1

The SFR FDP_ITT.1 is completely defined in the PP [6], §173, only the iteration [HW] is done here to prepare for other iterations that address any future major configurations of the TOE.

FDP_ITT.1[HW]	Basic internal transfer protection
Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_ITT.1.1[HW]	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.

[5] [selection: *Initialisation Data, Pre-personalisation Data, [assignment: other data]*]

[6] [assignment: *type of persistent memory*]

Refinement:	**The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor) are seen as physically-separated parts of the TOE.**

FPT_ITT.1

The SFR FPT_ITT.1 is completely defined in the PP [6], §174, only the iteration [HW] is done here to prepare for other iterations that address any future major configurations of the TOE.

FPT_ITT.1[HW]	Basic internal TSF data transfer protection
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_ITT.1.1[HW]	The TSF shall protect TSF data from disclosure when it is transmitted between separate parts of the TOE.
Refinement:	**The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor) are seen as separated parts of the TOE.**

FCS_RNG.1

SFR FCS_RNG.1 is defined in the PP [6], §178 and specified below. As the TOE is certified in German scheme, the operations performed in PP [6], §400 apply. Iteration [HW] is done here to prepare for other iterations that address any future major configurations of the TOE.

FCS_RNG.1/PTG.2[HW]	Random number generation - PTG.2
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FCS_RNG.1.1/PTG.2[HW]	The TSF shall provide a physical random number generator that implements:
(PTG.2.1)	A total failure test detects a total failure of entropy source immediately when the RNG has started. When a total failure is detected, no random numbers will be output.
(PTG.2.2)	If a total failure of the entropy source occurs while the RNG is being operated, the RNG *prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source*[7].
(PTG.2.3)	The online test shall detect non-tolerable statistical defects of the raw random number sequence (i) immediately when the RNG has started, and (ii) while the RNG is being operated. The TSF must not output any random numbers before the

[7] [selection: *prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source, generates the internal random numbers with a post-processing algorithm of class DRG.2 as long as its internal state entropy guarantees the claimed output entropy*]

power-up online test has finished successfully or when a defect has been detected.

(PTG.2.4) The online test procedure shall be effective to detect non-tolerable weaknesses of the random numbers soon.

(PTG.2.5) The online test procedure checks the quality of the raw random number sequence. It is triggered *at regular intervals or continuously*[8]. The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time.

Note: The TOE provides the two options of which the Security IC Embedded Software can choose one.

FCS_RNG.1.2/PTG.2[HW] The TSF shall provide *octets of bits*[9] that meet

(PTG.2.6) Test procedure A[10] does not distinguish the internal random numbers from output sequences of an ideal RNG.

(PTG.2.7) The average Shannon entropy per internal random bit exceeds 0.997.

Note: The Shannon entropy 0.997 per internal random bit compares to 7.976 per octet.

Note: Application Note 21 in the PP [6] requires that the Security Target specifies for the security capabilities in FCS_RNG.1.1 how the results of the total failure test of the random source are provided to the Security IC Embedded Software. The TOE features a hardware test which is called by the Security IC Embedded Software. The results of the internal test sequence are provided to the Security IC Embedded Software as a pass or fail criterion by means of a special function register.

The entropy of the random number is measured by the Shannon-Entropy as follows:

$E = -\sum_{i=0}^{255} p_i \cdot \log_2 p_i$, where p_i is the probability that the byte $(b_7, b_6, \mathrm{K}, b_0)$ is equal to i as binary number. Here term "bit" means measure of the Shannon-Entropy.

The value "7.976" is assigned due to the requirements of AIS31 [7].

FDP_SDI.2

The SFR FDP_SDI.2 is defined in PP [6], §169 and specified below. Iteration [HW] is done here to prepare for other iterations that address any future major configurations of the TOE and a refinement is added.

FDP_SDI.2[HW] Stored data integrity monitoring and action

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring

Dependencies: No dependencies.

[8] [selection*: externally, at regular intervals, continuously, applied upon specified internal events*]

[9] [selection: *bits, octets of bits, numbers* [assignment: *format of the numbers*]]

[10] [assignment: *additional standard test suites*], according to §295 in [8] this assignment may be empty

FDP_SDI.2.1[HW]	The TSF shall monitor user data stored in containers controlled by the TSF for *integrity violations due to ageing*[11] on all objects, based on the following attributes: *user data including code stored in Flash and EEPROM*[12].
FDP_SDI.2.2[HW]	Upon detection of a data integrity error, the TSF shall *in case of Flash and EEPROM inform the Security IC Embedded Software, and in case of EEPROM also adjust the EEPROM write operation*[13].
Refinement:	**Each Flash resp. EEPROM memory page is considered as one container and action is done for one complete Flash resp. EEPROM memory page.**

FDP_SDC.1

The SFR FDP_SDC.1 is defined in PP [6], §168 and specified below. Iteration [HW] is done here to prepare for other iterations that address any future major configurations of the TOE.

FDP_SDC.1[HW]	Stored data confidentiality
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FDP_SDC.1.1[HW]	The TSF shall ensure the confidentiality of the information of the user data while it is stored in the *RAM and Non-Volatile Memory*[14].

FMT_LIM.1/Loader

The SFR FMT_LIM.1/Loader is defined in PP [6], §355 and specified below.

FMT_LIM.1/Loader	Limited capabilities
Hierarchical to:	No other components.
Dependencies:	FMT_LIM.2 Limited availability.
FMT_LIM.1.1/Loader	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 Loader functionality after *permanently locking the Loader*[15] does not allow stored user data to be disclosed or manipulated by unauthorized user.

FMT_LIM.2/Loader

The SFR FMT_LIM.2/Loader is defined in PP [6], §356 and specified below.

FMT_LIM.2/Loader	Limited availability - Loader

[11] [assignment: *integrity errors*]

[12] [assignment: *user data attributes*]

[13] [assignment: *action to be taken*]

[14] [assignment: *memory area*]

[15] [assignment: *action*]

Hierarchical to:	No other components.
Dependencies:	FMT_LIM.1 Limited capabilities.
FMT_LIM.2.1/Loader	The TSF shall be designed in a manner that limits its availability so that in conjunction with “Limited capabilites (FMT_LIM.1)” the following policy is enforced: The TSF prevents deploying the Loader functionality after *permanently locking the Loader*[16].

FCS_COP.1/TDES

The FCS_COP.1/TDES is defined in PP [6], §379 and specified below.

FCS_COP.1/TDES	Cryptographic operation - TDES
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation], FCS_CKM.4 Cryptographic key destruction.
FCS_COP.1.1/TDES	The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm TDES in *ECB mode*[17] and cryptographic key sizes *168 bit*[18] that meet the following NIST SP 800-67 [26], NIST SP 800-38A [25].
Note:	The 112-bit key option (keying option 2) is supported but for legacy usage only as required by NIST SP 800-67 [26].
Note:	The cryptographic functionality FCS_COP.1/TDES provided by the TOE achieves a security level of maximum 80 Bits if keying option 2 is used.
Note:	The security functionality is resistant against side channel analysis and similar techniques. To fend off attackers with high attack potential a security level of at least 80 Bits must be used.

FCS_CKM.4/TDES

The FCS_CKM.4/TDES is defined in PP [6], §380 and specified below.

FCS_CKM.4/TDES	Cryptographic key destruction - TDES
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4.1/TDES	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *flushing the*

[16] [assignment: *action*]

[17] [selection: *ECB mode, CBC mode*]

[18] [selection: *112 bit, 168 bit*]

internal stored key [19] that meets the following: *standards: none* [20].

FCS_COP.1/AES

The FCS_COP.1/AES is defined in PP [6], §385 and specified below.

FCS_COP.1/AES	Cryptographic operation
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation], FCS_CKM.4 Cryptographic key destruction.
FCS_COP.1.1/AES	The TSF shall perform decryption and encryption in accordance with a specified cryptographic algorithm AES in *ECB mode*[21] and cryptographic key sizes *128, 192 or 256 bit*[22] that meet the following FIPS 197 [23], NIST SP 800-38A [25].

FCS_CKM.4/AES

The FCS_CKM.4/AES is defined in PP [6], §388 and specified below.

FCS_CKM.4/AES	Cryptographic key destruction - AES
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4.1/AES	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *flushing the internal stored key*[23] that meets the following: *standards :none* [24].

As required by Application Note 14 in the PP [6] the secure state is described in Section 7.2.2 in the rationale for SF.OPC.

Regarding Application Note 15 in the PP [6] generation of additional audit data is not defined for requirements FRU_FLT.2 and FPT_FLS.1.

As required by Application Note 19 in the PP [6] the automatic response of the TOE is described in Section 7.2.2 in the rationale for SF.PHY.

[19] [assignment: *cryptographic key destruction method*]

[20] [assignment: *list of standards*]

[21] [selection: *ECB mode, CBC mode*]

[22] [assignment*: cryptographic key sizes*]

[23] [assignment: *cryptographic key destruction method*]

[24] [assignment: *list of standards*]

6.1.2 Additional SFRs regarding access control

Access Control Policy

The hardware shall provide different CPU modes to Boot-ROM Software, Firmware Operating System and Security IC Embedded Software to separate access of these domains to code and data and to control their access to Special Function Registers. Separation of code and data in particular shall be achieved by control of access to memories based on these CPU modes, which is supported by memory partitioning and address mapping based on the CPU modes. Control of access to Special Function Registers shall enforce secure operation on all Special Function Registers depending on their functionality.

In addition, the hardware shall enforce separation of different applications running in User Mode as parts of the Security IC Embedded Software. This shall be achieved by access control to memories and Special Function Registers related to hardware components based on memory segmentation. Access of applications running in User Mode to memories and Special Function Registers related to hardware components shall be denied until explicitly granted by the Security IC Embedded Software running in System Mode.

Access of the Firmware Operating System to (a) Application-Flash and Application-EEPROM for data integrity checking during erase/programming, (b) areas in Application-RAM for data exchange and (c) Special Function Registers related to hardware components shall be denied until explicitly granted by the Security IC Embedded Software running in System Mode.

The hardware shall provide direct memory access to the Security IC Embedded Software without CPU interactions, which is realized by the copy machines, i.e. by both, the Copy Machine and the Full Duplex Copy Machine. These copy machines shall support different CPU modes and memory segmentation in User Mode.

The Security Function Policy (SFP) Access Control Policy uses the following definitions.

The subjects are

- The Security IC Embedded Software i.e. data in the memories of the TOE executed as instructions by the CPU.
- The Test-ROM Software as IC Dedicated Test Software, executed as instructions by the CPU.
- The Boot-ROM Software as part of the IC Dedicated Support Software, executed as instructions by the CPU.
- The Firmware Operating System as part of the IC Dedicated Support Software, executed as instructions by the CPU and stored data integrity monitoring for Flash/EEPROM write accesses of the Security IC Embedded Software.
- The Firmware firewall configured by the Security IC Embedded Software for restricted access of the Firmware Operating System to (a) Application-Flash and Application-EEPROM for data integrity checking during erase/programming, (b) areas in Application-RAM for data exchange and (c) Special Function Registers related to hardware components.
- The Copy Machine and the Full Duplex Copy Machine, both configured by the Security IC Embedded Software for direct memory access enforcing separation of CPU modes and memory segmentation in User Mode.

