Security Target Lite

M9900, M9905, M9906

including optional Software Libraries RSA-EC-SCL-HCL-PSL

According to Common Criteria CCv3.1 EAL5 augmented (EAL5+)

Version: 4.9

Date: 2024-09-20

Table of Content

1 Security Target Introduction (ASE_INT)

1.1 Security Target and Target of Evaluation Reference

The title of this document is:
 “Confidential Security Target M9900, M9905, M9906 including optional Software Libraries RSA-EC-SCL-HCL-PSL v4.9, 2024-09-20”.

The name of the TOE on the CC certificate is:
“ Infineon Technologies Smart Card IC (Security Controller) M9900 A22, M9900 C22, M9900 D22, M9900 G11, M9905 A11, M9906 A11 with optional Software Libraries RSA2048, RSA4096, EC, Base, SCL, HCL, and PSL, and with specific IC dedicated software”

The Target of Evaluation (TOE) comprises the Infineon Technologies Smart Card IC (Security Controller) M9900, M9905, M9906 with optional RSA v2.05.005/v2.07.003, EC v2.05.005/v2.07.003, SCL v2.01.011/v2.02.010/v2.04.003, PSL v4.00.09/ v4.00.10/v5.00.06 and HCL v1.01.003 libraries with specific IC dedicated software. The design step is A22,G11, C22, D22 for the M9900 and A11 for the M9905 and M9906.

The Security Target is based on the Protection Profile “Smartcard IC Platform Protection Profile” [1].

The Protection Profile and the Security Target are built in compliance with Common Criteria v3.1 ([2],[3],[4]) .

The ST takes into account all relevant current final interpretations.

1 Table 1 Identification

Type	Version	Date	Title/Registration/Explainantion
Security Target Method of identification is done by version, date and title			
Security Target	4.9	2024-09-20	Security Target Lite M9900, M9905, M9906 including optional Software Libraries RSA-EC-SCL-HCL-PSL
TOE Hardware Method of identification is done by reading of GCIM			
M9900	A22, G11, C22, D22 See Remark 1		M9900 with Firmware Identifier 80001141 and Firmware Identifier 80001142 and external Flash-memory (optional)
M9905	A11		M9905 with Firmware Identifier 80001151 and external Flash-memory (optional)
M9906	A11		M9906 with Firmware Identifier 80001150 and external Flash-memory (optional)
Libraries (optional) Method of identification is done by hash values (shown in confidential ST only)			
NRG Management	01.03.0927		Management of NRG cards
NRG Reader	01.02.0800		NRG reader mode support
ACL	2.05.005		RSA2048
			RSA4096
			EC
			Toolbox
	2.07.003		RSA2048
			RSA4096
			EC
			Toolbox
SCL	2.01.011		Symmetric Crypto Library
	2.02.010		Symmetric Crypto Library
	2.04.003		Symmetric Crypto Library
PSL v4	4.00.09		Platform Support Layer
	4.00.10		Platform Support Layer
PSL v5	5.00.06		Platform Support Layer
HCL	1.01.003		Hash Crypto Library
FTL	1.01.0008		Flash Translation Layer

Type	Version	Date	Title/Registration/Explainantion
Hardware Guidance Documentation Method of identification is done by version, date and title			
General Guidance	Revision 3.0	2019-08-28	SLE97 M9900 Hardware Reference Manual
	Edition 2024-09-20	2020-04-15	M9900 Security Guidelines User´s Manual
	ID070218	2018-02-07	ARMv7-M Architecture Reference Manual, ARM DDI 0403E.d (ID070218), 2018, ARM Limited
	4.4.2	2020-03-11	SLE97 Programmer´s Reference Manual
	Edition 2014-08-10a	2014-08-10	SLE97 / SLC14 Family Production and Personalization User´s Manual
M9900 Guidance	4.1	2019-09-24	M9900 Errata Sheet
M9905 / M9906 Guidance	3.1	2019-09-05	M9905 M9906 Errata Sheet
Library Guidance Documentation (optional) Method of identification is done by version, date and title			
ACL Guidance	2.05.005	2024-08-26	CL97 Asymmetric Crypto Library for Crypto@2304T RSA / ECC / Toolbox, User Interface
	2.07.003	2024-08-26	CL97 Asymmetric Crypto Library for Crypto@2304T RSA / ECC / Toolbox, User Interface
SCL Guidance	2.01.011	2016-08-02	SCL97 Symmetric Crypto Library for SCPv3 DES/AES 32-bit Security Controller User Interface
	2.02.010	2016-12-09	SCL97 Symmetric Crypto Library for SCPv3 DES/AES 32-bit Security Controller User Interface
	2.04.003	2018-05-22	SCL97-SCP-v3-L90 Symmetric Crypto Library for SCP-v3 DES / AES 32-bit Security Controller
PSL v4 Guidance	Update 2020-04-14	2016-08-04	SLI 97 Family PSL Reference Manual User's Manual[1]
	1.6	2018-06-07	PSL Security Guidelines[1]
	1.1	2016-09-16	Release Notes PSL v4.00.09
	1.1	2018-06-07	Release Notes PSL v4.00.10

[1] This applies to version 4.00.09 and 4.00.10 of the PSL

Type	Version	Date	Title/Registration/Explainantion
PSL v5 Guidance	5.5	2020-04-09	SLx97 Platform Support Layer Library 32-bit Security Controller Programmer's Reference Manual
	2.5	2018-07-06	SLI97 Security Guidelines PSL V5.00.06
	1.0	2018-05-18	Release Notes PSL v5.00.06
HCL Guidance	1.01.003	2018-05-22	HCL97-CPU-L90 Hash Crypto Library for CPU SHA
FTL Guidance	1.0	2012-07-10	SLE 97 Flash Translation Layer User´s Guidance
CC documents Method of indentification is done by version, date and title			
PP	1.0	2007-06-15	Security IC Platform Protection Profile BSI-PP-0035 The cert-id BSI-CC-PP-0035-2007 refers to the corresponding certification report.
CC	3.1 Revision 5	2017-04	Security Evaluation Part 1: CCMB-2017-04-001 Part 2: CCMB-2017-04-002 Part 3: CCMB-2017-04-003

This TOE is represented by a number of various products. They all differentiate by different mask sets with slight - neither functional nor security relevant - modifications, various configuration possibilities, done either by Infineon settings during production or, after delivery, by means of blocking at customer premises. Despite these variation possibilities, all products are derived from the same hardware design results, the M9900 A22, the M9900 G11, the M9905 A11 and the M9906 A11.

The TOE can be identified with the Generic Chip Identification Mode (GCIM). The M-number hardware is identified by the bytes 05 and 06, which are the first two bytes of the chip identification number, having for the M9900 always the hexadecimal value of 0x0007, for the M9905 the value 0x0010 and for the M9906 the value 0x0011, the design step, firmware identifier, mask identifier, temperature range and system frequency are also included in the GCIM. Additionally the customer can read the configuration area as defined in the SLE97 Programmer´s Reference Manual [11].

Remark 1:
The derivatives of the TOE produced in the factory Dresden with the additional top layer on board (WLP, WLB) are managed with an own design step. These derivatives output a C22 in the GCIM for the WLP derivative and a D22 for the WLB derivative, which is always linked to the A22 design step. The C22 and D22 design step is only outputted at the derivatives with the additional top layer. All other identification options, i.e. the various metal option identifiers of the GCIM remain unchanged.

The derivatives of the TOE produced in the factory TSMC coming with the additional top layer on board (WLB) are managed with the same design step. These derivatives output a G11 in the GCIM for WLB derivative. All other identification options, i.e. the various metal option identifiers of the GCIM remain unchanged.

All products are identical with respect to module design and layout, but may include further package options require flexibility in design and could also depend on user requirements. In these cases one or more additional metal layer are added on top of one of the TOE mask set. These additional metal layers, it could also be more than one, just reroute the pads. Therefore, this last rerouting on top does not change the function of the TOE itself and is depending on the package only. These top metal layers are flexible in design, could depend also on user requirements and are of course not relevant for the security of the TOE. For these reasons, the metal layers are out the scope of the certification and do not belong to the TOE. Of course, in all cases passivation and isolation coating is applied on top of the last layers carrying wires. Further clear declaration and overview is given in chapter 2.1 Definition of the TOE.

Despite all these options and the resulting flexibility, all differences are comparable to the scenario where for example someone takes a piece of wire and reconnects the pads of the TOE using a soldering bolt. This does not change anything on the TOE security or security policy.

To each of the TOE relevant optional different mask set variants, an individual value is assigned, which is part of the data output of the Generic Chip Identification Mode (GCIM). By that the various hardware mask sets can be clearly identified and differentiated by the GCIM output. The interpretation of the output GCIM data is clearly explained in the user guidance, Hardware Reference Manual [7].

There are no other differences between the mask sets the TOE is produced with, and all these changes have no impact on the TOEs security policies and related functions. Details are explained in the user guidance Hardware Reference Manual [7] and in the Errata Sheet1 [12].

In addition to these hardware differences, the **M9900, M9905, M9906** allows a maximum of configuration possibilities defined by the customer order following the market needs. A detailed description of the TOE configuration possibilities is given in chapter 2.9 TOE Configuration.

1.2 Target of Evaluation overview

The TOE comprises the Infineon Technologies AG security controller M9900, M9905, M9906 with specific IC dedicated software and optional RSA, EC, SCL, PSL, HCL.

The Toolbox and Flash Translation Layer (FTL) libraries are additionally supported software which is out of scope of this certification.

The Toolbox library does not provide cryptographic support or additional security functionality as it provides only the following basic long integer arithmetic and modular functions in software, supported by the cryptographic coprocessor: Addition, subtraction, division, multiplication, comparison, reduction, modular addition, modular subtraction, modular multiplication, modular inversion and modular exponentiation. No security relevant policy, mechanism or function is supported. The toolbox library is deemed for software developers as support for simplified implementation of long integer and modular arithmetic operations.

The Flash Translation Layer (FTL) is the interface to the external Flash-memory and is provided optional to the customer as a binary link library.

The TOE is a member of the Infineon Technologies AG security controller family SLE97 meeting high requirements in terms of performance and security. The SLE97 family has been developed with a modular concept and different memory configurations, sets of peripherals and interfaces as well as different security features to satisfy market requirements. A summary product description is given in this Security Target (ST).

The TOE offer all functions that are both required and useful in security systems, and integrated peripherals that are typically needed in chipcard applications, such as information security, identification, access control, GSM and UMTS projects, electronic banking, digital signature and multi-application cards, ID cards, transportation and e-purse applications.