- The Fame2 coprocessor configured by the Security IC Embedded Software for implementation of asymmetric cryptographic algorithms and direct memory access to the FXRAM for accessing operands and storing resulting data.

The objects are

- the memories consisting of
  - ROM, which is reserved for the IC Dedicated Software.
  - Flash, which is partitioned into partitions reserved for the IC Dedicated Support Software called IC-Dedicated-Support-Software-Flash, and a partition for the Security IC Embedded Software called Application-Flash. The Application-Flash is composed of a ROMinized part and a non-ROMinized part acc. to minor configuration option “ROMinization”.
  - EEPROM, which is partitioned into two parts. To simplify referencing, the part reserved for Firmware Operating System is called Firmware-EEPROM, the other part Application-EEPROM.
  - RAM, which is partitioned into two parts. To simplify referencing, the part reserved for the Firmware Operating System is called Firmware-RAM, the other part Application-RAM.
  - The memory segments defined by the Memory Management Unit in Application-Flash, Application-EEPROM and Application-RAM. Note that this memory is a subset of the above four memories.
  - The TOE Manufacturer EEPROM area, which is part of the Application-EEPROM.
- The physical memory area in Application-Flash, Application-EEPROM and/or Application-RAM that stores the MMU Segment Table.
- The Special Function Registers consisting of
  - Special Function Registers to configure the MMU segmentation. This group contains the registers that define the pointer to the MMU Segment Table.
  - Special Function Registers related to system management. These are intended to be used for overall system management by the operating system.
  - Special Function Registers to configure the Firmware firewall. This group allows the Security IC Embedded Software to modify the Firmware firewall regarding access rights of the Firmware Operating System.
  - Special Function Registers used by the Firmware Operating System.
  - Special Function Registers related to testing. This group is reserved for boot and testing purposes. It also includes a subset of Special Function Registers, which is reserved for test purpose only.
  - Special Function Registers related to hardware components. This group is used to access hardware components like the coprocessors and the interrupt system.
  - Special Function Registers related to general CPU functionality. This group contains e.g. the accumulator, stack pointer and data pointers. This group also includes a subset of Special Function Registers, which is implemented separately for System Mode and User Mode. This subset contains CPU watch exception register for System Mode and User Mode.

The memory operations are

- read data from all memories,
- write data to all memories,

- execute code stored to all memories,
- write-once to TOE Manufacturer EEPROM area,
- protectable[25] write to TOE Manufacturer EEPROM area,
- verify for all zero and programming successful in Flash memory,
- verify for all zero in EEPROM memory.

With regard to read/write operation, if one is allowed, so is the other.

The Special Function Register operations are

- read data from a Special Function Register,
- write data to a Special Function Register.

The security attributes are

- CPU mode: The values of specific Special Function Registers define whether a CPU instruction is executed in either Boot Mode or Test Mode or Firmware Mode or System Mode or User Mode.
- The values of Special Function Registers to configure the MMU segmentation and Special Function Registers related to system management.
- MMU Segment Table: Configuration of up to 64 memory segments, each comprising (a) access rights to the segment, (a) lower and upper virtual memory address of the segment, (c) offset of virtual to physical memory addresses in the segment, (d) access rights of the segment to the Special Function Registers related to hardware components.
- The values of Special Function Registers to configure the Firmware firewall.

In the following the term “code executed” combined with a CPU mode (e.g. “code executedin System Mode”) is used to name subjects.

Note: Use of a Memory Segment is disabled in case no access permissions are granted. It is not necessary to define all 64 possible Memory Segments; the Memory Management Unit is capable of managing an arbitrary number of segments up to the limit of 64.

The TOE shall meet requirements FDP_ACC.1 as specified below.

FDP_ACC.1[MEM]	Subset access control
Hierarchical to:	No other components.
FDP_ACC.1.1[MEM]	The TSF shall enforce the *Access Control Policy*[26] on *all code running on the TOE, all memories and all memory operations*[27].
Dependencies:	FDP_ACF.1 Security attribute based access control
Application Note:	The Access Control Policy shall be enforced by implementing a Memory Management Unit, which maps virtual addresses to physical addresses. The CPU always uses virtual addresses, which are mapped to physical addresses by the Memory Management Unit. Prior to accessing the respective memory

[25] means that each byte in this area can be write protected by another bit that is located in this area.

[26] [assignment: *access control SFP*]

[27] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

address, the Memory Management Unit checks if the access is allowed.

FDP_ACC.1[SFR]	Subset access control
Hierarchical to:	No other components.
FDP_ACC.1.1[SFR]	The TSF shall enforce the *Access Control Policy*[28] on *all code running on the TOE, all Special Function Registers, and all Special Function Register operations*[29].
Dependencies:	FDP_ACF.1 Security attribute based access control
Application Note:	The Access Control Policy shall be enforced by implementing hardware access control to each Special Function Register. For every access the CPU mode is used to determine if the access shall be granted or denied. In addition, in User Mode resp. Firmware Mode the access rights to the Special Function Registers related to hardware components are provided by the MMU Segment Table stored in memory resp. the Special Function Registers to configure the Firmware firewall. A denied read or write access triggers an exception. The read and/or write access to a Special Function Register may be not allowed depending on the function of the register or on the CPU mode to enforce the access control policy or ensure a secure operation. The peripheral access control is enforced by the peripherals the Special Function Registers belong to.

The TOE shall meet requirements FDP_ACF.1 as specified below.

FDP_ACF.1[MEM]	Security attribute based access control
Hierarchical to:	No other components.
FDP_ACF.1.1[MEM]	The TSF shall enforce the *Access Control Policy*[30] to objects based on the following: *all subjects and objects and the attributes CPU mode, the MMU Segment Table, the Special Function Registers to configure the MMU segmentation, Special Function Registers MMU_MXBASH, MMU_MXBASL, MMU_MXSZH, MMU_MXSZL and MMU_FMACC belonging to the group of Special Function Registers to configure the Firmware firewall and the Special Function Registers related to system management*[31].
FDP_ACF.1.2[MEM]	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *Code executed in Boot Mode* • *has read and execute access to all code/data in ROM, except for code/data of the Bootloader Software in ROM,*

[28] [assignment: *access control SFP*]

[29] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

[30] [assignment: *access control SFP*]

[31] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

- *has read and execute access to code/data of the Firmware Operating System in IC-Dedicated-Support-Software-Flash,*
- *has read and write access to all code/data in Firmware-EEPROM,*
- *has read, write and execute access to the TOE Manufacturer EEPROM area,*
- *has read, write and execute access to all data in Firmware-RAM.*

Code executed in Test Mode

- *has read and execute access to all code/data in ROM, but no execute access to code/data of the Bootloader Software in ROM,*
- *has read, write and execute access to all code/data in Flash,*
- *has read, write and execute access to all code/data in EEPROM,*
- *has read, write and execute access to all data in RAM.*

Code executed in Firmware Mode

- *has read and execute access to code/data of the Firmware Operating System in ROM,*
- *has read, write and execute access to all code/data of the Firmware Operating System in IC-Dedicated-Support-Software-Flash,*
- *has read and write access to all code/data in Firmware-EEPROM,*
- *has read and execute access to the TOE Manufacturer EEPROM area and within this TOE Manufacturer EEPROM area*
  - *also write-once access to User Write Once area,*
  - *also protectable write access to NXP Write Protected area,*
  - *also write access to the Bootloader Software area,*
  - *dedicated access restrictions to EEPROM fuses area,*
- *has read, write and execute access to all data in Firmware-RAM.*

Code executed in System Mode

- *has read and execute access to all code/data in the ROMinized part of the Application-Flash,*
- *has read, write and execute access to all code/data in the non-ROMinized part of the Application-Flash,*
- *has read, write and execute access to all code/data in Application-EEPROM, but within TOE Manufacturer EEPROM area*

- *no write access to User Read Only area,*
- *only protectable write access to User Write Protected area,*
- *only write-once access to User Write Once area and not write access to its uppest but three byte,*
- *no write access to NXP Write Protected area but protectable write access to the upper nibble of one byte,*
- *dedicated access restrictions to EEPROM fuses area,*

- *has read, write and execute access to all data in Application-RAM.*

Code executed in User Mode

- *has access permission to all code/data in the ROMinized part of the Application-Flash as controlled by the MMU Segment Table and the Special Function Registers to configure the MMU segmentation which are used by the Memory Management Unit to enforce the following access permissions: "---" (no access), "r--" (read only) and "r-x" (read and execute),*
- *has access permission to code/data in the non-ROMinized part of the Application-Flash as controlled by the MMU Segment Table and the Special Function Registers to configure the MMU segmentation which are used by the Memory Management Unit to enforce the following access permissions: "---" (no access), "r--" (read only) and "r-x" (read and execute),*
- *has access permission to code/data in the Application-EEPROM as controlled by the MMU Segment Table and the Special Function Registers to configure the MMU segmentation which are used by the Memory Management Unit to enforce the following access permissions: "---" (no access), "rw-" (read and write) and "rwx" (no restriction). But within TOE Manufacturer EEPROM area, the following access permissions are enforced regardless of the MMU Segment Table settings:*
  - *no write and execute access to NXP Write Protected area*
  - *not more than dedicated access restrictions to EEPROM fuses area,*
- *has access permission to data in the Application-RAM as controlled by the MMU Segment Table and the Special Function Registers to configure the MMU segmentation which are used by the Memory Management Unit to enforce the following access permissions: "---" (no access), "rw-" (read and write) and "rwx" (no restriction).*[32]

[32] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]

FDP_ACF.1.3[MEM]	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *Code executed in Firmware Mode* • *has verify for all zero or programming successful access to Application-Flash when allowed via Special Function Register MMU_FMACC of the group of Special Function Registers to configure the Firmware firewall,* • *has verify for all zero access to Application-EEPROM when allowed via Special Function Register MMU_FMACC of the group of Special Function Registers to configure the Firmware firewall,* • *has read and write access to data in an area of the Application-RAM that is set in Special Function Registers MMU_MXBASL, MMU_MXBASH, MMU_MXSZL and MMU_MXSZH of the group of Special Function Registers to configure the Firmware firewall.*[33] *The Fame2 coprocessor has read and write access to FXRAM, which is part of the Application-RAM.*
FDP_ACF.1.4[MEM]	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: • *in case minor configuration option "ROM read instructions executed from NVM allowed" is set to "NO", code executed in EEPROM or non-ROMinized part of the Application-Flash cannot read ROM or ROMinized part of the Application-Flash,* • *in case minor configuration option "code execution from RAM allowed" is set to "NO", code cannot be executed in RAM,* • *in case minor configuration option "ROM read instructions by Copy Machine allowed" is set to "NO", the Copy Machine cannot read ROM or ROMinized part of the Application-Flash,* • *in case minor configuration option "NVM read instructions by Copy Machine allowed" is set to "NO", the Copy Machine cannot read EEPROM or non-ROMinized part of the Application-Flash,* • *The Full Duplex Copy Machine has no access to ROM, Flash and EEPROM in Test Mode, Boot Mode, Firmware Mode and System Mode.* [34]
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation
FDP_ACF.1[SFR]	Security attribute based access control
Hierarchical to:	No other components.

[33] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]
[34] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]

FDP_ACF.1.1[SFR] The TSF shall enforce the *Access Control Policy*[35] to objects based on the following: *all subjects and objects and the attributes CPU mode, MMU Segment Table and Special Function Registers MMU_FWCTRLL and MMU_FWCTRLH belonging to the group of Special Function Registers to configure the Firmware firewall*[36].

FDP_ACF.1.2[SFR] The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

- *Code executed in Boot Mode is allowed to access all Special Function Register groups except for Special Function Registers to configure the MMU segmentation, the subset in Special Function Registers related to testing for test purpose and the subset in Special Function Registers related to general CPU functionality that is implemented separately for System Mode and User Mode.*
- *Code executed in Test Mode is allowed to access all groups of Special Function Registers except for Special Function Registers to configure the MMU segmentation and the subset in Special Function Registers related to general CPU functionality that is implemented separately for System Mode and User Mode.*
- *Code executed in Firmware Mode is allowed to read Special Function Registers to configure the Firmware firewall, to access Special Function Registers used by the Firmware Operating System and to access Special Function Registers related to general CPU functionality but its subset implemented separately for System Mode and User Mode. Access to Special Function Registers related to hardware components is based on the access rights set in Special Function Registers MMU_FWCTRLL and MMU_FWCTRLH of the group of Special Function Registers to configure the Firmware firewall.*
- *Code executed in System Mode is allowed to access Special Function Registers to configure the MMU segmentation, Special Function Registers related to system management, Special Function Registers to configure the Firmware firewall, Special Function Registers related to hardware components and Special Function Registers related to general CPU functionality.*
- *Code executed in the User Mode is allowed to access Special Function Registers related to general CPU functionality. Access to Special Function Registers related to hardware components is based on the access rights set*

[35] [assignment: *access control SFP*]

[36] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

for the respective memory segment in the MMU Segment Table from which the code is actually executed. [37]

Application Note: Once started, the Copy Machine continues operation with the access rights to Special Function Registers of the CPU mode or User Mode segment in which it was started, independent of any changes that are afterwards initiated by the Security IC Embedded Software.

FDP_ACF.1.3[SFR] The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *Access to Special Function Registers related to general CPU functionality is allowed in all CPU modes except for access to its subset implemented separately for System Mode and User Mode. Access to this subset is allowed in System Mode and User Mode. Special Function Register CPU_CSR belonging to group Special Function Registers related to system management can also be read in Firmware Mode and User Mode. Special Function Register CFG_CLKSEL of group Special Function Registers related to hardware components can also be read in the Firmware Mode regardless of the access rights set in Special Function Registers MMU_FWCTRLL and MMU_FWCTRLH. Special Function Register MMU_ESIZE of group Special Function Registers related to testing can also be read in Firmware Mode.* [38]

FDP_ACF.1.4[SFR] The TSF shall explicitly deny access of subjects to objects based on the following additional rules:

- *Access to Special Function Registers to configure the MMU segmentation is denied in all CPU modes but System Mode. Access to the subset in Special Function Registers related to general CPU functionality that implemented separately for System and User Mode is denied in Boot Mode, Test Mode and Firmware Mode. Write access to Special Function Registers to configure the Firmware firewall is denied in Firmware Mode. Special Function Register MMU_RPT2 of the group Special Function Registers related to system management is not readable. Special Function Register RNG_RNR of the group Special Function Registers related to hardware components is read-only. Special Function Registers SBC_KEY used as key registers for AES and DES coprocessors of the group Special Function Registers related to hardware components are not readable*[39]*.*
- *The Full Duplex Copy Machine has no access to Special Function Registers.*

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialisation

[37] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]

[38] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]

[39] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]

Implications of the Access Control Policy

The Access Control Policy has some implications, that can be drawn from the policy and that are essential parts of the TOE security functionality.

- Code executed in Boot Mode or Test Mode is quite powerful and used to configure and test the TOE.
- Code executed and data used in Firmware Mode is separated from code executed and data used in System Mode or User Mode. Small memory areas in Application-RAM can be configured by code executed in System Mode, but not by code executed in Firmware Mode, to exchange data. Very limited access of Firmware Mode to Application-Flash and Application-EEPROM must be explicitly granted by code executed in System Mode for data integrity checking during erase/programming.
- Code executed in System Mode can administrate configuration of the Memory Management Unit, since System Mode has access to the Special Function Registers to configure the MMU segmentation, which allows modification of the pointer to the MMU Segment Table. The MMU Segment Table itself can be modified in case this pointer is set to a memory area that can be written in System Mode.
- Code executed in the User Mode cannot administrate configuration of the Memory Management Unit, since User Mode has no access to the Special Function Registers to configure the MMU segmentation, so that the pointer to the MMU Segment Table cannot be modified.
- It may be possible for code executed in User Mode to modify the MMU Segment Table in case this table itself resides in a memory area, which is part of a memory segment that the code in User Mode has write access to.

The TOE shall meet requirements FMT_MSA.3 as specified below.

FMT_MSA.3[MEM]	Static attribute initialisation
Hierarchical to:	No other components.
FMT_MSA.3.1[MEM]	The TSF shall enforce the *Access Control Policy*[40] to provide *restrictive*[41] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2[MEM]	The TSF shall allow ~~the~~ *no subject*[42] to specify alternative initial values to override the default values when an object or information is created.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
Application Note:	Restrictive means here that the reset values of the Special Function Registers to configure the MMU segmentation are set to zero, which effectively disables any memory segment so that no code in User Mode can be executed by the CPU. Furthermore, memory partitions cannot be modified. The TOE does not provide objects or information that can be created, since it provides access to memory areas. The

[40] [assignment: *access control SFP, information flow control SFP*]

[41] [selection, choose one of: *restrictive, permissive,* [assignment: *other property*]]

[42] [assignment: *the authorised identified roles*]

	definition of objects that are stored in the TOE's memory is subject to the Security IC Embedded Software.
FMT_MSA.3[SFR]	Static attribute initialisation
Hierarchical to:	No other components.
FMT_MSA.3.1[SFR]	The TSF shall enforce the *Access Control Policy*[43] to provide *restrictive*[44] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2[SFR]	The TSF shall allow ~~the~~ *no subject*[45] to specify alternative initial values to override the default values when an object or information is created.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
Application Note:	The TOE does not provide objects or information that can be created since no further security attributes can be derived, (i.e. the set of Special Function Registers that contain security attributes is fixed). The definition of objects that are stored in the TOE's memory is subject to the Security IC Embedded Software.

The TOE shall meet requirements FMT_MSA.1 as specified below.

FMT_MSA.1[MEM]	Management of security attributes
Hierarchical to:	No other components.
FMT_MSA.1.1[MEM]	The TSF shall enforce the Access Control Policy[46] to restrict the ability to *modify*[47] the security attributes *Special Function Registers to configure the MMU segmentation*[48] to *code executed in the System Mode*[49].
Dependencies:	[FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
Application Note:	The MMU Segment Table is not included in this requirement because it is located in the memory of the TOE and access to it is possible for every role that has access to the respective memory locations. This component does not include any management functionality for configuration of memory partitioning. This is because memory partitioning is fixed and cannot be changed after TOE Delivery.
FMT_MSA.1[SFR]	Management of security attributes

[43] [assignment: *access control SFP, information flow control SFP*]

[44] [selection, choose one of: *restrictive, permissive,* [assignment*: other property*]]

[45] [assignment: *the authorised identified roles*]

[46] [assignment: *access control SFP(s), information flow control SFP(s)*]

[47] [selection: *change_default, query, modify, delete,* [assignment: *other operations*]]

[48] [assignment: *list of security attributes*]

[49] [assignment: *the authorised identified roles*]

Hierarchical to:	No other components.
FMT_MSA.1.1[SFR]	The TSF shall enforce the *Access Control Policy*[50] to restrict the ability to *modify*[51] the security attributes *defined in Special Function Registers*[52] to *code executed in a CPU mode which has write access to the respective Special Function Registers*[53].
Dependencies:	[FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions

The TOE shall meet requirement FMT_SMF.1 as specified below.

FMT_SMF.1[HW]	Specification of Management Functions
Hierarchical to:	No other components.
FMT_SMF.1.1[HW]	The TSF shall be capable of performing the following management functions: • *change of the CPU mode by writing to the respective bits in Special Function Register CPU_CSR,* • *change of the CPU mode by calling a system vector (SVEC) or firmware vector (FVEC) address,* • *change of the CPU mode with a special LCALL, ACALL or ECALL address,* • *change of the CPU mode by invoking an exception or interrupt,* • *change of the CPU mode by finishing an exception or interrupt (with a RETI instruction),* • *modification of the Special Function Registers containing security attributes,* • *modification of the MMU Segment Table.*[54]
Dependencies:	No dependencies
Application Note:	The iteration of FMT_MSA.1 with the dependency to FMT_SMF.1[HW] may imply a separation of the Specification of Management Functions. Iteration of FMT_SMF.1[HW] is not needed because all management functions rely on the same features implemented in the hardware.

6.2 Security Assurance Requirements

Table 24 lists all security assurance requirements that are valid for this Security Target. These security assurance requirements are defined in the PP “Security IC Platform Protection Profile” [6] and/or in CC part [3] for EAL6, exept for requirements ASE_TSS.2

50 [assignment: *access control SFP(s), information flow control SFP(s)*]

51 [selection: *change_default, query, modify, delete,* [assignment: *other operations*]]

52 [assignment: *list of security attributes*]

53 [assignment: *the authorised identified roles*]

54 [assignment: *list of management functions to be provided by the TSF*]

and ALC_FLR.1, which are augmentations of this Security Target to EAL6, see section 2.2.

ASE_TSS.2 is an augmentation in this Security Target to give architectural information on the security functionality of the TOE. ALC_FLR.1 is an augmentation in this Security Target to cover policies and procedures of the developer applied to track and correct flaws and support surveillance of the TOE.