The TOE implements a dedicated security 32-bit RISC CPU designed on the basis of the ARMv7_M architecture designed in 90 nm CMOS technology. The integrated peripheral combine enhanced performance and optimized power consumption for a minimized die size to make the SLE97 controllers ideal for chipcard applications. The TOE offer a wide range of peripherals, including a UART (using the ISO interface), four timers, two watchdogs, a CRC module, a true RNG (TRNG), coprocessors for symmetric (e.g. DES, AES) and asymmetric (e.g. RSA, EC) cryptographic algorithms. Additionally a range of communication interfaces, such as GPIO, I2C, SWP, USB, SSC/SPI and a NRG interface are offered to provide maximum flexibility in terms of simultaneously communication ability.

The TOE provides a real 32-bit CPU-architecture and is compatible to the ARMv7-M instruction set architecture. The major components of the core system are the 32-bit CPU as a variant of the ARM Secure Core SC300, the Cache system, the Memory Protection Unit and the Memory Encryption/Decryption Unit. The TOE implements a full 32-bit addressing with up to 4 GByte linear addressable memory space, a simple scalable memory management concept and a scalable stack size. The flexible memory concept is built on the non volatile memory, respectively SOLID FLASH™ NVM[1]. For the SOLID FLASH™ NVM the Unified Channel Programming (UCP) memory technology is used. Additionally an optional external Flash-memory connected via the SPI interface is available.

The TOE provides the low-level firmware components Boot Software (BOS) and Resource Management System (RMS) and the high-level firmware Flash Loader (FL) and NRG software.

The NRG software includes the NRG operating system and additionally the optional library Management of NRG cards (version 01.03.0927) and the optional library NRG Reader Mode Support (01.02.0800). The Management of NRG cards provides an API for the management and

[1] SOLID FLASH™ is an Infineon Trade Mark and stands for the Infineon EEPROM working as Flash memory. The abbreviation NVM is short for Non Volatile Memory.

generation of NRG cards. The optional NRG reader mode support library (01.02.0800) enables an access to external NRG cards.

NRG software is not part of the TSF and do not implement any Security Functional Requirement.

The RMS firmware providing some functionality via an API to the Smartcard Embedded Software contains for example SOLID FLASH™ NVM service routines and functionality for the tearing save write into the SOLID FLASH™ NVM. The BOS firmware (BOS-V1 and BOS-V2) is used for test purposes during start-up and the FL allows downloading of user software to the NVM during the manufacturing process. The BOS is implemented in a separated Test-ROM being part of the TOE. For the M9900 two different versions of the BOS are provided (BOS-V1 and BOS-V2). The version BOS-V1 (Firmware Identifier 80001141, 80001150, 80001151) executes the UMSLC test during the startup phase; the version BOS-V2 (Firmware Identifier 80001142) does not execute the UMSLC test during the startup phase to short the time duration of the startup phase. The derivate M9906 with Firmware Identifier 80001150 includes the feature "hardening" and the derivate M9905 with Firmware Identifier 80001151 includes the features "hardening" and the "Burn-In Test". The feature "hardening" analyzing a random SOLID FLASH™ NVM page after every regular program operation for written bits that are losing their charge, and, in this very unlikely case, the page is rewritten. The "Burn-In Test" during production is used to stress the chip in a high temperature, high internal voltage and active operation for a certain time and filtering out defect parts to get a low failure rate. The derivatives M9905 and M9906 are qualified for an extended temperature range from -40°C to +105°C.

The two cryptographic co-processors serve the need of modern cryptography: The symmetric co-processor (SCP) combines both AES and Triple-DES with dual-key or triple-key hardware acceleration. The Asymmetric Crypto Co-processor, called Crypto2304T in the following, supports RSA-2048 bit (4096-bit with CRT) and Elliptic Curve (EC) cryptography with high performance.

A True Random Number Generator (TRNG) specially designed for smart card applications is implemented. The TRNG fulfils the requirements from the functionality class PTG.2 of the AIS31 and produces genuine random numbers which then can be used internally or by the user software.

The software part of the TOE consists of the cryptographic libraries RSA and EC and asymmetric Base libraries, the optional Symmetric Crypto Libraries (SCL) and Platform Support Layer (PSL) libraries. If a RSA or EC library is part of the shipment, the corresponding asymmetric Base library is automatically included. If the PSL library v4.00.09 or v4.00.10 is part of the shipment, the RSA, EC, Base libraries v2.05.005 and the SCL library v2.01.011 are automatically included. If the PSL library v5.00.06 is part of the shipment, the RSA, EC, Base libraries v2.07.003, the SCL library v2.04.003 and the HCL library v1.01.003 are automatically included.

The RSA library is used to provide a high-level interface to RSA (Rivest, Shamir, Adleman) cryptography implemented on the hardware component Crypto2304T and includes countermeasures against SPA, DPA and DFA attacks. The routines are used for RSA signature verification, RSA signature generation and RSA modulus recalculation.The hardware Crypto2304T unit provides the basic long number calculations (add, subtract, multiply, square with 1100 bit numbers) with high performance. The RSA library is delivered as object code. The RSA library can perform RSA operations from 512 to 4096 bits.

The EC library is used to provide a high-level interface to Elliptic Curve cryptography implemented on the hardware component Crypto2304T and includes countermeasures against SPA, DPA and DFA attacks. The routines are used for ECDSA signature generation, ECDSA signature cerification, ECDSA key generation and Elliptic Curve Diffie-Hellman key agreement. The EC library is delivered as object code. The certification covers the standard NIST [DSS] and Brainpool [ECC] Elliptic Curves with key lengths of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 Bits. Note that there are numerous other curve types, being also secure in terms of side channel attacks on this TOE, which can the user optionally add in the composition certification process.

The asymmetric Base library provides the low level interface to the asymmetric cryptographic coprocessor and has no user available interface. The asymmetric Base library does not provide any security functionality, implements no security mechanisms and does not contribute to a security functional requirement.

The Flash Translation Layer Library provides the interface to the external Flash-memory. The Flash Translation Layer Library does not provide any security functionality, implements no security mechanism, and does not contribute to a security functional requirement.

The Symmetric Crypto library (SCL) is used to provide a high level interface to DES/3DES and AES symmetric cryptographic operations. It uses the SCP of the underlying hardware but implements also countermeasures against all known weaknesses of the SCP (e.g. dummy calculations and block repetitions). The symmetric crypto library consists of three C-library files Cipher.lib, AES.lib and DES.lib. Those library files will not be distributed individually. Therefore we call those three library files simply the Symmetric Crypto Library (SCL)

The Hash Cryptographic Library (HCL) provides the hash functions form the SHA-1 and SHA-2 family. The hash functions are hardened against SPA template attacks.

The Platform Support Layer (PSL) library is used to provide a standardized interface to the hardware by making use of the RSA, ECC, SCL and HCL libraries. The provided interfaces are syntactically similar to Windows NT device driver calls.

To fulfill the high security standards for smartcards today and also in the future, this TOE utilizes an integral security concept comprising countermeasure mechanisms specially designed against possible attack scenarios. The TOE provide a robust set of sensors for the purpose of monitoring proper chip operating conditions and detecting fault attack scenarios. The sensors are complemented with digital error detection mechanisms such as parities, error detection codes and instruction stream signatures. Probing and forcing attacks will be counteracted by the security optimized wiring approach, implemented by an Infineon-specific shielding combined with secure wiring of security critical signals, partly masking of security critical signals and by encryption of all memories inside the chip (RAM, ROM, NVM). A decentralized alarm propagation and system deactivation principle is implemented, further decreasing the risk of manipulating and tampering. Additionally, an online check of the security mechanisms is available by using the User Mode Security Life Control (UMSLC). Side-channel attacks (e.g. Timing Attack, SPA, DPA, EMA) are typically defeated using a combination of hardware and software mechanisms, for this the TOE provides several supporting features e.g. trash register writes and instruction interrupt prevention. The Instruction Stream Signature Checking (ISS) is a powerful countermeasure against fault attacks that try to manipulate the execution sequence of the instruction stream. All executed instructions are hashed in the CPUs signature register and the hardware automatically checks the fitting of the values.

In this security target the TOE is described and a summary specification is given. The security environment of the TOE during its different phases of the lifecycle is defined. The assets are identified which have to be protected through the security policy. The threats against these assets are described. The security objectives and the security policy are defined, as well as the security requirements. These security requirements are built up of the security functional requirements as part of the security policy and the security assurance requirements. These are the steps during the evaluation and certification showing that the TOE meets the targeted requirements. In addition, the functionality of the TOE matching the requirements is described.

The assets, threats, security objectives and the security functional requirements are defined in this Security Target and in [1] and are referenced here. These requirements build up a minimal standard common for all Smartcards.

The security functions are defined here in the security target as property of this specific TOE. Here it is shown how this specific TOE fulfils the requirements for the standard defined in the Protection Profile [1].

The user software can be implemented in various options depending on the user's choice and described in chapter 2.9. Thereby the user software can be implemented the NVM or coming without user software. In the latter case, the user downloads his entire software on his own using the Flash Loader software.

The TOE uses also Special Function Registers SFR. These SFR registers are used for general purposes and chip configuration. These registers are located in the SOLID FLASH™ NVM as configuration area page.

A shielding algorithm finishes the upper layers above security critical signals and wires, finally providing the so called "security optimized wiring".

The TOE with its integrated security features meets the requirements of all smart card applications such as information integrity, access control, mobile telephone and identification, as well as uses in electronic funds transfer and healthcare systems.

To sum up, the TOE is a powerful smart card IC with a large amount of memory and special peripheral devices with improved performance, optimized power consumption, at minimal chip size while implementing high security.

2 Target of Evaluation Introduction

The TOE description helps to understand the specific security environment and the security policy. In this context the assets, threats, security objectives and security functional requirements can be employed. The following is a more detailed description of the TOE than in [1] as it belongs to the specific TOE.

2.1 Definition of the TOE

The TOE comprises three parts:

- Hardware of the smart card security controller including all configurations and derivatives
- Associated firmware, software and optional software
- Documents.

The hardware configuration options and configuration methods are described in the chapters 1.1 and 2.9. The second part of this TOE includes the associated firmware and software required for operation. The TOE can be delivered in various configurations, achieved by means of blocking and depending on the customer order.