In compliance with Application Note 22 in the PP [6] the third column in Table 24 shows, which security assurance requirements are added to this Secuirty Target compared to the PP [6]. In this context, entry “EAL6 / PP” means, that the requirerment is defined in both, CC part [3] for EAL6 and the PP [6], entry “EAL6” means, that the requirement is defined in CC part [3] for EAL6 but not in the PP [6], and entry “ST” means, that the requirement is defined neither in CC part [3] for EAL6 nor in the PP [6], but in this Security Target.

All refinements of the security assurance requirements in the PP [6], which must be adapted for EAL6 and/or for this ST, are described in section 6.2.1.

Table 24. SARs for this ST

SAR	Title	Required by
ADV_ARC.1	Security architecture description	EAL6 / PP
ADV_FSP.5	Complete semi-formal functional specification with additional error information	EAL6
ADV_IMP.2	Complete mapping of the implementation representation of the TSF	EAL6
ADV_INT.3	Minimally complex internals	EAL6
ADV_TDS.5	Complete semiformal modular design	EAL6
ADV_SPM.1	Formal TOE security policy model	EAL6
AGD_OPE.1	Operational user guidance	EAL6 / PP
AGD_PRE.1	Preparative procedures	EAL6 / PP
ALC_CMC.5	Advanced support	EAL6
ALC_CMS.5	Development tools CM coverage	EAL6
ALC_DEL.1	Delivery procedures	EAL6 / PP
ALC_DVS.2	Sufficiency of security measures	EAL6 / PP
ALC_FLR.1	Basic flaw remediation	ST
ALC_LCD.1	Developer defined life-cycle model	EAL6 / PP
ALC_TAT.3	Compliance with implementation standards – all parts	EAL6
ASE_CCL.1	Conformance claims	EAL6 / PP
ASE_ECD.1	Extended components definition	EAL6 / PP
ASE_INT.1	ST introduction	EAL6 / PP
ASE_OBJ.2	Security objectives	EAL6 / PP
ASE_REQ.2	Derived security requirements	EAL6 / PP
ASE_SPD.1	Security problem definition	EAL6 / PP
ASE_TSS.2	TOE summary specification with architectural design summary	ST
ATE_COV.3	Rigoruos analysis of coverage	EAL6 / PP
ATE_DPT.3	Testing: modular design	EAL6
ATE_FUN.2	Ordered functional testing	EAL6

SAR	Title	Required by
ATE_IND.2	Independent testing - sample	EAL6 / PP
AVA_VAN.5	Advanced methodical vulnerability analysis	EAL6 / PP

6.2.1 Refinements of the Security Assurance Requirements

In compliance to Application Note 23 in the PP [6] this Security Target has to conform to all refinements of the security assurance requirements in the PP [6]. These refinements are defined for the security assurance requirements of EAL4. Thus, some of these refinements must be adapted to security assurance requirements of higher levels acc. to EAL6 as claimed in this Security Target. All other security assurance requirements defined in this Security Target and in particular the augmentations to EAL6 supplement and extent the security assurance requirements in the PP [6] and can be added without contradictions.

Table 25 lists all security assurance requirements that are refined in the PP [6] based on their definitions in CC part [3] and their effect on this Security Target.

Table 25. SARs refined in the PP [6] and their effect on this ST

Refined SAR in PP [6]	Effect on Security Target
ADV_ARC.1	SAR same as in PP, refinement in PP valid without change
ADV_FSP.4	SAR moves to ADV_FSP.5, refinement in PP valid without change
ADV_IMP.1	SAR moves to ADV_IMP.2, refinement in PP valid without change
AGD_OPE.1	SAR same as in PP, refinement in PP valid without change
AGD_PRE.1	SAR same as in PP, refinement in PP valid without change
ALC_CMC.4	SAR moves to ALC_CMC.5, refinement in PP valid without change
ALC_CMS.4	SAR moves to ALC_CMS.5, refinement in PP valid without change
ALC_DEL.1	SAR same as in PP, refinement in PP adapted for ST
ALC_DVS.2	SAR same as in PP, refinement in PP valid without change
ATE_COV.2	SAR moves to ATE_COV.3, refinement in PP valid without change
AVA_VAN.5	SAR same as in PP, refinement in PP valid without change [55]

All differences in Table 25 are discussed below.

6.2.1.1 Refinement regarding Functional Specification (ADV_FSP.5)

This Security Target requires a higher assurance level for family ADV_FSP compared to the PP [6], namely ADV_FSP.5 instead of ADV_FSP.4. The refinement in section 6.2.1.6 of the PP [6] regarding ADV_FSP.4 addresses the complete representation of the TSF, the purpose and method of use of all TSFIs, and the accuracy and completeness of the SFR instantiations. The refinement is not a change in the wording of the action elements, but a more detailed definition of the items above.

Compared to ADV_FSP.4 component ADV_FSP.5 requires a Functional Specification in a “semi-formal style" (ADV_FSP.5.2C). In addition, component ADV_FSP.5 extends the scope of the error messages to be described from those resulting from an invocation of a TSFI (ADV_FSP.5.6C) to also those not resulting from an invocation of a TSFI (ADV_FSP.5.7C). A rationale shall be provided for these latter ones a (ADV_FSP.5.8C).

55 According to Application Note 29 in the PP [6] the Security Target should indicate the version of the document [5] used for the vulnerability analysis. The current version is given in the bibliography.

Since the higher level ADV_FSP.5 only affects the style of description and the scope of and rationale for error messages, the refinement in the PP [6] regarding ADV_FSP.4 can be applied without changes and is valid for ADV_FSP.5.

6.2.1.2 Refinement regarding Implementation Representation (ADV_IMP.2)

This Security Target requires a higher assurance level for family ADV_IMP compared to the PP [6], namely ADV_IMP.2 instead of ADV_IMP.1. The refinement in section 6.2.1.7 of the PP [6] regarding ADV_IMP.1 states that it must be checked that the provided implementation representation is complete and sufficient to ensure that analysis activities are not curtailed due to lack of information.

This Security Target targets assurance level EAL6 augmented, which requires access to all source code of the TOE so that the above refinement is implicitly fulfilled.

6.2.1.3 Refinement regarding CM capabilities (ALC_CMC.5)

This Security Target requires a higher evaluation level for family ALC_CMC compared to the PP [6], namely ALC_CMC.5 instead of ALC_CMC.4. The refinement in section 6.2.1.4 of the PP [6] regarding ALC_CMC.4 is a clarification of the “TOE” and the term "configuration items".

Since the higher level ALC_CMC.5 requires a higher assurance regarding the defined TOE and the configuration items, the refinement in the PP [6] regarding ADV_CMC.4 can be applied without changes and is valid for ADV_CMC.5.

6.2.1.4 Refinement regarding CM scope (ALC_CMS.5)

This Security Target requires a higher evaluation level for family ALC_CMS compared to the PP [6], namely ALC_CMS.5 instead of ALC_CMS.4. The refinement in section 6.2.1.3 of the PP [6] regarding ALC_CMS.4 is a clarification of the configuration item “TOE implementation representation”.

Compared to ALC_CMS.4 component ALC_CMS.5 only adds the requirement for a new configuration item to be included in the configuration list (ALC_CMS.51C) so that the refinement in the PP [6] regarding ADV_CMS.4 can be applied without changes and is valid for ADV_CMS.5.

6.2.1.5 Refinement regarding Development Security (ALC_DEL.1)

This Security Target requires the same evaluation level for family ALC_DEL like the PP [6], namely ALC_DEL.1.

The refinement in section 6.2.1.1 of the PP [6] regarding ALC_DEL.1 is valid and is extended according to application note 25 in the PP [6]. This extension is done for service Trust Provisioning as ordered in [17]. This extention addresses the interfaces of the TOE Manufacturer to the Composite Prodcuct Manufacturer in terms of Security IC Embedded Software Development (phase 1) as well as IC Packaging (phase 4) resp. Composite Product Integration (phase 5) according to the Security IC product life cycle in the PP [6]. In this context, the interface to Security IC Embedded Software Development is extended by definitions of key data formats and related data like storage location in Flash and/or EEPROM. The interface to IC Packaging resp. Composite Product Integration is extended by the keys, which are defined by the Composite Product Manufacturer. These keys are part of the User Data. The TOE Manufacturer provides the services for generating, programming and distributing keys as part of the pre-personalisation data of the Security IC Embedded Software.

Application Note: Generation, programming and distribution of keys as part of the pre-personalisation data is taken into account under ALC_DVS.2 in order to ensure

confidentiality and integrity during programming into Flash and/or EEPROM as well as during distribution of the keys.

6.2.1.6 Refinement regarding Test Coverage (ATE_COV.3)

This Security Target requires a higher evaluation level for family ALC_COV compared to the PP [6], namely ALC_COV.3 instead of ALC_COV.2. The refinement in section 6.2.1.8 of the PP [6] regarding ALC_COV.2 defines that test coverage must include different operating conditions and “ageing” and that existence and effectiveness of countermeasures against physical attacks cannot be tested but must be given by evidence.

The refinement regarding test coverage is not a change in the wording of the action elements, but a more detailed definition of the items to be applied, so that it can be applied without changes and is valid for ADV_COV.3. The refinement regarding existence and effectiveness of countermeasures against physical attacks is implicitly fulfilled since his Security Target targets assurance level EAL6 augmented, which requires access to all source code and layout data.

6.2.2 Definition of ADV_SPM

The developer shall provide a formal security policy model for the *Access Control Policy*.[56]

The Access Control Policy comprises the following Security Functional Requirements: FDP_ACC.1[MEM], FDP_ACC.1[SFR], FDP_ACF.1[MEM], FDP_ACF.1[SFR] with the associated dependencies. Further on, the secure state as required by FPT_FLS.1 is included in the security policy model. In addition, parts of the life cycle control as required by FMT_LIM.2 and limited capabilities as required by FMT_LIM.1, and FMT_LIM.2/Loader and FMT_LIM.1/Loader with respect to the Bootloader locking are part of the model.

6.3 Security Requirements Rationale

6.3.1 Rationale for the security functional requirements

Section 6.3.1 and sections 7.3.1. 7.4.1 and 7.4.2 due to the chosen packages in the PP [6] provide a rationale for the mapping between security functional requirements and security objectives defined in the PP [6]. The mapping is reproduced in the following table.

Table 26. Security Requirements versus Security Objectives

Objective	TOE Security Functional Requirements
O.Leak-Inherent	FDP_ITT.1[HW], FPT_ITT.1[HW], FDP_IFC.1
O.Phys-Probing	FPT_PHP.3, FDP_SDC.1
O.Malfunction	FRU_FLT.2, FPT_FLS.1
O.Phys-Manipulation	FPT_PHP.3 (as in O.Phys-Probing), FDP_SDI.2
O.Leak-Forced	FDP_ITT.1[HW], FPT_ITT.1[HW], FDP_IFC.1 (as in O.Leak-Inherent) FPT_PHP.3 (as in O.Phys-Probing) FRU_FLT.2, FPT_FLS.1 (as in O.Malfunction)
O.Abuse-Func	FMT_LIM.1, FMT_LIM.2 FDP_ITT.1[HW], FPT_ITT.1[HW], FDP_IFC.1 (as in O.Leak-Inherent) FPT_PHP.3 (as in O.Phys-Probing)

[56] [assignment*: list of policies that are formally modelled*].