The documents as described in section 2.6 and listed in Table 1, are supplied as user guidance. All product derivatives of this TOE, including all configuration possibilities differentiated by the GCIM data and the configuration information output, are manufactured by Infineon Technologies AG. In the following descriptions, the term “manufacturer” stands short for Infineon Technologies AG, the manufacturer of the TOE. The Smartcard Embedded Software respectively user software is not part of the TOE. New configurations can occur at any time depending on the user blocking or by different configurations applied by the manufacturer. In any case the user is able to clearly identify the TOE hardware, its configuration and proof the validity of the certificate independently, meaning without involving the manufacturer. The various blocking options, as well as the means used for the blocking, are done during the manufacturing process or at user premises. Entirely all means of blocking and the the blocking involved firmware respectively software parts, used at Infineon Technologies AG and/or the user premises, are subject of the evaluation. All resulting configurations of a TOE derivative are subject of the certificate. All resulting configurations are either at the predefined limits or within the predefined configuration ranges.
One or more additional metal layer may be added on top of one of the TOE mask set. These additional metal layers, it could also be more than one, just reroute the pads. Therefore, this last rerouting on top does not change the function of the TOE itself and is depending on the package only, and are not relevant for the security of the TOE. For these reasons, the metal layers are out the scope of the certification and do not belong to the TOE. Of course, in all cases passivation and isolation coating is applied on top of the last layers carrying wires.

A shielding algorithm finishes the upper layers above security critical signals and wires, finally providing the so called “security optimized wiring”.

The firmware used for the TOE internal testing and TOE operation, the firmware and software parts exclusively used for the blocking, the parts of the firmware and software required for cryptographic support are part of the TOE and therefore part of the certification. The documents as described in chapter 2.6 are supplied as user guidance.

The term Smartcard Embedded Software is used in the following for all operating systems and applications stored and executed on the TOE. The TOE is the platform for the Smartcard Embedded Software. The Smartcard Embedded Software itself is not part of the TOE. The TOE does not require any non-TOE hardware/software/firmware.

2.1.1 Major security functions of the TOE

The major security functions of the TOE are:

- Memory Protection Unit
- Memory Encryption/Decryption Unit
- Sensors for the purpose of monitoring proper chip operating conditions and detecting fault attack scenarios complemented with digital error detection mechanisms such as parities, error detection codes and instruction stream signatures
- Security optimized wiring for protection of security critical signals
- Instruction Stream Signature Checking (ISS) as a countermeasure against fault attacks that try to manipulate the execution sequence of the instruction stream
- Symmetric cryptographic coprocessor supporting AES and 3DES (optional)
- Crypto2304T, an asymmetric crypto coprocessor supporting RSA and EC (optional)
- Cryptographic libraries for RSA and EC computations (optional)
- Cryptographic libraries for DES and AES computations (optional)
- Hash library for SHA-1 and SHA-2 hash functions (optional)
- A true random number generator, which can be used as a security service to the user and for internal purpose
-

2.1.2 Not part of the TOE and not part of the certification

Not part of the TOE and not part of the certification are

- the Smartcard Embedded Software respectively user software, and
- the piece of software running at user premises and collecting the BPU receipts coming from the TOE. This BPU software part is the commercially deemed part of the BPU software, not running on the TOE, but allowing refunding the customer, based on the collected user blocking information. The receipt from each blocked TOE is collected by this software – chip by chip.
- The NRG software

2.2 Hardware of the TOE

The hardware part of the TOE (see Figure 2) as defined in [1] is comprised of:

Core System

- 32-bit CPU implementation of ARM Secure Core SC300 based on ARMv7-M Instruction set architecture including the Instruction Stream Signature Checking (ISS)
- CACHE for code and data buffering
- Memory Encryption/Decryption Unit (MED) and Error Detection Unit
- Memory Protection Unit (MPU)
- Nested Vectored Interrupt Controller (NVIC)

Interfaces

- Universal Asynchronous Receiver/Transmitter (UART)
- Single-Wire Protocol (SWP) with NRG interface
- Inter Integrated Circuit (I2C) interface
- General Purpose Input Output (GPIO)

- Synchronous Serial Communication (SSC) which provides the Serial Peripheral Interface (SPI)
- Universal Serial Bus (USB) interface
- Standard ISO Interface (PAD)

Memories

- Read-Only Memory (ROM, for internal firmware)
- Random Access Memory (RAM)
- SOLID FLASH™ NVM memory (NVM)

Note that the TOE has implemented a SOLID FLASH™ NVM memory module. Parts of this memory module are configured to work as an EEPROM.

Peripherals

- True Random Number Generator (TRNG)
- System Module (SYS)
- Clock Unit (CLK)

Coprocessors

- Crypto2304T co-processor for asymmetric algorithms like RSA and EC (Crypto, optional)
- Symmetric Crypto co-processor for 3DES and AES Standards (SCP, optional)
-

Analog Module (ANA)

- Glitch Sensor
- Temperature Sensor
- Backside Light Detector
- User Mode Security Life Control (UMSLC)

Buses

- Memory Bus
- Peripheral Bus

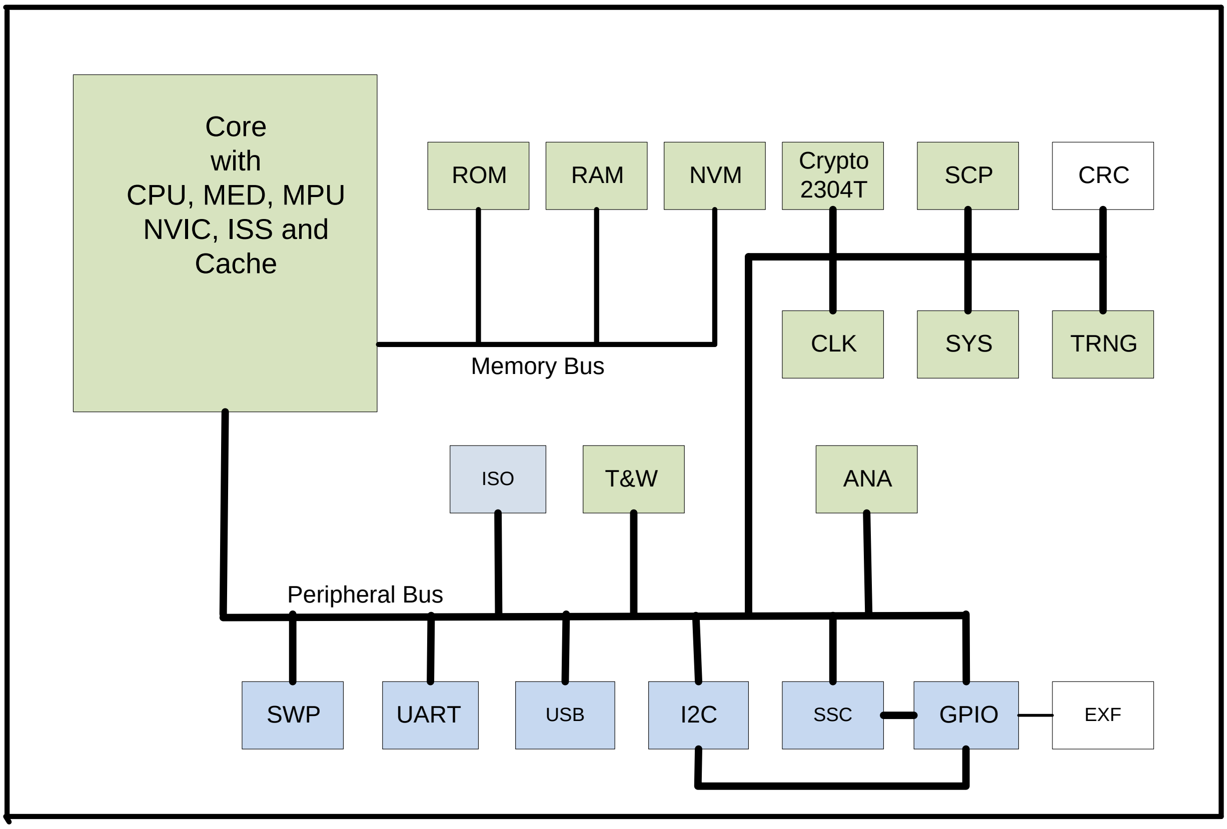


Figure 1

Core	Core System	ROM	Read Only Memory
NVM	SOLID FLASH™ NVM	RAM	Random Access Memory
CLK	Clock Unit	SYS	System Module
Crypto	Crypto2304T	SCP	Symmetric Crypto Processor
CRC	Cyclic Redundancy Check	TRNG	True Random Number Generator
T&W	Timer and Watchdog	UART	UART
I2C	Inter Integrated Circuit	GPIO	General Purpose IO
SSC	Synchronous Serial Communication	SWP	Single Wire Protocol
USB	Universal Serial Bus	ANA	Analog Units
ISO	Standard Interface	ISO	Standard ISO Interface
EXF	External Flash-memory (optional)		

Figure 2 Block diagram of the M990X products (TOE parts are filled with light green, interface parts are filled in light blue)

The TOE consists of smart card ICs (Security Controllers) meeting high requirements in terms of performance and security. They are manufactured by Infineon Technologies AG in a 90 nm CMOS-technology (L90). This TOE is intended to be used in smart cards for particularly security-relevant applications and for its previous use as developing platform for smart card operating systems according to the lifecycle model from [1]

The term Smartcard Embedded Software is used in the following for all operating systems and applications stored and executed on the TOE. The TOE is the platform for the Smartcard Embedded Software. The Smartcard Embedded Software itself is not part of the TOE.

The TOE consists of a core system, memories, co-processors, security peripherals, control logic and peripherals. The major components of the core system are the 32-bit CPU (Central Processing Unit), the MPU (Memory Protection Unit), the MED (Memory Encryption/Decryption Unit), the Nested Vectored Interrupt Controller (NVIC), the Instruction Stream Signature Checking (ISS) and the

Cache system. The TOE contains the co-processors for RSA/EC (Crypto2304T) and DES/AES (SCP) processing, a CRC module and the peripherals random number generator, four timers and two watchdog timers and several external interface services. All data of the memory block is encrypted, RAM and ROM are equipped with an error detection code (EDC) and the SOLID FLASH™ NVM is equipped in addition with an error correction code (ECC).
The memories are connected to the Core with the Memory Bus and the peripherals are connected with the Peripheral Bus.

The Analog Modules (ANA) serve for operation within the specified range and manage the alarms. A set of sensors (temperature sensor, backside light detector, glitch sensor) is used to detect excessive deviations from the specified operational range and serve for robustness of the TOE and the UMSLC function can be used to test the alarm lines.

The CPU is compatible with the instruction set of the ARMv7_M architecture. Despite its compatibility the CPU implementation is entirely proprietary and not standard.

The CPU accesses the memory via the integrated Memory Encryption and Decryption unit (MED). The memory model of the TOE provides two distinct, independent levels. Additionally up to eight regions can be defined with different access rights controlled by the Memory Protection Unit (MPU). Errors in RAM and ROM are automatically detected (EDC, Error Detection Code), in terms of the SOLID FLASH™ NVM errors are detected and 1-Bit-errors are also corrected (ECC, Error Correction Code).