Objective	TOE Security Functional Requirements
	FRU_FLT.2, FPT_FLS.1 (as in O.Malfunction)
O.Identification	FAU_SAS.1[HW]
O.RND	FCS_RNG.1[HW] FDP_ITT.1[HW], FPT_ITT.1[HW], FDP_IFC.1 (as in O.Leak-Inherent) FPT_PHP.3 (as in O.Phys-Probing) FRU_FLT.2, FPT_FLS.1 (as in O.Malfunction)
O.Cap_Avail_Loader	FMT_LIM.1/Loader, FMT_LIM.2/Loader
O.TDES	FCS_COP.1/TDES FCS_CKM.4/TDES
O.AES	FCS_COP.1/AES FCS_CKM.4/AES

The Security Target extends SFR defined in the PP [6] and additionally defines SFRs as listed in Table 27. The following table gives an overview, how the requirements are combined to meet the security objectives.

Table 27. Mapping of security objectives and requirements

Objective	TOE Security Functional Requirement
O.NVM_INTEGRITY	FDP_SDI.2[HW]
O.FM_FW	FDP_ACC.1[MEM], FDP_ACF.1[MEM], FMT_MSA.3[MEM]
O.MEM_ACCESS	FDP_ACC.1[MEM], FDP_ACF.1[MEM], FMT_MSA.3[MEM], FMT_MSA.1[MEM], FMT_MSA.1[SFR], FMT_SMF.1[HW]
O.SFR_ACCESS	FDP_ACC.1[SFR], FDP_ACF.1[SFR], FMT_MSA.3[SFR], FMT_MSA.1[SFR], FMT_SMF.1[HW]

The justification related to security objective O.NVM_INTEGRITY is as follows:

SFR FDP_SDI.2[HW] requires a control function, that informs the Security IC Embedded Software and in case of EEPROM also adjusts the conditions of an EEPROM block so that integrity of the data read from Flash can be ensured by the Security IC Embedded Software and integrity of the data read from EEPROM is ensured by the TOE. This is true even if the characteristics of the memory changed e.g. due to ageing. Therefore, SFR FDP_SDI.2[HW] is suitable to meet O.NVM_INTEGRITY.

The justification related to security objective O.FM_FW is as follows:

SFR FDP_ACC.1[MEM] with the related SFP “Access Control Policy” exactly requires to implement memory partitioning as demanded by O.FM_FW. Therefore, SFR FDP_ACC.1[MEM] with its SFP is suitable to meet O.FM_FW.

SFR FDP_ACF.1[MEM] with the related SFP “Access Control Policy” defines the rules to implement memory partitioning as demanded by O.FM_FW. Therefore, FDP_ACF.1[MEM] with its SFP is suitable to meet O.FM_FW.

SFR FMT_MSA.3[MEM] requires that the TOE provides restrictive default values for the security attributes used by the Memory Managmeent Unit to enforce memory partitioning. In this context, restrictive with respect to memory partitioning means that memory partitioning cannot be changed at all. Therefore, SFR FMT_MSA.3 (as dependency from FDP_ACF.1) is suitable to meet O.FM_FW.

SFR FMT_MSA.1 requires that the ability to update the security attributes is restricted to privileged subject(s). No management ability is specified in there that can be used to change memory partitioning. Also no related management function is specified in SFR FMT_SMF.1[HW]. Thus, memory partitioning is fixed and cannot be changed any subject, which is required by O.FM_FW.

The justification related to security objective O.MEM_ACCESS is as follows:

SFR FDP_ACC.1[MEM] with the related SFP “Access Control Policy” exactly require to implement an area based memory access control as demanded by O.MEM_ACCESS. Therefore, FDP_ACC.1[MEM] with its SFP is suitable to meet O.MEM_ACCESS.

SFR FDP_ACF.1[MEM] with the related SFP “Access Control Policy” defines the rules to implement the area based memory access control as demanded by O.MEM_ACCESS. Therefore, SFR FDP_ACF.1[MEM] with its SFP is suitable to meet O.MEM_ACCESS.

SFR FMT_MSA.3[MEM] requires that the TOE provides restrictive default values for the security attributes used by the Memory Management Unit. Default values of the relevant Special Function Registers are set during reset of the TOE the way that the Memory Management Unit always starts in restrictive default configuration. Therefore, SFR FMT_MSA.3[MEM] (as dependency from FDP_ACF.1) is suitable to meet O.MEM_ACCESS.

SFR FMT_MSA.1 requires that the ability to update the security attributes is restricted to privileged subject(s). These management functions ensure that the required access control can be realized using the functions provided by the TOE. The iteration of FMT_MSA.1 into FMT_MSA.1[MEM] and FMT_MSA.1[SFR] are needed because the different types of objects have different security attributes. The security attributes of the Memory Management Unit can be changed by the Security IC Embedded Software. Since the pointer to the MMU Segment Table can only be changed in System Mode and this protection is implemented by access control to the respective Special Function Registers, both iterations are needed for O.MEM_ACCESS.

SFR FMT_SMF.1 specifies the management functions to be provided by the TOE as required by O.MEM_ACCESS. Therefore, FMT_SMF.1[HW] is suitable to meet to O.MEM_ACCESS.

The justification related to security objective O.SFR_ACCESS is as follows:

SFR FDP_ACC.1[SFR] with the related SFP “Access Control Policy” requires to implement access control to Special Function Register as demanded by O.SFR_ACCESS. Therefore, SFR FDP_ACC.1[SFR] with its SFP is suitable to meet O.SFR_ACCESS.

SFR FDP_ACF.1[SFR] with the related SFP “Access Control Policy” exactly requires certain security attributes to implement access control to Special Function Registers as required by O.SFR_ACCESS. Therefore, FDP_ACF.1[SFR] with its SFP is suitable to meet O.SFR_ACCESS.

SFR FMT_MSA.3[SFR] requires that the TOE provides default values for the Special Function Registers. Default values of the Special Function Registers are set during reset of the TOE. These are needed to ensure a defined start-up of the TOE. Therefore, SFR FMT_MSA.3[SFR] is suitable to meet O.SFR_ACCESS.

SFR FMT_MSA.1[SFR] is realised in a way that – apart from the definition of access rights to Special Function Registers related to hardware components in User Mode and Firmware Mode - no management of the security attributes is possible because the attributes are implemented in the hardware and cannot be changed.

SFR FMT_SMF.1[HW] specifies the management functions to be provided by the TOE as demanded by O.SFR_ACCESS. Therefore, FMT_SMF.1[HW] is suitable to meet O.SFR_ACCESS.

Note that the iteration of FDP_ACF.1 and FDP_ACC.1 with the respective dependencies are needed to separate the different types of objects because they have different security attributes.

6.3.2 Dependencies of security functional requirements

The dependencies of SFRs listed in the PP [6] are independent of the additional dependencies listed in the table below. The dependencies of the PP [6] are fulfilled within the PP [6] and at least one dependency is considered to be satisfied.

The following discussion demonstrates how the dependencies defined by CC Part 2 [2] for the requirements specified in sections 6.1.1 and 6.1.2 are satisfied.

The dependencies defined in the Common Criteria are listed in the table below:

Table 28. Dependencies of security functional requirements

Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
FCS_COP.1/TDES	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	See discussion below
FCS_CKM.4/TDES	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	See discussion below
FCS_COP.1/AES	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	See discussion below
FCS_CKM.4/AES	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	See discussion below
FDP_ACC.1[MEM]	FDP_ACF.1	Yes, by FDP_ACF.1[MEM]
FDP_ACC.1[SFR]	FDP_ACF.1	Yes, by FDP_ACF.1[SFR]
FDP_ACF.1[MEM]	FDP_ACC.1 FMT_MSA.3	Yes, by FDP_ACC.1[MEM] Yes
FDP_ACF.1[SFR]	FDP_ACC.1 FMT_MSA.3	Yes, by FDP_ACC.1[SFR] Yes
FMT_MSA.3[MEM]	FMT_MSA.1 FMT_SMR.1	Yes, by FMT_MSA.1[MEM] See discussion below
FMT_MSA.3[SFR]	FMT_MSA.1 FMT_SMR.1	Yes, by FMT_MSA.1[SFR] See discussion below
FMT_MSA.1[MEM]	FDP_ACC.1 or FDP_IFC.1 FMT_SMR.1 FMT_SMF.1[HW]	Yes, by FDP_ACC.1[MEM] See discussion below Yes
FMT_MSA.1[SFR]	FDP_ACC.1 or FDP_IFC.1 FMT_SMR.1 FMT_SMF.1[HW]	Yes, by FDP_ACC.1[SFR] See discussion below Yes

The developer of the Security IC Embedded Software must ensure that the SFR FCS_COP.1/TDES, FCS_COP.1/AES, FCS_CKM.4/TDES and FCS_CKM.4/AES are used as specified and that the user data processed by the related security functionality is protected as defined for the application context.

The dependent requirements of FCS_COP.1/TDES, FCS_COP.1/TDES, FCS_CKM.4/TDES and FCS_CKM.4/AES completely address the appropriate management of cryptographic keys used by the specified cryptographic function and the management of access control rights as specified for the memory access control function. All requirements concerning these management functions shall be fulfilled by the environment (Security IC Embedded Software).

SFRs [FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] are not included in this Security Target since the TOE only provides a pure engine for encryption and decryption without additional features for the handling of cryptographic keys except key destruction. Therefore, the Security IC Embedded Software must fulfil these requirements related to the needs of the realized application.

The dependency of SFRs FMT_MSA.1 and FMT_MSA.3 to SFR FMT_SMR.1 must be fulfilled by the Security IC Embedded Software. The definition and maintenance of the roles that act on behalf of the functions provided by the hardware must be subject of the Security IC Embedded Software.

6.3.3 Rationale for the Assurance Requirements

The selection of assurance components is based on the underlying PP [6]. The Security Target uses the same augmentations as the PP, but chooses a higher assurance level. The level EAL6 is chosen in order to meet assurance expectations of digital signature applications and electronic payment systems. Additionally, the requirement of the PP [6] to choose at least EAL4 is fulfilled.

The rationale for the augmentations is the same as in the PP. The assurance level EAL6 is an elaborated pre-defined level of the CC, part 3 [3]. The assurance components in an EAL level are chosen in a way that they build a mutually supportive and complete set of components. The requirements chosen for augmentation do not add any dependencies, which are not already fulfilled for the corresponding requirements contained in EAL 6. Therefore, these components add additional assurance to EAL 6, but the mutual support of the requirements is still guaranteed.

As stated in the section 6.3.3 of the PP [6], it has to be assumed that attackers with high attack potential try to attack smartcards used for digital signature applications or payment systems. Therefore, specifically AVA_VAN.5 was chosen by the PP [6] in order to assure that even these attackers cannot successfully attack the TOE.

6.3.4 Security Requirements are internally Consistent

The discussion of security functional requirements and security assurance components in the preceding sections has shown that mutual support and consistency are given for both groups of requirements. The arguments given for the fact that the assurance components are adequate for the functionality of the TOE also show that the security functional and assurance requirements support each other and that there are no inconsistencies between these groups.

The SFRs required to meet O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced also protect the cryptographic algorithms, the integrity support of data stored in Flash and EEPROM and the memory access/separation control function as well as the access control to Special Function Register implemented

according to the security functional requirement FCS_COP.1/AES, FDP_SDI.2[HW] and FDP_ACC.1[MEM], FDP_ACC.1[SFR] with reference to the Access Control Policies defined in FDP_ACF.1[MEM] and FDP_ACF.1[SFR]. Therefore, these SFRs support secure implementation and operation of SFRs FCS_COP.1/TDES, FCS_COP.1/AES, of SFR FDP_ACC.1 with SFR FDP_ACF.1 and of the dependent SFRs.

The extension of SFRs FDP_ITT.1[HW] and FTP_ITT.1[HW] compared to the PP [6] adds the detection of faults during transfer of User Data or TSF data between internal components of the TOE. The protection against leakage is not weakened by this extension.

A hardware platform including the IC Dedicated Software requires Security IC Embedded Software to build a secure product. Thereby the Security IC Embedded Software must support the security functionality of the hardware as well as the IC Dedicated Support Software and implement a sufficient management of the security services implemented by the hardware platform including the IC Dedicated Software. The realisation of the Security Functional Requirements within the TOE provides a good balance between flexible configuration and restrictions to ensure a secure behaviour of the TOE.

7. TOE Summary Specification

This chapter is composed of sections “Portions of the TOE Security Functionality” and “TOE Summary Specification Rationale”.

7.1 Portions of the TOE Security Functionality

The TOE Security Functionality (TSF) directly corresponds to the TOE security functional requirements defined in chapter 6. This TOE Security Functionality (TSF) is split into Security Services (SS) and Security Features (SF). The whole security functionality is active at TOE Delivery, i.e. after phase 3 or phase 4 of the Security IC product life cycle in the PP [6], depending on the package type of the TOE.

The TOE also comprises security mechanisms, which are not part of its Security Services and Security Features. Such mechanisms can be used by the Security IC Embedded Software to implement new Security Services and/or Security Features into a composite product, e.g. use of the Fame2 coprocessor to implement leakage-resistant asymmetric cryptographic algorithms.

7.1.1 Security Services

SS.RNG: Random Number Generator

The Random Number Generator continuously produces random numbers with a length of one byte. The TOE implements SS.RNG by means of a physical hardware Random Number Generator working stable within the valid ranges of operating conditions, which are guaranteed by SF.OPC.

The TSF provides a hardware test functionality, which can be used by the Security IC Embedded Software to detect hardware defects and bad quality of the random numbers.

According to AIS31 [7] the Random Number Generator claims functionality class PTG2. This means that the Random Number Generator is suitable for generation of signature key pairs, generation of session keys for symmetric encryption mechanisms, random padding bits, zero-knowledge proofs, generation of seeds for DRNGs and fulfils the online test requirements defined in AIS31 [7].

SS.HW_DES: Triple-DES coprocessor

The TOE provides the Triple Data Encryption Algorithm (TDEA) according to the Data Encryption Standard (DES). SS.HW_DES is a modular basic cryptographic function, which provides the TDEA algorithm as defined in NIST SP 800-67 [26] by means of a hardware coprocessor and supports (a) the 3-key Triple-DEA algorithm according to keying option 1 and (b) the 2-key Triple DEA algorithm according to keying option 2 in NIST SP 800-67 [26]. The two/three 56-bit keys (112-/168-bit) for the 2-key/3-key Triple-DES algorithm shall be provided by the Security IC Embedded Software. SS.HW_DES performs hardware XOR-operation of two data blocks to support chaining modes of the TDES when configured by the Security IC Embedded Software.

SS.HW_AES: AES coprocessor

The TOE provides the Advanced Encryption Standard (AES) algorithm according to the Advanced Encryption Standard as defined by FIPS PUB 197 [23]. SS.HW_AES is a modular basic cryptographic function, which provides the AES algorithm by means of a hardware coprocessor and supports the AES algorithm with three different key lengths of 128, 192 or 256 bit. The keys for the AES algorithm shall be provided by the Security IC Embedded Software. SS.HW_AES performs hardware XOR-operation of two data blocks

to support chaining modes of the AES when configured by the Security IC Embedded Software.

7.1.2 Security Features

SF.OPC: Control of Operating Conditions

SF.OPC ensures correct operation of the TOE (functions offered by the microcontroller including the standard CPU as well as Triple-DES coprocessor, AES coprocessor, Fame2 coprocessor, memories, registers, I/O interfaces and the other system peripherals) during execution of IC Dedicated Support Software and Security IC Embedded Software. This includes all security mechanisms of the TOE, which directly contribute to a Security Service or a Security Feature.

The TOE ensures its correct operation and prevents any malfunction using the following mechanisms: filtering of power supply, clock frequency and reset input signals as well as monitoring of voltage supplies, clock frequencies and on-chip temperature by means of sensors. There are multiple voltage sensors for the different ISO/IEC 7816 voltage classes. Light sensors are distributed over the chip surface and are used to detect light attacks. Flash and EEPROM in particular provide additional functions to detect light attacks of which the EEPROM light attack detection function is controlled by the Security IC Embedded Software.

Further on, the Security IC Embedded Software is provided with the Secure Fetch to protect transfer of code and data to the CPU and with a watchdog timer to protect code execution.

Specific functional units of the TOE are equipped with further fault injection detection. This comprises dedicated checks for processing faults in Tripple-DES, AES and Fame2 coprocessors as well as checks on program counter, stack pointers, upper and lower limits of the stack pointers, several CPU control registers, MMU address and data cache registers, hardware configuration registers, control registers for the SBC interface to access Tripple-DES and AES coprocessors, for Fame2 coprocessor and for the Copy Machine.

Finally, minor configuration option “Inverse EEPROM Error Correction Attack Detection activated” can be set to “YES” to increase the probability to detect fault injections to EEPROM memory and interface.

In case a monitored parameter is out of specified range or a fault is detected, the TOE either (i) aborts code execution and forces a reset or (ii) raises an exception, which interrupts code execution and jumps to an exception vector so that the Security IC Embedded Software can react by an appropriate exception routine. In case of reset the TOE is reset to its initial state and provides information on the reset source to the Security IC Embedded Software. In case of exception the TOE jumps to the exception vector and provides information on the exception source to the Security IC Embedded Software.

SF.PHY: Protection against Physical Manipulation

SF.PHY protects the TOE against manipulation of (i) the IC hardware, (ii) the IC Dedicated Software, (iii) the Security IC Embedded Software and (iv) User Data in Flash, EEPROM and RAM as well as TSF data. It also protects User Data and TSF data against disclosure by physical probing when stored to or while being processed by the TOE.

This protection comprises several security mechanisms in design and construction, which effectively hamper reverse-engineering and tamper attacks. These mechanisms

include dedicated shielding techniques for the IC hardware and specific encryption mechanisms for all memories and internal buses.

Further on, SF.PHY checks integrity of the content in all memories. ROM is equipped with a partiy check during read access that forces the TOE to reset on failure. Flash and EEPROM are each provided with an automatic error correction of 1 bit every intrinsic word size. Bit errors beyond these, that are detected but can't be corrected, are acknowledged by reset. CXRAM and FXRAM are each equipped with a parity watchdog, which continuously scans for integrity of its content. Any discrepancy forces the TOE to reset. In addition, a parity check on content read by the Security IC Embedded Software can be enabled separately for CXRAM and FXRAM via minor configuration options. Such parity check also replies to failures by reset.

SF.PHY supports the efficiency of other portions of the TOE security functionality.

SF.LOG: Logical Protection

SF.LOG implements security mechanisms to limit or even eliminate information in the shape and amplitude of signals or in the time between events. These might be measurable on signals like power supply, on signals at other pads that are not intentionally used for communication as well as on emanation of the IC hardware. In this context, SF.LOG prevents from disclosure of User Data and TSF data stored to and/or processed by the TOE through measurement of power consumption or emanation and subsequent complex signal analysis. This protection of the TOE is enforced by several security mechanisms in the design.

The Triple-DES coprocessor includes specific security mechanisms to prevent SPA/DPA analysis of shape and amplitude of the power consumption and emanation. The implementation of the Triple-DES coprocessor further ensures that the calculation time is independent from the chosen key value and plain or cipher text.

The AES coprocessor includes specific security mechanisms to prevent SPA/DPA analysis of shape and amplitude of the power consumption and emanation. The implementation of the AES coprocessor further ensures that the calculation time is independent from the chosen key and any plain or cipher text for a given key length.

The Fame2 coprocessor provides measures to prevent timing attacks on basic modular function. The calculation time of an operation depends on the lengths of the operands, but not on the value of the operands, with the following exceptions: multiplication with reduction, modular inversion and modular division. These three operations have no constant timing because of correction cycles that are needed for the calculation methods. In addition, mechanisms are included, which limit the capability to analyse shape and amplitude of power consumption. However, the Fame2 coprocessor does not realise an algorithm on its own and algorithm-specific leakage countermeasures have to be added by the Security IC Embedded Software when using the Fame2 coprocessor.

Further on, the Security IC Embedded Software can use clock configurations to hamper synchronization of internal operations to an external clock or to characteristics of the power consumption that can be used as trigger signal to support leakage attacks like DPA or timing attacks.

SF.COMP: Protection of Mode Control

SF.COMP provides control of the CPU modes within Super System Mode and transitions to Boot Mode.

- Boot Mode is always entered with start-up or reset of the TOE,

- Boot Mode cannot be entered but by via start-up or reset of the TOE,
- Boot Mode can switch to Test Mode or Firmware Mode,
- Firmware Mode cannot switch to Test Mode,
- Test Mode cannot be entered after TOE Delivery and the Test-ROM Software is permanently disabled.

The above rules prevent abuse of test functions after TOE Delivery and abuse of mechanisms, which are used during start-up or reset to configure the TOE to its initial state. These rules also ensure, that after TOE Delivery and then each time the TOE completed start-up or reset, Firmware Mode is the only Super System Mode available.

Further on, SF.COMP terminates permanently the Bootloader Software before TOE Delivery and therewith prevents abuse of Bootloader functionality after TOE Delivery.