The controller of this TOE stores both code and data in a linear 4-GByte memory space, allowing direct access without the need to swap memory segments in and out of memory using a memory protection unit.
Additionally an optional external Flash-memory (EXF) connected via the SSC/GPIO interfaces is available. The data stored in the external Flash-memory are not protected as the external Flash-memory is not part of the security functional requirements (SFR) of the TOE and not in the scope of the evaluation.

The CACHE is a high-speed memory-buffer located between the CPU and the (external) main memories holding a copy of some of the memory contents to enable access, which is considerably faster than retrieving the information from the main memory. In addition to its fast access speed, the CACHE also consumes less power than the main memories. The CACHE is equipped with a integrity check to verify the contents of the cache memories.

A True Random Number Generator (TRNG) specially designed for smart card applications is implemented. The TRNG fulfils the requirements from the functionality class PTG.2 of the AIS31 and produces genuine random numbers which then can be used internally or by the user software.

The implemented sleep mode logic (clock stop mode per ISO/IEC 7816-3) is used to reduce the overall power consumption. The timers permits easy implementation of communication protocols such as T=1 and all other time-critical operations. The UART-controlled I/O interface allows the smart card controller and the terminal interface to be operated independently.

The Clock Unit (CLKU) supplies the clocks for all components of the TOE. It generates the system clock and an approximately 1MHz clock for the timers. The 1MHz clock is derived from an internal oscillator, while the system clock may either be based on the internal oscillator clock (internal clock mode) or on an external clock (external clock mode). Additionally a sleep mode is available. When operating in the internal clock mode the system frequency can be configured by the user software combined with the current limitation functionality. In the external clock mode the clock is derived from the external clock and a parameter with the range of 1 to 8. The system frequency may be 1 up to 8 times the externally applied frequency but is of course limited to the maximum system frequency and can be combined with the current limitation function.

Two co-processors for cryptographic operations are implemented on the TOE. The Crypto2304T for calculation of asymmetric algorithms like RSA and Elliptic Curve (EC) and the Symmetric Cryptographic Processor (SCP) for dual-key or triple-key triple-DES and AES calculations. These co-processors are especially designed for smart card applications with respect to the security and power consumption. The SCP module computes the complete DES algorithm within a few clock cycles and is especially designed to counter attacks like DPA, EMA and DFA. The Crypto2304T module provides basic functions for the implementation of RSA and EC cryptographic libraries.

Note that this TOE can be delivered with both crypto co-processors accessible, or with a blocked SCP or with a blocked Crypto2304T, or with both crypto co-processors blocked. The blocking depends on the customer demands prior to the production of the hardware. No accessibility of the deselected cryptographic co-processors is without impact on any other security policy of the TOE; it is exactly equivalent to the situation where the user decides just not to use the cryptographic co-processors.

The cyclic redundancy check (CRC) module is a 16-bit checksum generator, which shall not be used for security-critical data. The TOE includes two timer modules each with two 16-bit general purpose timers. The timer module can be used also as watchdog timer to monitor system operation for possible timeouts and to check the correct order of operation.

An Interface Management module, located in the System Module (SYS), provides the TOE with the possibility to maintain two or more data interfaces simultaneously. The TOE is provided with, dependent on the configuration, different peripherals and interfaces as the Universal Serial Bus (USB), the SWP Slave Peripheral (SWP), the Synchronous Serial Communication (SSC), which provides the serial Peripheral Interface (SPI), the GPIO module (GPIO), the Inter-Integrated Cirquit Module (I2C) and the Standard ISO Interface (PAD) to satisfy the different market requirements.

2.2.1 Non-TOE parts of the hardware

The following parts of the hardware are not part of the certification scope.

- Checksum module (CRC)
- External Flash-memory (EXF, optional)
-

2.3 Firmware of the TOE

The entire firmware and software of the TOE consists of different parts:

The BOS (Boot Software) and the RMS (Resource Management System) compose the TOE firmware stored in the ROM and the patches hereof in the SOLID FLASH™ NVM. All mandatory functions for start-up and internal testing (BOS) are protected by a dedicated hardware firewall. Additionally two levels are provided, the privileged level and the non-privilege level, both are protected by a hardwired Memory Protection Unit (MPU) setting. For the TOE two different versions of the BOS are provided (BOS-V1 and BOS-V2). The version BOS-V1 (Firmware Identifier 80001141, 80001150, 80001151) executes the UMSLC test during the startup phase, the version BOS-V2 (Firmware Identifier 80001142) does not execute the UMSLC test during the startup phase to shorten the time duration of the startup phase. For the M9906 the BOS-V1 version (Firmware Identifier 80001150) includes the feature “hardening” and for the M9905 the BOS-V1 version (Firmware Identifier 80001151) includes the features “hardening” and the “Burn-In Test”. The feature “hardening” analyzing a random SOLID FLASH™ NVM page after every regular program operation for written bits that are losing their charge, and, in this very unlikely case, the page is rewritten. The “Burn-In Test” during production is used to stress the chip in a high temperature, high internal voltage and active operation for a certain time and filtering out defect parts to get a low failure rate. The derivatives M9905 and M9906 are qualified for an extended temperature

range from -40°C to +105°C.
The RMS is accessible in privileged level only. The FL (Flash Loader) allows downloading of user software to the NVM during the manufacturing process and can be completely deactivated.

2.4 Optional software of the TOE

The optional software part of the TOE consists of the cryptographic libraries RSA and EC and asymmetric Base libreries, the optional SCL, the optional Platform Support Library (PSL).
The RSA library is used to provide a high-level interface to the RSA cryptography implemented on the hardware component Crypto2304T and includes countermeasures against SPA, DPA and DFA attacks. The routines are used for the RSA signature verification, the RSA signature generation and the RSA modulus recalculation. The module provides the basic long number calculations (add, subtract, multiply, square with 1100-bit numbers) with high performance.
The RSA library is delivered as object code and is integrated in this way into the user software. The RSA library can perform RSA operations from 512 to 4096 bits. Depending on the customer's choice, the TOE can be delivered with the 4096 code portion or with the 2048 code portion only. The 2048 code portion is included in both.
Part of the evaluation are the RSA straight operations with key lengths from 1024 bits to 2048 bits, and the RSA CRT operations with key lengths of 1024 bits to 4096 bits. Note that key lengths below 1024 bits are not included in the certificate.
The EC library is used to provide a high level interface to Elliptic Curve cryptography and includes countermeasures against SPA, DPA and DFA attacks. The routines are used for ECDSA signature generation, ECDSA signature verification, ECDSA key generation and Elliptic Curve Diffie-Hellman key agreement. The EC library is delivered as object code and integrated in this way into the user software. The certification covers the standard NIST [DSS] and Brainpool [ECC] Elliptic Curves with key lengths of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 Bits. Note that there are numerous other curve types, being also secure in terms of side channel attacks on this TOE, which can the user optionally add in the composition certification process.
The Asymmetric Base library provides the low level interface to the asymmetric cryptographic coprocessor for the RSA and ECC cryptographic libraries and has no user available interface. It does not support any security relevant policy or function. The Base, ECC and RSA library can optionally be delivered in different versions:

- The version v2.05.005 , which is a dependency for the PSL library v 4.00.09 and v4.00.10,
- The version v2.07.003 , which is a dependency for the PSL library v5.00.06.
-

The Symmetric Crypto library (SCL) is used to provide a high level interface to DES/3DES and AES symmetric cryptographic operations. It uses the SCP of the underlying hardware but implements also countermeasures against all known weaknesses of the SCP (e.g. dummy calculations). The symmetric crypto library consists of three C-library files Cipher.lib, AES.lib and DES.lib. Those library files will not be distributed individually. Therefore we call those three library files simply the Symmetric Crypto Library (SCL). The SCL library can optionally delivered in different versions

- The legacy version v2.01.011 for backward compatibility, which is a dependency for the PSL v4.00.09 and v4.00.10,
- The legacy version v2.02.010 for backward compatibility,
- The most recent version v2.04.003, which is a dependency for the PSL v5.00.06.

The Hash Cryptographic Library (HCL) provides interfaces to the SHA-1 and SHA-2 family. The HCL library can optionally delivered in version

- v1.01.003, which is a dependency for the PSL v5.00.06.

The Platform Support Layer (PSL) library is used to provide a standardized interface to the hardware, directly or via the RSA, ECC and SCL library. The provided interfaces are syntactically similar to Windows NT device driver calls. The PSL library can optionally be delivered in different versions

- v4.00.09,
- v4.00.10,
- v5.00.06.

Table 2 Chip and optional software delivery matrix

Chip	Waferfab	Toplayer	Firmware-ID	RSA/ECC lib	SCL	PSL	HCL
M9900 A22	Dresden	none	80001141 (BOS-V1) 80001142 (BOS-V2)	2.05.005 2.07.003	2.01.011 2.02.010 2.04.003	4.00.09 4.00.10 5.00.06	1.01.003
M9900 C22	Dresden	WLP	80001141 (BOS-V1) 80001142 (BOS-V2)	2.05.005 2.07.003	2.01.011 2.02.010 2.04.003	4.00.09 4.00.10 5.00.06	1.01.003
M9000 D22	Dresden	WLB	80001141 (BOS-V1) 80001142 (BOS-V2)	2.05.005 2.07.003	2.01.011 2.02.010 2.04.003	4.00.09 4.00.10 5.00.06	1.01.003
M9900 G11	TSMC	WLB	80001141 (BOS-V1) 80001142 (BOS-V2)	2.07.003	n.A.	n.A.	n.A.
M9905 A11	Dresden	none	80001151 (BOS-V1)	2.05.005 2.07.003	2.01.011 2.02.010 2.04.003	4.00.09 4.00.10 5.00.06	1.01.003
M9906 A11	Dresden	none	80001150 (BOS-V1)	2.05.005 2.07.003	2.01.011 2.02.010 2.04.003	4.00.09 4.00.10 5.00.06	1.01.003

2.5 Interfaces of the TOE

- The physical interface of the TOE to the external environment is the entire surface of the IC.
- The electrical interface of the TOE to the external environment is constituted by the pads of the chip:
  - The five ISO 7816 pads consist particularly of the contacted RES, I/O, CLK lines and supply lines VCC and GND. The contact based communication is according to ISO 7816/ETSI/EMV. The I2C communication can be driven via the ISO 7816 pads. In this case no other communication using the ISO 7816 pads is possible.
  - The GPIO interface consists of 4 pads which can be individually configured and combined.
  - Also the I2C and the SSC/SPI communication can be exclusively driven via the GPIO pads. In this case no other communication using the GPIO pads is possible.
  - The USB interface is build out of two dedicated pads for data communication and two pads used from the ISO 7816 interface supplying power and ground.
  - The SWP interface is build out of one pad to support the SWP slave functionality.
- The data-oriented I/O interface to the TOE is formed by the I/O pad.
- The interface to the firmware is constituted by special registers used for hardware configuration and control (Special Function Registers, SFR).

- The interface of the TOE to the operating system is constituted on one hand by the RMS routine calls and on the other by the instruction set of the TOE.
- The interface of the TOE to the test routines is formed by the BOS test routine call, i.e. entry to test mode (OS-TM entry).
- The interface to the RSA calculations is defined by the RSA library (optionally).
- The interface to the EC calculations is defined by the EC library (optionally).
- The interface to the symmetric crypto operations DES/3DES/AES is defined by the SCL library (optionally).
- The interface to the PSL library is defined by the PSL Specification (optionally).

2.6 Guidance documentation

The guidance documentation is listed in Table 1

Finally the certification report may contain an overview of the recommendations to the software developer regarding the secure use of the TOE. These recommendations are also included in the ordinary documentation.

2.7 Forms of delivery

The TOE can be delivered in form of bare dies, in form of plain wafers, in form of complete modules (wire bond module M4.x, provided as single chip wire bond or as stacked wire bond), or in one of the following an IC cases: MFC5.8 (FCOS), PG-VQFN-8-1, PG-VQFN-32-13 (SMD) and P-M2M4.7-8-1 (for M9905 and M9906). The form of delivery does not affect the TOE security and it can be delivered in any form, as long as the processes applied and sites involved have been subject of the appropriate audit.

The delivery can therefore be at the end of phase 3 or at the end of phase 4 which can also include pre-personalization steps according to PP [1]. Nevertheless in both cases the TOE is finished and the extended test features are removed. In this document are always both cases mentioned to avoid incorrectness but from the security policy point of view the two cases are identical.

The delivery to the software developer (phase 2 ➔ phase 1) contains the development package and is delivered in form of documentation as described above, data carriers containing the tools and emulators as development and debugging tool.
Part of the software delivery could also be the Flash Loader program, provided by Infineon Technologies, running on the TOE and receiving via the UART interface the transmitted information of the user software to be loaded into the SOLID FLASH™ NVM memory. The download is only possible after successful authentication. The user software can also be downloaded in an encrypted way. In addition, the user can permanently block further use of the Flash Loader. Whether the Flash Loader program is present or not depends on the procurement order.

Table 3 TOE deliveries: forms and methods

TOE Component	Delivered Format	Delivery Method	Comment
M9900 C11/D11/G11/A22	See text above	Postal transfer in cages	All materials are delivered to distribution centers in cages, locked.
M9905 A11/M9906 A11	See text above	Postal transfer in cages	All materials are delivered to distribution centers in cages, locked.
All Firmware	–	–	Stored on the delivered hardware.

TOE Component	Delivered Format	Delivery Method	Comment
All software libraries	ARM Library File (object code)	Secured download[1]	–
All User Guidance documents	Personalized PDF	Secured download	–

2.8 Production sites

The silicon of the design A11, A22, C22 and D22 is produced in Dresden.
The silicon of the design G11 is produced at TSMC/Taiwan
The delivery measures are described in the ALC_DVS aspect.

Table 4 Production site in chip identification

Production Site	Chip Identification
Dresden, Germany	byte number 13 (Fab number): 02_H
TSMC, Taiwan	byte number 13 (Fab number): $0A_H$

2.9 TOE Configuration

This TOE is represented by various configurations called products, which are all derived from the equal hardware design M9900, M9905 and M9906. The same mask is used to produce different products of the TOE. The first metal mask (called the M1 mask) contains the specific information to identify the TOE.
The M9900, M9905 and M9906 product offers different configuration options, which a customer can choose. The mechanism to choose a configuration can be done by the following methods:

1. by product selection or dialog-based in Tools,
2. via Bill-per-Use (BpU) and Flash Loader (FL),

The degree of freedom for configuring the TOE is predefined by Infineon Technologies AG. The list of predefined TOE configurations is given, as an example in Table 5 and in the SLE97 Hardware Reference Manual [7], section 18. Additional the Table 5 gives an overview about the maximum configurable memory and frequency sizes of the TOE.
All these possible TOE configurations equal and/or within the specified ranges are covered by the certificate.

For details about the TOE configurations, please see [ST]

Beside fix TOE configurations, which can be ordered as usual, this TOE implements optionally the so called Bill-Per-Use (BPU) ability. This solution enables the customer to tailor the product on his own to the required configuration by blocking parts of the chip on demand into the final configuration at his own premises, without further delivery or involving support by Infineon Technology AG. Customers, who are intended to use this feature receiving the TOE in a predefined configuration including the Flash Loader software, enhanced with the BPU blocking software. The

[1] Secured download is a way of delivery of documentation and TOE related software using a secure ishare connected to Infineon customer portal. The TOE user needs a DMZ Account to login (authenticate) via the Internet.

blocking information is part of a chip configuration area and can be modified by customers using specific APDUs. Once a final blocking is done, further modifications are disabled.
The BPU software part is only present on the products which have been ordered with the BPU option. In all other cases this software is not present on the product.
Additionally the user can choose between different firmware BOS versions and optional software libraries.
For the M9900 derivative the user can choose the TOE with the BOS firmware in the version BOS-V1 or BOS-V2.
The user can choose between one of the management of NRG libraries (version 01.03.0927) and the NRG reader mode support library (01.02.0800) or the user can choose only one of the three libraries. Please note that the NRG libreries are not part of this certification.
In the case the TOE is equipped with the External Flash memory the user can choose the Flash Translation Layer (V1.01.0008) library.
The hardware of this TOE can be delivered with the following configuration options:

- both crypto co-processors accessible
- with a blocked SCP
- with a blocked Crypto2304T
- both crypto co-processors blocked

In case the SCP is blocked, no AES and 3DES computation supported by hardware is possible. In the case the Crypto2304T is blocked, no RSA and EC computation supported by hardware is possible.
No accessibility of the deselected cryptographic co-processors is without impact on any other security policy of the TOE; it is exactly equivalent to the situation where the user decides just not to use the cryptographic co-processors.
The TOE can be delivered with the following optional libraries

- RSA
- ECC
- Asymmetric Base library for RSA and ECC
- SCL for AES/DES
- PSL

The libraries of this TOE can be delivered according to the following dependencies:

- If the PSL library v4.00.09 or 4.00.10 is delivered, the RSA, EC and Base v2.05.005 libraries as well as the SCL v2.01.011 library are automatically part of it.
- If the PSL library v5.00.06 is delivered, the RSA, EC and Base v2.07.003 libraries as well as the SCL v2.04.003 library and the HCL library are automatically part of it.

In case of deselecting one or several of these libraries the TOE does not provide the respective functionality.

2.10 TOE initialization with Customer Software

Beside the various TOE configurations further possibilities of how the user inputs his software on the TOE are in place. This provides a maximum of flexibility and for this an overview is given in the following table:

Table 5 Options to implement user software at Infineon production premises

1	The user or/and a subcontractor downloads the software into the SOLID FLASH™ NVM	The Flash Loader can be activated or reactivated by the user or subcontractor to

	memory on his own. Infineon Technologies AG has not received user software and there are no user data in the ROM.	download his software in the SOLID FLASH™ NVM memory.
2	The user provides software for the download into the SOLID FLASH™ NVM memory to Infineon Technologies AG. The software is downloaded to the SOLID FLASH™ NVM memory during chip production. There are no user data in the ROM.	The Flash Loader is deactivated.
3	The user provides software for the download into the SOLID FLASH™ NVM memory to Infineon Technologies AG. The software is downloaded to the SOLID FLASH™ NVM memory during chip production. There are no user data in the ROM	The Flash Loader is blocked afterwards but can be activated or reactivated by the user or subcontractor to download his software in the SOLID FLASH™ NVM memory. Precondition is that the user has provided an own reactivation procedure in software prior chip production to Infineon Technologies AG.

The Generic Chip Identification Mode (GCIM) data of the TOE allows a unique identification of each TOE and provides several detailed production information. The Chip Identification Mode data is accessible by a non-ISO reset or can be read directly from the configuration area located at the NVM by the user operating system. The SLE97 Hardware Reference Manual [7] gives a detailed description of the GCIM data.

3 Conformance Claims (ASE_CCL)

3.1 CC Conformance Claim

This Security Target (ST) and the TOE claim conformance to Common Criteria version v3.1 part 1 [2], part 2 [3] and part 3 [4].

Conformance of this ST is claimed for:
Common Criteria part 2 extended and Common Criteria part 3 conformant.

3.2 PP Claim

This Security Target is in **strict conformance** to the
Security IC Platform Protection Profile [1].

The Security IC Platform Protection Profile is registered and certified by the Bundesamt für Sicherheit in der Informationstechnik[1] (BSI) under the reference BSI-PP-0035, Version 1.0, dated 15.06.2007.

The security assurance requirements of the TOE are according to the Security IC Platform Protection Profile [1]. They are all drawn from Part 3 of the Common Criteria version v3.1.

The augmentations of the PP [1] are listed below.

Table 6 Augmentations of the assurance level of the TOE

Assurance Class	Assurance components	Description
Life-cycle support	ALC_DVS.2	Sufficiency of security measures
Vulnerability assessment	AVA_VAN.5	Advanced methodical vulnerability analysis

3.3 Package Claim

This Security Target does not claim conformance to a package of the PP [1].

The assurance level for the TOE is EAL5 augmented with the components ALC_DVS.2 and AVA_VAN.5.

[1] Bundesamt für Sicherheit in der Informationstechnik (BSI) is the German Federal Office for Information Security

3.4 Conformance Rationale

This security target claims strict conformance only to one PP, the PP [1].

The Target of Evaluation (TOE) is a typical security IC as defined in PP chapter 1.2.2 comprising:

- the circuitry of the IC (hardware including the physical memories),
- configuration data, initialisation data related to the IC Dedicated Software and the behaviour of the security functionality
- the IC Dedicated Software with the parts
- the IC Dedicated Test Software,
- the IC Dedicated Support Software.

The TOE is designed, produced and/or generated by the TOE Manufacturer.

Security Problem Definition:

Following the PP [1], the security problem definition is enhanced by adding an additional threat, an organization security policy and an augmented assumption. Including these add-ons, the security problem definition of this security target is consistent with the statement of the security problem definition in the PP [1], as the security target claimed strict conformance to the PP [1].

Conformance Rationale:

The augmented organizational security policy P.Add-Functions, coming from the additional security functionality of the cryptographic libraries, the augmented assumption A.Key-Function, related to the usage of key-depending function, and the threat memory access violation , due to specific TOE memory access control functionality, have been added. These add-ons have no impact on the conformance statements regarding CC [2] and PP [1], with following rational:

The security target remains conformant to CC [2], claim 482 as the possibility to introduce additional restrictions is given.

The security target fulfils the strict conformance claim of the PP [1] due to the application notes 5, 6 and 7 which apply here. By those notes the addition of further security functions and security services are covered, even without deriving particular security functionality from a threat but from a policy.

Due to additional security functionality, one coming from the cryptographic libraries - O.Add-Functions, the memory access control - O.Mem-Access, and the hash additional security objectives have been introduced. These add-ons have no impact on the conformance statements regarding CC [2] and PP [1], with following rational:

The security target remains conformant to CC [2], claim 482 as the possibility to introduce additional restrictions is given.

The security target fulfils the strict conformance of the PP [1] due to the application note 9 applying here. This note allows the definition of high-level security goals due to further functions or services provided to the Security IC Embedded Software.

Therefore, the security objectives of this security target are consistent with the statement of the security objectives in the PP [1], as the security target claimed strict conformance to the PP [1].

All security functional requirements defined in the PP [1] are included and completely defined in this ST. The security functional requirements listed in the following are all taken from Common Criteria part 2 [3] and additionally included and completely defined in this ST:

- FDP_ACC.1 “Subset access control”
- FDP_ACF.1 “Security attribute based access control”
- FMT_MSA.1“Management of security attributes”
- FMT_MSA.3“Static attribute initialisation”
- FMT_SMF.1“Specification of Management functions”
- FCS_COP.1 “Cryptographic support”
- FCS_CKM.1 “Cryptographic key generation”
- FDP_SDI.1 “Stored data integrity monitoring
- FDP_SDI.2 “Stored data integrity monitoring and action

The security functional requirement

- FPT_TST.2 “Subset TOE security testing“(Requirement from [3])
- FCS_RNG.1 “Generation of Random Numbers”

is included and completely defined in this ST, section 6.

All assignments and selections of the security functional requirements are done in the PP [1] and in this security target in section 7.5.

The Assurance Requirements of the TOE obtain the Evaluation Assurance Level 5 augmented with the assurance components ALC_DVS.2 and AVA_VAN.5 for the TOE.

3.5 Application Notes

The functional requirement FCS_RNG.1 is a refinement of the FCS_RNG.1 defined in the Protection Profile [1] according to “Anwendungshinweise und Interpretationen zum Schema (AIS)” [15].

4 Security Problem Definition (ASE_SPD)

The content of the PP [1] applies to this chapter completely.

4.1 Threats

The threats are directed against the assets and/or the security functions of the TOE. For example, certain attacks are only one step towards a disclosure of assets while others may directly lead to a compromise of the application security. The more detailed description of specific attacks is given later on in the process of evaluation and certification. An overview on attacks is given in PP [1] section 3.2.

The threats to security are defined and described in PP [1] section 3.2.

Table 7 Threats according PP [1]

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

4.1.1 Additional Threat due to TOE specific Functionality

The additional functionality of introducing sophisticated privilege levels and access control allows the secure separation between the operation system(s) and applications, the secure downloading of applications after personalization and enables multitasking by separating memory areas and performing access controls between different applications. Due to this additional functionality "area based memory access control" a new threat is introduced.

The Smartcard Embedded Software is responsible for its User Data according to the assumption "Treatment of User Data (A.Resp-Appl)". However, the Smartcard Embedded Software may comprise different parts, for instance an operating system and one or more applications. In this case, such parts may accidentally or deliberately access data (including code) of other parts, which may result in a security violation.

The TOE shall avert the threat "Memory Access Violation (T.Mem-Access)" as specified below.

T.Mem-Access Memory Access Violation

Parts of the Smartcard Embedded Software may cause security violations by accidentally or deliberately accessing restricted data (which may include code) or privilege levels. Any restrictions are defined by the security policy of the specific application context and must be implemented by the Smartcard Embedded Software.

Table 8 Additional threats due to TOE specific functions and augmentations

T.Mem-Access	Memory Access Violation

For details see PP [1] section 3.2.

4.1.2 Assets regarding the Threats

The primary assets concern the User Data which includes the user data as well as program code (Security IC Embedded Software) stored and in operation and the provided security services. These assets have to be protected while being executed and or processed and on the other hand, when the TOE is not in operation.

This leads to four primary assets with its related security concerns:

- SC1 Integrity of User Data and of the Security IC Embedded Software (while being executed/processed and while being stored in the TOE's memories),
- SC2 Confidentiality of User Data and of the Security IC Embedded Software (while being processed and while being stored in the TOE's memories)
- SC3 Correct operation of the security services provided by the TOE for the Security IC Embedded Software.
- SC4 Continuous availability of random numbers

SC4 is an additional security service provided by this TOE which is the availability of random numbers. These random numbers are generated either by a true random number or a deterministic random number generator or by both, when a true random number is used as seed for the deterministic random number generator. Note that the generation of random numbers is a requirement of the PP [1].

To be able to protect the listed assets the TOE shall protect its security functionality as well. Therefore critical information about the TOE shall be protected. Critical information includes:

- logical design data, physical design data, IC Dedicated Software, and configuration data
- Initialisation Data and Pre-personalisation Data, specific development aids, test and characterisation related data, material for software development support, and reticles.

The information and material produced and/or processed by the TOE Manufacturer in the TOE development and production environment (Phases 2 up to TOE Delivery) can be grouped as follows:

- logical design data,
- physical design data,
- IC Dedicated Software, Security IC Embedded Software, Initialisation Data and Pre-personalisation Data,
- specific development aids,
- test and characterisation related data,
- material for software development support, and
- reticles and products in any form

as long as they are generated, stored, or processed by the TOE Manufacturer.

For details see PP [1] section 3.1.

4.2 Organizational Security Policies

The TOE has to be protected during the first phases of their lifecycle (phases 2 up to TOE delivery which can be after phase 3 or phase 4). Later on each variant of the TOE has to protect itself. The organizational security policy covers this aspect.

P.Process-TOE Protection during TOE Development and Production

An accurate identification must be established for the TOE. This requires that each instantiation of the TOE carries this unique identification.

The organizational security policies are defined and described in PP [1] section 3.3. Due to the augmentations of PP [1] an additional policy is introduced and described in the next chapter.

Table 9 Organizational Security Policies according PP [1]

P.Process-TOE	Protection during TOE Development and Production

4.2.1 Augmented Organizational Security Policy

Due to the augmentations of the PP [1] an additional policy is introduced.

The TOE provides specific security functionality, which can be used by the Smartcard Embedded Software. In the following specific security functionality is listed which is not derived from threats identified for the TOE's environment because it can only be decided in the context of the smartcard application, against which threats the Smartcard Embedded Software will use the specific security functionality.

The IC Developer / Manufacturer must apply the policy "Additional Specific Security Functionality (P.Add-Functions)" as specified below.

P.Add-Functions	Additional Specific Security Functionality

The TOE shall provide the following specific security functionality to the Smartcard Embedded Software:

- Advanced Encryption Standard (AES)
- Triple Data Encryption Standard (3DES)
- Rivest-Shamir-Adleman Cryptography (RSA)
- Elliptic Curve Cryptography (EC)
- Hash Cryptographic Functions (SHA)

Note:This TOE can be delivered with the SCP accessible or blocked. The blocking depends on the customer demands prior to the production of the hardware. In case the SCP is blocked, no 3DES or AES computation supported by hardware is possible. The 3DES and AES functionality has then to be removed from this policy.

Note:The TOE can also be delivered with an optional SCL. Any optional SCL contains AES and 3DES algorithms with additional security countermeasures. The optional SCL needs an accessible SCP. The 3DES and AES functionality has then to be removed from this policy.

Note:This TOE can be delivered with the Crypto2304T coprocessor accessible or blocked. In case the Crypto2304T is blocked, no RSA or ECC computation supported by hardware is possible. The RSA and ECC functionality has then to be removed from this policy.

Note:The TOE can also be delivered with the optional RSA library. The optional RSA library needs an accessible Crypto2304T. If the optional RSA library is not delivered then RSA functionality has to be removed from this policy.

Note:The TOE can also be delivered with the optional ECC library. The optional ECC library needs an accessible Crypto2304T. If the optional ECC library is not delivered then ECC functionality has to be removed from this policy.

Note:The TOE can be delivered with the optional HCL library. If the optional HCL library is not delivered then SHA functionality has to be removed from this policy.

4.3 Assumptions

The TOE assumptions on the operational environment are defined and described in PP [1] section 3.4.

The assumptions concern the phases where the TOE has left the chip manufacturer.

A.Process-Sec-IC Protection during Packaging, Finishing and Personalization:
It is assumed that security procedures are used after delivery of the TOE by the TOE Manufacturer up to delivery to the end-consumer to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use).

A.Plat-Appl Usage of Hardware Platform:
The Security IC Embedded Software is designed so that the requirements from the following documents are met: (i) TOE guidance documents (refer to the Common Criteria assurance class AGD) such as the hardware data sheet, and the hardware application notes, and (ii) findings of the TOE evaluation reports relevant for the Security IC Embedded Software as documented in the certification report.

A.Resp-Appl Treatment of User Data:
All User Data are owned by Security IC Embedded Software. Therefore, it must be assumed that security relevant User Data (especially cryptographic keys) are treated by the Security IC Embedded Software as defined for its specific application context.

The support of cipher schemas needs to make an additional assumption.

Table 10 Assumption according PP [1]

A.Process-Sec-IC	Protection during Packaging, Finishing and Personalization
A.Plat-Appl	Usage of Hardware Platform
A.Resp-Appl	Treatment of User Data

4.3.1 Augmented Assumptions

The developer of the Smartcard Embedded Software must ensure the appropriate “Usage of Key-dependent Functions (A.Key-Function)” while developing this software in Phase 1 as specified below.

A.Key-Function	Usage of Key-dependent Functions

Key-dependent functions (if any) shall be implemented in the Smartcard 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 Smartcard Embedded Software. In contrast to this, the threats T.Leak-Inherent and T.Leak-Forced address (i) the cryptographic routines which are part of the TOE (For details see PP [1] section 3.4.).

5 Security objectives (ASE_OBJ)

This section shows the subjects and objects where are relevant to the TOE.
A short overview is given in the following.

The user has the following standard high-level security goals related to the assets:

- SG1 maintain the integrity of User Data and of the Security IC Embedded Software
- SG2 maintain the confidentiality of User Data and of the Security IC Embedded Software
- SG3 maintain the correct operation of the security services provided by the TOE for the Security IC Embedded Software
- SG4 provision of random numbers.

5.1 Security objectives for the TOE

The security objectives of the TOE are defined and described in PP [1] section 4.1.

Table 11 Objectives for the TOE according to PP [1]

O.Phys-Manipulation	Protection against Physical Manipulation
O.Phys-Probing	Protection against Physical Probing
O.Malfunction	Protection against Malfunction
O.Leak-Inherent	Protection against Inherent Information Leakage
O.Leak-Forced	Protection against Forced Information Leakage
O.Abuse-Func	Protection against Abuse of Functionality
O.Identification	TOE Identification
O.RND	Random Numbers

The TOE provides “Additional Specific Security Functionality (O.Add-Functions)” as specified below.

O.Add-Functions : Additional Specific Security Functionality

The TOE must optionally provide the following specific security functionality to the Smartcard Embedded Software:

- Advanced Encryption Standard (AES)
- Triple Data Encryption Standard (3DES)
- Rivest-Shamir-Adleman (RSA)
- Elliptic Curve Cryptography (EC)
- Hash Cryptographic functions (SHA)

The hardware of this TOE can be delivered with the following configuration options:

- both crypto co-processors accessible
- with a blocked SCP
- with a blocked Crypto2304T
- both crypto co-processors blocked

In case the SCP is blocked, no AES and 3DES computations supported by hardware are possible. In the case the Crypto2304T is blocked, no RSA and EC computations supported by hardware are possible.

The optional security relevant software part of the TOE consists of the following optional libraries:

- RSA Cryptographic Library
- EC Cryptographic Library
- Symmetric Cryptographic Library (SCL)
- Hash cryptographic library (HCL)
- Platform Support Library (PSL)

The TOE shall provide "Area based Memory Access Control (O.Mem-Access)" as specified below.

O.Mem-Access: Area based Memory Access Control

The TOE must provide the Smartcard Embedded Software with the capability to define restricted access memory areas. The TOE must then enforce the partitioning of such memory areas so that access of software to memory areas and privilege levels is controlled as required, for example, in a multi-application environment.

Table 12 Additional objectives due to TOE specific functions and augmentations

O.Add-Functions	Additional specific security functionality
O.Mem-Access	Area based Memory Access Control

5.2 Security Objectives for the development and operational Environment

The security objectives for the security IC embedded software development environment and the operational environment is defined in PP [1] section 4.2 and 4.3. The table below lists the security objectives.

Table 13 Security objectives for the environment according to PP [1]

Phase 1	OE.Plat-Appl	Usage of Hardware Platform
	OE.Resp-Appl	Treatment of User Data
Phase 5 – 6 optional Phase 4	OE.Process-Sec-IC	Protection during composite product manufacturing

5.2.1 Clarification of "Usage of Hardware Platform (OE.Plat-Appl)"

Regarding the cryptographic services this objective of the environment has to be clarified. The TOE supports cipher schemes as additional specific security functionality. If required the Smartcard Embedded Software shall use these cryptographic services of the TOE and their interface as specified. When key-dependent functions implemented in the Smartcard Embedded Software are just being executed, the Smartcard Embedded Software must provide protection against disclosure of confidential data (User Data) stored and/or processed in the TOE by using the methods described under "Inherent Information Leakage (T.Leak-Inherent)" and "Forced Information Leakage (T.Leak-Forced)".

The objectives of the environment regarding the memory, software and firmware protection and the SFR and peripheral-access-rights-handling have to be clarified. For the separation of different applications the Smartcard Embedded Software (Operating System) may implement a memory management scheme based upon security functions of the TOE.

5.2.2 Clarification of "Treatment of User Data (OE.Resp-Appl)"

Regarding the cryptographic services this objective of the environment has to be clarified. By definition cipher or plain text data and cryptographic keys are User Data. The Smartcard Embedded Software shall treat these data appropriately, use only proper secret keys (chosen from a large key space) as input for the cryptographic function 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. The keys must be unique with a very high probability, as well as cryptographically strong. For example, it must be ensured that it is beyond practicality to derive the private key from a public key if asymmetric algorithms are used. 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 realized in the environment.

Regarding the memory, software and firmware protection and the SFR and peripheral access rights handling these objectives of the environment has to be clarified. The treatment of User Data is also required when a multi-application operating system is implemented as part of the Smartcard Embedded Software on the TOE. In this case the multi-application operating system should not disclose security relevant user data of one application to another application when it is processed or stored on the TOE.

5.2.3 Clarification of "Protection during Composite product manufacturing (OE.Process-Sec-IC)"

The protection during packaging, finishing and personalization includes also the personalization process (Flash Loader software) and the personalization data (TOE software components) during Phase 4, Phase 5 and Phase 6.

5.3 Security Objectives Rationale

The security objectives rationale of the TOE are defined and described in PP [1] section 4.4. For organizational security policy P.Add-Functions, OE.Plat-Appl and OE.Resp-Appl the rationale is given in the following description.

Table 14 Security Objective Rationale

Assumption, Threat or Organisational Security Policy	Security Objective
P.Add-Functions	O.Add-Functions
A.Key-Function	OE.Plat-Appl OE.Resp-Appl
T.Mem-Access	O.Mem-Access

The justification related to the security objective "Additional Specific Security Functionality (O.Add-Functions)" is as follows: Since O.Add-Functions requires the TOE to implement exactly the same specific security functionality as required by P.Add-Functions; the organizational security policy is covered by the objective.

Nevertheless the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced define how to implement the specific security functionality required by P.Add-Functions. (Note that these objectives support that the specific security functionality is provided in a secure way as expected from P.Add-Functions.) Especially O.Leak-Inherent and O.Leak-Forced refer to the protection of confidential data (User Data or TSF data) in general. User Data are also processed by the specific security functionality required by P.Add-Functions.

Compared to PP [1] clarification has been made for the security objective "Usage of Hardware Platform (OE.Plat-Appl)": If required the Smartcard Embedded Software shall use these cryptographic services of the TOE and their interface as specified. In addition, the Smartcard Embedded Software must implement functions which perform operations on keys (if any) in such a manner that they do not disclose information about confidential data. The non disclosure due to leakage A.Key-Function attacks is included in this objective OE.Plat-Appl. This addition ensures that the assumption A.Plat-Appl is still covered by the objective OE.Plat-Appl although additional functions are being supported according to O.Add-Functions.

Compared to the PP [1] a clarification has been made for the security objective "Treatment of User Data (OE.Resp-Appl)": By definition cipher or plain text data and cryptographic keys are User Data. So, the Smartcard 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 realized in the environment. That is expressed by the assumption A.Key—Function which is covered from OE.Resp–Appl. These measures make sure that the assumption A.Resp-Appl is still covered by the security objective OE.Resp-Appl although additional functions are being supported according to P.Add-Functions.

Compared to the PP [1] an enhancement regarding memory area protection has been established. The clear definition of privilege levels for operated software establishes the clear separation of different restricted memory areas for running the firmware, downloading and/or running the operating system and to establish a clear separation between different applications. Nevertheless, it is also possible to define a shared memory section where separated applications may exchange defined data. The privilege levels clearly define by using a hierarchical model the access right from one level to the other. These measures ensure that the threat T.Mem-Access is clearly covered by the security objective O.Mem-Access.

The justification of the additional policy and the additional assumption show that they do not contradict to the rationale already given in the Protection Profile for the assumptions, policy and threats defined there.

6 Extended Component Definition (ASE_ECD)

There are four extended components defined and described for the TOE:

- the family **FCS_RNG** at the class FCS Cryptographic Support
- the family **FMT_LIM** at the class FMT Security Management
- the family **FAU_SAS** at the class FAU Security Audit
- the component **FPT_TST.2** at the class FPT Protection of the TSF

The extended components FMT_LIM and FAU_SAS are defined and described in PP [1] section 5. The components FPT_TST.2 and FCS_RNG are defined in the following sections.

6.1 "Subset TOE security testing (FPT_TST)"

The security is strongly dependent on the correct operation of the security functions. Therefore, the TOE shall support that particular security functions or mechanisms are tested in the operational phase (Phase 7). The tests can be initiated by the Smartcard Embedded Software and/or by the TOE or is done automatically and continuously.

Part 2 of the Common Criteria provides the security functional component "TSF testing (FPT_TST.1)". The component FPT_TST.1 provides the ability to test the TSF's correct operation.

For the user it is important to know which security functions or mechanisms can be tested. The functional component FPT_TST.1 does not mandate to explicitly specify the security functions being tested. In addition, FPT_TST.1 requires verification of the integrity of TSF data and of the stored TSF executable code which might violate the security policy. Therefore, the functional component**"Subset TOE security testing (FPT_TST.2)"** of the family TSF self test has been newly created. This component allows that particular parts of the security mechanisms and functions provided by the TOE are tested.

6.2 Definition of FPT_TST.2

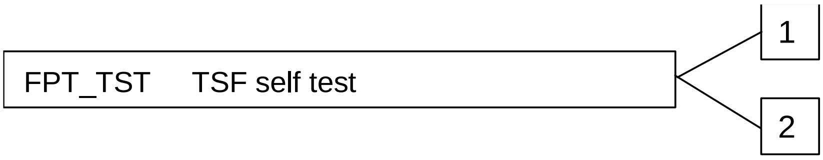
The functional component "Subset TOE security testing (FPT_TST.2)" has been newly created (Common Criteria Part 2 extended). This component allows that particular parts of the security mechanisms and functions provided by the TOE can be tested after TOE Delivery or are tested automatically and continuously during normal operation transparent for the user.
This security functional component is used instead of the functional component FPT_TST.1 from Common Criteria Part 2. For the user it is important to know which security functions or mechanisms can be tested. The functional component FPT_TST.1 does not mandate to explicitly specify the security functions being tested. In addition, FPT_TST.1 requires verifying the integrity of TSF data and stored TSF executable code which might violate the security policy.

The functional component "Subset TOE testing (FPT_TST.2)" is specified as follows (Common Criteria Part 2 extended).

6.3 TSF self test (FPT_TST)

Family Behavior The Family Behavior is defined in [3] section 15.14 (442, 443).

Component leveling



FPT_TST.1 The component FPT_TST.1 is defined in [3] section 15.14 (444, 445, 446).

FPT_TST.2 Subset TOE security testing, provides the ability to test the correct operation of particular security functions or mechanisms. These tests may be performed at start-up, periodically, at the request of the authorized user, or when other conditions are met. It also provides the ability to verify the integrity of TSF data and executable code.

Management: FPT_TST.2
The following actions could be considered for the management functions in FMT: management of the conditions under which subset TSF self testing occurs, such as during initial start-up, regular interval or under specified conditionsmanagement of the time of the interval appropriate.

Audit: FPT_TST.2
There are no auditable events foreseen.

FPT_TST.2 Subset TOE testing

Hierarchical to: No other components.

Dependencies: No dependencies

FPT_TST.2.1 The TSF shall run a suite of self tests [selection: during initial start-up, periodically during normal operation, at the request of the authorized user, and/or at the conditions [assignment: conditions under which self test should occur]] to demonstrate the correct operation of [assignment: functions and/or mechanisms].

6.4 Family “Generation of Random Numbers (FCS_RNG)”

The component “Generation of Random Numbers (FCS_RNG.1)” has to be newly created according the new version of the “Anwendungshinweise und Interpretationen zum Schema (AIS)” [15]. This security functional component is used instead of the functional component FCS_RNG.1 defined in the protection profile [1].

The component “Generation of Random Numbers (FCS_RNG.1)” is specified as follows (Common Criteria Part 2 extended).

6.5 Definition of FCS_RNG.1

This section describes the functional requirements for the generation of random numbers, which may be used as secrets for cryptographic purposes or authentication. The IT security functional requirements for the TOE are defined in an additional family (FCS_RNG) of the Class FCS (Cryptographic support).

FCS_RNG Generation of random numbers

Family Behaviour

This family defines quality requirements for the generation of random numbers that are intended to be used for cryptographic purposes.

Component levelling:

FCS_RNG: Generation of random numbers
1

FCS_RNG.1 Generation of random numbers, requires that the random number generator I mplements defined security capabilities and that the random numbers meet a defined quality metric.

Management: FCS_RNG.1
There are no management activities foreseen.

Audit: FCS_RNG.1
There are no actions defined to be auditable.

FCS_RNG.1 Random number generation

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1: The TSF shall provide a [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*] random number generator that implements: [assignment: *list of security capabilities*].

FCS_RNG.1.2: The TSF shall provide random numbers that meet [assignment: *a defined quality metric*].

Note: The functional requirement FCS_RNG.1 is a refinement of the FCS_RNG.1 defined in the Protection Profile [1] according to "Anwendungshinweise und Interpretationen zum Schema (AIS)" [15].

7 Security Requirements (ASE_REQ)

For this section the PP [1] section 6 can be applied completely.

7.1 TOE Security Functional Requirements

The security functional requirements (SFR) for the TOE are defined and described in the PP [1] section 6.1 and in the following description.

The Table 15 provides an overview of the functional security requirements of the TOE, defined in the in PP [1] section 6.1. In the last column it is marked if the requirement is refined. The refinements are also valid for this ST.

Table 15 Security functional requirements defined in PP [1]

Security Functional Requirement		Refined in PP [1]
FRU_FLT.2	Limited fault tolerance	Yes
FPT_FLS.1	Failure with preservation of secure state	Yes
FMT_LIM.1	Limited capabilities	No
FMT_LIM.2	Limited availability	No
FAU_SAS.1	Audit storage	No
FPT_PHP.3	Resistance to physical attack	Yes
FDP_ITT.1	Basic internal transfer protection	Yes
FPT_ITT.1	Basic internal TSF data transfer protection	Yes
FDP_IFC.1	Subset information flow control	No

The Table 16 provides an overview about the augmented security functional requirements, which are added additional to the TOE and defined in this ST. All requirements are taken from Common Criteria Part 2 [3], with the exception of the requirement FPT_TST.2 and FCS_RNG.1, which are defined in this ST completely.

Table 16 Augmented security functional requirements

	Security Functional Requirement
FPT_TST.2	Subset TOE security testing
FDP_ACC.1	Subset access control
FDP_ACF.1	Security attribute based access control
FMT_MSA.1	Management of security attributes
FMT_MSA.3	Static attribute initialization
FMT_SMF.1	Specification of Management functions
FCS_COP.1	Cryptographic support
FCS_CKM.1	Cryptographic key generation
FDP_SDI.1	Stored data integrity monitoring
FDP_SDI.2	Stored data integrity monitoring and action
FCS_RNG.1	Quality metric for random numbers

All assignments and selections of the security functional requirements of the TOE are done in PP [1] and in the following description.

The above marked extended components FMT_LIM.1 and FMT_LIM.2 are introduced in PP [1] to define the IT security functional requirements of the TOE as an additional family (FMT_LIM) of the Class FMT (Security Management). This family describes the functional requirements for the Test Features of the TOE. The new functional requirements were defined in the class FMT because this class addresses the management of functions of the TSF.

The additional component FAU.SAS is introduced to define the security functional requirements of the TOE of the Class FAU (Security Audit). This family describes the functional requirements for the storage of audit data and is described in the next chapter.

The requirement FPT_TST.2 is the subset of TOE testing and originated in [3]. This requirement is given as the correct operation of the security functions is essential. The TOE provides mechanisms to cover this requirement by the smartcard embedded software and/or by the TOE itself.

7.1.1 Extended Components FCS_RNG.1 and FAU_SAS.1

7.1.1.1 FCS_RNG

To define the IT security functional requirements of the TOE an additional family (FCS_RNG) of the class FCS (cryptographic support) is defined in chapter 6.5. This family describes the functional requirements for random number generation used for cryptographic purposes.

FCS_RNG.1/HW Random Number Generation

Hierarchical to: No other components

Dependencies: No dependencies

FCS_RNG.1 Random numbers generation Class PTG.2 according to [6]

FCS_RNG.1.1 The TSF shall provide a physical random number generator which 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.

PTG.2.3: The online test shall detect non-tolerable statistical defects of the rawrandom 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 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 num ber sequence. It is triggered continuously. 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.

FCS_RNG.1.2 The TSF shall provide numbers in the format 8- or 16-bit that meet

PTG.2.6: Test procedure A, as defined in [6] 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 functional requirement FCS_RNG.1/HW is a refinement of the FCS_RNG.1 defined in chapter 6.5

Note:

FCS_RNG.1/PSL Random Number Generation

Hierarchical to: No other components

Dependencies: No dependencies

FCS_RNG.1 Random numbers generation Class PTG.2 according to [6]

FCS_RNG.1.1 The TSF shall provide a physical random number generator which 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.

PTG.2.3: The online test shall detect non-tolerable statistical defects of the rawrandom 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 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 num ber sequence. It is triggered continuously. 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.

FCS_RNG.1.2 The TSF shall provide a number n of caller requested bytes (n = 0…2^{32}, 4 | n) , that meet

PTG.2.6: Test procedure A, as defined in [6] 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 functional requirement FCS_RNG.1/PSL is a refinement of the FCS_RNG.1 defined in chapter 6.5.

Note:The TOE can be delivered with the optional PSL library v4.00.10 and v5.00.06. If none of those optional PSL libraries is available then this SFR is not applicable.

7.1.1.2 FAU_SAS

To define the security functional requirements of the TOE an additional family (FAU_SAS) of the Class FAU (Security Audit) is defined here. This family describes the functional requirements for the storage of audit data. It has a more general approach than FAU_GEN, because it does not necessarily require the data to be generated by the TOE itself and because it does not give specific details of the content of the audit records.
The TOE shall meet the requirement “Audit storage (FAU_SAS.1)” as specified below (Common Criteria Part 2 extended).

FAU_SAS.1 Audit Storage

Hierarchical to: No other components

Dependencies: No dependencies.

FAU_SAS.1.1 The TSF shall provide the test process before TOE Delivery with the capability to store the Initialization Data and/or Pre-personalization Data and/or supplements of the Security IC Embedded Software in the not changeable configuration page area and non-volatile memory.

7.1.2 Subset of TOE testing

The security is strongly dependent on the correct operation of the security functions. Therefore, the TOE shall support that particular security functions or mechanisms are tested in the operational phase (Phase 7). The tests can be initiated by the Smartcard Embedded Software and/or by the TOE.
The TOE shall meet the requirement “Subset TOE testing (FPT_TST.2)” as specified below (Common Criteria Part 2 extended).

FPT_TST.2 Subset TOE testing

Hierarchical to: No other components

Dependencies: No dependencies

FPT_TST.2.1 The TSF shall run a suite of self tests at the request of the authorized user to demonstrate the correct operation of the alarm lines and/or the environmental sensor mechanisms

7.2 Memory access control

Usage of multiple applications in one Smartcard often requires code and data separation in order to prevent that one application can access code and/or data of another application. For this reason the TOE provides Area based Memory Access Control. The underlying Memory Protection Unit (MPU) is documented in section 4 of the [7].

The security service being provided is described in the Security Function Policy (SFP) **Memory Access Control Policy**. The security functional requirement **"Subset access control (FDP_ACC.1)"** requires that this policy is in place and defines the scope were it applies. The security functional requirement **"Security attribute based access control (FDP_ACF.1)"** defines security attribute usage and characteristics of policies. It describes the rules for the function that implements the Security Function Policy (SFP) as identified in FDP_ACC.1. The decision whether an access is permitted or not is taken based upon attributes allocated to the software. The Smartcard Embedded Software defines the attributes and memory areas. The corresponding permission control information is evaluated "on-the-fly" by the hardware so that access is granted/effective or denied/inoperable.

The security functional requirement **"Static attribute initialisation (FMT_MSA.3)"** ensures that the default values of security attributes are appropriately either permissive or restrictive in nature. Alternative values can be specified by any subject provided that the **Memory Access Control Policy** allows that. This is described by the security functional requirement **"Management of security attributes (FMT_MSA.1)"**. The attributes are determined during TOE manufacturing (FMT_MSA.3) or set at run-time (FMT_MSA.1).

From TOE's point of view the different roles in the Smartcard Embedded Software can be distinguished according to the memory based access control. However the definition of the roles belongs to the user software.

The following Security Function Policy (SFP) **Memory Access Control Policy** is defined for the requirement "Security attribute based access control (FDP_ACF.1)":

7.2.1 Memory Access Control Policy

The TOE shall support the standard ARMv7 Protected Memory System Architecture model.
The MPU provides full support for:

- Protection regions.
- Overlapping protection regions, with ascending region priority:
  - Region 7 = highest priority.
  - Region 0 = lowest priority.
- Access permissions.
- MPU mismatches and permission violations invoke the programmable-priority MemManage fault handler.

The MPU can be used to:

- Enforce privilege rules, preventing user applications from corrupting operating system data.
- Separate processes, blocking the active task from accessing other tasks' data.
- Enforce access rules, allowing memory regions to be defined as read-only or detecting unexpected memory accesses.

Subjects, Objects and Operations of the policy

- Subjects: privilege or non-privilege level of the ARM processor
- Objects: memory/code addresses
- Operations: Read a/o write a/o execute access

Attributes of the policy:

- MPU enable/disable bit.

- 8 regions with the following attributes
  - A unique priority
  - The enable bit
  - the start address and size
  - an access matrix which defines if an Operation of a Subject to an Object lying in the region is allowed or denied
- The default region with the following security attribute:
  - A bit which defines if an Operation for the Subject (privilege level) is allowed or if no Operation is allowed for any Subject.

Roles of the policy:
The roles correspond 1-1 to the subjects