SF.COMP also provides the TOE Manufacturer EEPROM area, which is located in the the top-most 768 Bytes of the Application-EEPROM and accessible via reserved logical addresses in the memory map. A dedicated access control is applied to the TOE Manufacturer EEPROM area based on CPU modes and on configuration of address mapping to memory partitions in System Mode and User Mode.

The EEPROM fuses in the TOE Manufacturer EEPROM area are also subject to an integrity control, which ensures secure storage of configuration and calibration data as well as their secure setup during start-up or reset of the TOE.This particularly enforces that the configuration of security functionality can not be modified, abused or manipulated so that the TSF provides self-protection against interference and tampering by untrusted Security IC Embedded Software.

The TOE Manufacturer EEPROM area also provides three memory areas for use by the Security IC Embedded Software. These are the User Read Only area, the User Write Protected area and the User Write Once area. The User Read Only area contains 32 bytes, which are read-only for the Security IC Embedded Software. The User Write Protected area contains 16 bytes, which can be write-protected by the Security IC Embedded Software on demand. The User Write Once area contains 32 bytes of which each bit can separately be set to ‘1’ once only, and not reset to ‘0’.

Some bytes in the User Write Once area are utilized by minor configuration options. In case minor configuration option “Activation of “Card Disable” feature allowed” is set to “YES” the Security IC Embedded Software can inhibit any further start-up after next reset by setting a corresponding byte. In case minor configuration option “EEPROM application content erase allowed” is set to “YES” the Security IC Embedded Software can destroy all contents stored to Application-Flash and Application-EEPROM by setting a corresponding byte.

SF.COMP also provides the FabKey area in Application-EEPROM to store initialisation and/or pre-personalisation data. The FabKey area as well as the TOE Manufacturer EEPROM area can be used for die-individual identification. The User Write Protected area and the User Write Once area are designed to store the identification of a (fully personalised) Security IC or a sequence of events over the life cycle, that can be coded by an increasing number of bits set to "one" or protecting bytes, respectively.

User Read Only area, User Write Protected area and User Write Once area in the TOE Manufacturer EEPROM area, the Fabkey area and also contents in Application-Flash and other Application-EEPROM are defined by the Composite Product Manufacturer acc. to the Order Entry Form [17]. Such contents may include identification and/or pre-personalisation data and/or Security IC Embedded Software. Some values of the

EEPROM fuses in the TOE Manufacturer EEPROM area also depend on configuration options in the Order Entry Form [17]. All the contents from the Composite Product Manufacturer as well as all the other contents of the TOE Manufacturer EEPROM area are stored to memories during phase 3 IC Manufacturing.

SF.MEM_ACC: Memory Access Control

SF.MEM_ACC controls access of CPU instructions to the memories of the TOE through the Memory Management Unit. In this context, the CPU always uses virtual addresses. The Memory Management Unit maps these virtual addresses to the physical addresses of the memories, which are then passed to the memory interfaces for access. Access control is established on two principles, which are

- Memory partitioning: Each memory is partitioned as described in the Access Control Policy in section 6.1.2. The physical addresses of these partitions are mapped to virtual addresses depending on the CPU mode and on configuration of address mapping to memory partitions in System Mode and User Mode so that the partitions reserved for IC Dedicated Software are not visible at all when Security IC Embedded Software is running in System Mode or User Mode. In addition, visible memory partitions and in particular those accessible in System Mode and User Mode to the Security IC Embedded Software are under access control based on the CPU mode.
- Memory segmentation in User Mode: The three accessible parts of the memory in ROM, RAM and EEPROM can be segmented into smaller memory areas. Access rights including “---” (no access), “r--” (read only), and “r-x” (read and execute) can be defined for the ROM segments. Access rights including “---” (no access), “rw-” (read and write) and “rwx” (no restriction) can be defined for the EEPROM segments. Access rights including “---” (no access), “rw-” (read and write) and “rwx” (no restriction) can be defined for the RAM segments. In addition, access rights to Special Function Registers related to hardware components can be defined for code executed in User Mode.

Memory partitions are fixed and cannot be changed. They are set during production of the TOE and solely may depend on major and minor configurations of the TOE. Mapping of partitions to virtual addresses in Sytem Mode and User Mode can be configured so that either Security IC Embedded Software or Bootloader Software is mapped to address 80:0000h in System Mode. Control of access to memory partitions accessible in System Mode and User Mode to the Security IC Embedded Software is based on acess rights, which are fixed except for those for Firmware Mode to

(i) read/write access to an area in Application-RAM, when granted in System Mode as defined in Special Function Registers MMU_MXBASL, MMU_MXBASH, MMU_MXSZL and MMU_MXSZH of the group of Special Function Registers to configure the Firmware firewall,

(ii) verify for all zero access to Application-EEPROM and verify for all zero and programming successful access to Application-Flash when granted in System Mode via Special Function Register MMU_FMACC of the group of Special Function Registers to configure the Firmware firewall.

Memory segmentation is active when the CPU switches to User Mode. The segments are defined in System Mode in the MMU Segment Table, which is located in memory. The table can be split into several parts being stored to different locations in memory even so that it is placed in segments accessible to User Mode. The definition of each memory segment also includes whether access to Special Function Registers related to

hardware components is granted or denied for code running this segment. Memory segmentation in User Mode cannot overrule memory partitioning.

Access violations in System Mode and User Mode are reported in an exception including access to memory addresses that do not point to physically implemented memory.

SF.MEM_ACC also controls access of copy machines to the memories of the TOE through the Memory Management Unit. The access rights of the Copy Machine to memories are defined when the Copy Machine is started. The Copy Machine then is assigned with the CPU mode, which starts the Copy Machine. The access right of the Full Duplex Copy Machine to memories are defined when its XDATA start address is set. The Full Duplex Copy Machine then is assigned with the CPU mode, which sets its XDATA start address. This means, the copy machines keep the access rights of the assigned CPU mode independent of proceeding CPU mode transitions. Access violations are reported in an exception.

SF.MEM_ACC provides SF.SFR_ACC with access rights in Firmware Mode and User Mode to Special Function Registers related to hardware components.

SF.SFR_ACC: Special Function Register Access Control

SF.SFR_ACC controls access to Special Function Registers based on CPU modes and CPU mode transitions based on specific Special Function Registers.

Some Special Function Registers are implemented thrice, one for User Mode, a second one for System Mode and a third one for Super System Mode. Such Special Function Registers are inherently separated so that each CPU mode has its own register. Other Special Function Registers are implemented once for all CPU modes. Then, the purpose of a Special Function Register and the CPU mode determine, whether read and/or write access is allowed or not. For example, key registers of the SBC interface are write-only in all CPU modes to support confidentiality of the keys, and the output register of the Random Number Generator is read-only in all CPU modes to protect from modification. Access rights to Special Function Registers versus CPU modes for example are implemented as required by SFRs FDP_ACC.1[SFR] and FDP_ACF.1[SFR].

Access rights to Special Function Registers are pre-defined and cannot be configured, except for Special Function Registers related to hardware components. These are accessible in System Mode, but, by default, not in Firmware Mode and User Mode. Such access must explicitly be granted by Security IC Embedded Software running in System Mode. This also implies that both, User Mode and Firmware Mode are not able to use e.g. SS.RNG, SS.HW_DES and SS.HW_AES until access to their Special Function Registers is granted. An exception is raised in case an access is not allowed or the addressed Special Function Register is not implemented. The Security IC Embedded Software can react on such exception.

The Full Duplex Copy Machine has no access to Special Function Registers. These can only be accessed by CPU instructions and by the Copy Machine. The access rights of the Copy Machine are defined when the Copy Machine is started. Then, the same access rights are assigned, which are valid for the CPU mode that starts the Copy Machine. This means, the Copy Machine started in System Mode has full access rights of System Mode, whereas the Copy Machine started in User Mode has the access rights of the User Mode segment that started the Copy Machine. Independent of CPU mode transitions following the start of the Copy Machine its access rights once assigned are valid during its whole transaction until transaction is completed. Access violations are reported in an exception.

SF.SFR_ACC also implements transitions among CPU modes based on specific Special Function Registers as follows.

- Write access to a bit in Special Function Register CPU_CSR. This bit can be accessed in System Mode, but not in User Mode, so that System Mode can switch to User Mode but User Mode cannot use this bit to switch back to System Mode.
- Call to a system vector (SVEC) address or to a firmware vector (FVEC) address. An SVEC call activates System Mode, an FVEC call activates Firmware Mode. SVEC calls are allowed in User Mode, an exception is raised when called in System Mode. A return address is pushed onto the stack.
- LCALL/ACALL/ECALL instruction call to address 80:000h in Boot Mode activates System Mode and jumps to this virtual address. No return address is pushed onto the stack.
- LCALL/ACALL/ECALL instruction call to address 80:000h in System Mode activates User Mode and jumps to this virtual address. No return address is pushed onto the stack.
- Execution of an exception. Exceptions are processed in the CPU mode they are called, except for interrupts in User Mode, which can be configured by the Security IC Embedded Software running in System Mode to either (a) not be processed at all or (b) be processed in System Mode but watch exceptions, which then are processed in User Mode. Exception during excpetion forces the TOE to reset.
- Execution of an interrupt. Interrupts are processed in the CPU mode they are called, except for interrupts in User Mode, which can be configured by the Security IC Embedded Software running in System Mode to be processed in System Mode.
- RETI instruction call switches to the suspended CPU mode. In User Mode RETI is restricted to RETI from interrupt, otherwise an exception is raised.

System Mode and User Mode are available to the Security IC Embedded Software. System Mode is more privileged since it allows access to all Special Function Registers related to hardware components, Special Function Registers to configure the MMU segmentation, Special Function Registers to configure the Firmware firewall and Special Function Registers related to system management, which all is denied in User Mode by default.

SF.SFR_ACC and SF.COMP together ensure that Super System Mode is not available to the Security IC Embedded Software, but reserved for the IC Dedicated Software.

SF.FFW: Firmware Firewall

The Firmware Operating System serves an FVEC interface, which is accessible to the Security IC Embedded Software. This interface provides erase and/or programming of Application-Flash and Application-EEPROM and also provides write and read access to the TOE Manufacturer EEPROM area.

SF.FFW controls access to the FVEC interface based on the CPU Mode. FVEC0 calls are granted in System Mode and in User Mode, whereas FVEC7 calls are granted in System Mode and are denied in User Mode. SF.FFW also implements control of write and read access to the TOE Manufacturer EEPROM area.

SF.FIRMWARE: Firmware Support

SF.FIRMWARE provides specific support functionality to the Security IC Embedded Software. This support functionality comprises integrity protection of code and data stored to Application-Flash and Application-EEPROM.

7.2 TOE Summary Specification Rationale

7.2.1 Mapping of Security Functional Requirements and TOE Security Functionality

Table 29 maps the portions of the TOE security functionality to the Security Functional Requirements. An "X" in the table means that the specific portion of the TOE security functionality realises the functionality required by the respective Security Functional Requirement. An “O” in the table means that the specific portion of the TOE security functionality supports the functionality required by the respective Security Functional Requirement.

Table 29. Mapping of TOE Security Functionality to SFRs

SFR	SS.RNG	SS.HW_DES	SS.HW_AES	SF.OPC	SF.PHY	SF.LOG	SF.COMP	SF.MEM_ACC	SF.SFR_ACC	SF.FFW	SF.FIRMWARE
FRU_FLT.2				X							
FPT_FLS.1		O	O	X			O	O	O		
FMT_LIM.1							X				
FMT_LIM.2							X	X	X		
FMT_LIM.1/Loader							X				
FMT_LIM.2/Loader							X	X	X		
FAU_SAS.1[HW]							X				
FDP_SDC.1[HW]						X					
FPT_PHP.3					X						
FDP_ITT.1[HW]	O	O	O			X					
FPT_ITT.1[HW]	O	O	O			X					
FDP_IFC.1	O	O	O			X					
FCS_RNG.1/PTG.2[HW]	X										
FCS_COP.1/TDES		X									
FCS_CKM.4/TDES		X									
FCS_COP.1/AES			X								
FCS_CKM.4/AES			X								
FDP_SDI.2[HW]											X
FDP_ACC.1[MEM]							X	X		X	
FDP_ACC.1[SFR]									X		
FDP_ACF.1[MEM]							X	X		X	

SFR	SS.RNG	SS.HW_DES	SS.HW_AES	SF.OPC	SF.PHY	SF.LOG	SF.COMP	SF.MEM_ACC	SF.SFR_ACC	SF.FFW	SF.FIRMWARE
FDP_ACF.1[SFR]									X		
FMT_MSA.3[MEM]								X		X	
FMT_MSA.3[SFR]									X		
FMT_MSA.1[MEM]								X		X	
FMT_MSA.1[SFR]									X		
FMT_SMF.1[HW]								X	X	X	

7.2.2 Rationale for the portions of the TOE security functionality

(deleted here, only available in the full version of the Security Target)

7.2.3 Security architectural information

Since this Security Target claims the assurance requirement ASE_TSS.2 security architectural information on a very high level is supposed to be included in the TSS to inform potential customers on how the TOE protects itself against interference, logical tampering and bypass. In the security architecture context, this covers the aspects self-protection and non-bypassability.

(details deleted here, available only in the full version of the Security Target)

8. Annexes

8.1 Further Information contained in the PP

Chapter 7 of the PP [6] provides further information. Section 7.1 in the PP [6] describes the development and production process of Security ICs including a detailed life cycle description and a description of the assets of the IC Designer/Manufacturer. Section 7.2 in the PP [6] comprises security aspects of the Security IC Embedded Software, i.e further information regarding A.Resp-Appl and examples of specific Functional Requirements for the Security IC Embedded Software. Section 7.3 in the PP [6] contains examples for Attack Scenarios.

8.2 Glossary and Vocabulary

Composite Product Manufacturer	see glossary and vocabulary in the PP [6]
CPU mode	Mode in which the CPU operates. The TOE supports five such CPU modes, which are Boot Mode, Test Mode, Firmware Mode, System Mode and User Mode.
End-consumer	see glossary and vocabulary in the PP [6]
exception interrupt	Non-maskable interrupt of code execution jumping to fixed addresses depending on the exception source and enabling System Mode. Sources of exceptions are hardware breakpoints, single fault injection detections, illegal instructions, stack overflows, unauthorised system call vector calls, execution of RETI instruction in User Mode, and the MMU exceptions access violation and access collision.
FabKey area	A memory area in the Application-EEPROM of the TOE, which contains data programmed by TOE Manufacturer during production test. The amount of data and the type of information can be selected by the customer.
IC Dedicated Software	see glossary and vocabulary in the PP [6]
IC Dedicated Test Software	see glossary and vocabulary in the PP [6]
IC Dedicated Support Software	see glossary and vocabulary in the PP [6]
Initialisation Data	belong to TSF data, see glossary and vocabulary in the PP [6]
Integrated Circuit (IC)	see glossary and vocabulary in the PP [6]
Kbyte(s)	1 Kbyte = 1024 bytes
memory	IC hardware ressource that stores code and/or data, like ROM, Flash, EEPROM or RAM.
Memory Management Unit	The MMU maps physical addresses of the memories to virtual addresses used by the CPU. This mapping is done based on (a) memory partitioning and (b) memory segments for code

	running in User Mode. Memory partitioning is fixed, whereas up to 64 memory segments can be configured individually. Each segment can be (i) positioned and sized (ii) enabled and disabled, (iii) configured for access rights in terms of read, write and execute in User Mode and (iv) configured for User Mode access rights to Special Function Registers related to hardware components of code executed in this segment. The MMU also controls access to memory partitions and memory segments.
memory segment	memory area specified in the MMU Segment Table, which under access control by the Memory Management Unit to support separation of different applications running in User Mode.
MMU Segment Table	This table specifies memory segments for code running in User Mode, which are controlled by the MMU. The table can be located anywhere in the memory that is accessible in System Mode. It also contains User Mode access rights to Special Function Registers related to hardware components of code executed in each segment.
Pre-personalisation Data	belong to User Data, see glossary and vocabulary in the PP [6]
Security IC	see glossary and vocabulary in the PP [6]
Security IC Embedded Software	see glossary and vocabulary in the PP [6]
Special Function Registers	Registers used to access, control and configure the hardware resources of the TOE.
Test Features	see glossary and vocabulary in the PP [6]
TOE Delivery	see glossary and vocabulary in the PP [6]
TOE Manufacturer	see glossary and vocabulary in the PP [6]
TSF data	see glossary and vocabulary in the PP [6]
User Data	see glossary and vocabulary in the PP [6]

8.3 List of Abbreviations

CC	Common Criteria Version 3.1
CPU	Central Processing Unit
DEA	Data Encryption Algorithm
DES	Data Encryption Standard
DRNG	Deterministic Random Number Generator
EAL	Evaluation Assurance Level
ECC	Elliptic Curve Cryptography
IC	Integrated Circuit
MMU	Memory Management Unit

PP	Protection Profile
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SIM	Subscriber Identity Module
SF	Security Feature
SPI	Serial Peripheral Interface
SS	Security Service
ST	Security Target
SWP	Single Wire Protocol
TOE	Target of Evaluation
TRNG	True Random Number Generator
TSF	TOE Security Functionality
TSFI	TSF Interface
TSP	TOE Security Policy
UART	Universal Asynchronous Receiver and Transmitter

9. Bibliography

9.1.1 Evaluation Documents

[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 for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002

[3] Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-003

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

[5] Supporting Document Mandatory Technical Document Application of Attack Potential to Smartcards, Version 2.7, Revision 1, March 2009, CCDB-2009-03-001

[6] Security IC Platform Protection Profile PP-0084 “Security IC Platform Protection Profile with Augmentation Packages”, Version 1.0, 2014-01-13, BSI-CC-PP-0084-2014, available at https://www.bsi.bund.de

[7] Anwendungshinweise und Interpretationen zum Schema, AIS31: Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren, Version 2.1, 02.12.2011, Bundesamt für Sicherheit in der Informationstechnik

[8] A proposal for: Functionality classes for random number generators, Version 2.0, 18 September 2011

9.1.2 Developer Documents

[9] SmartMX2 P61N1M3 Secure high-performance mobile controller, Product data sheet, NXP Semiconductors

[10] P61N1M3 VD, NV Properties, data sheet addendum, NXP Semiconductors

[11] P61N1M3 VE, NV Properties, data sheet addendum, NXP Semiconductors

[12] Instruction Set for the SmartMX2 family, Secure smart card controller, NXP Semiconductors, Business Unit Identification

[13] Chip Health Mode (CHM) for P61N1M3, data sheet addendum, NXP Semiconductors

[14] P61N1M3 Firmware interface specification, data sheet addendum, NXP Semiconductors

[15] NXP Secure Smart Card Controller P61N1M3PVD/VD-1/VE-1 Information on Guidance and Operation, NXP Semiconductors, Business Unit Identification

[16] SmartMX2 family P61N1M3 VD/VE Wafer and delivery specification, data sheet addendum, NXP Semiconductors

[17] Order Entry Form P61N1M3PVD/E, online document, NXP Semiconductors, Business Unit Identification

[18] Order Entry Form P61N1M3PVD-1/E-1, online document, NXP Semiconductors, Business Unit Identification

[19] Trust Provisioning – Trust Provisioning concept and security architecture, NXP Semiconductors

[20] Key Delivery Procedures for Trust Provisioning, NXP Semiconductors

[21] SmartMX2 family P61N1M3 Product errata sheet, NXP Semiconductors, Revision 1.2, Document Number 474012, 13 November 2018

9.1.3 Other Documents

[22] FIPS PUB 46-3 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION DATA ENCRYPTION STANDARD (DES) Reaffirmed 1999 October 25

[23] FIPS PUB 197 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION, ADVANCED ENCRYPTION STANDARD (AES), National Institute of Standards and Technology, 2001 November 26

[24] PKCS #1: RSA Cryptography Specifications, Version 2.0. RSA Laboratories, September 1998

[25] NIST SP 800-38A Recommendation for Block Cipher Modes of Operation, 2001, with Addendum Recommendation for Block Cipher Modes of Operation: Three Variants of Ciphertext Stealing for CBC Mode, October 2010

[26] NIST SP 800-67 Rev.2, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, revised November 2017, National Institute of Standards and Technology

10. Legal information

10.1 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

10.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of a NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on a weakness or default in the customer application/use or the application/use of customer's third party customer(s) (hereinafter both referred to as "Application"). It is customer's sole responsibility to check whether the NXP Semiconductors product is suitable and fit for the Application planned. Customer has to do all necessary testing for the Application in order to avoid a default of the Application and the product. NXP Semiconductors does not accept any liability in this respect.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from national authorities.

10.3 Licenses

ICs with DPA Countermeasures functionality

NXP ICs containing functionality implementing countermeasures to Differential Power Analysis and Simple Power Analysis are produced and sold under applicable license from Cryptography Research, Inc.

10.4 Patents

Notice is herewith given that the subject device uses one or more of the following patents and that each of these patents may have corresponding patents in other jurisdictions.

<Patent ID> — owned by <Company name>

10.5 Trademarks

Notice: All referenced brands, product names, service names and trademarks are property of their respective owners.

<Name> — is a trademark of NXP B.V.

11. List of figures

12. List of tables

13. Contents

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an please send an email to: salesaddresses@nxp.com

Date of release: 10 January 2019
Document identifier:

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an please send an email to:
salesaddresses@nxp.com

Date of release: 10 January 2019
Document identifier: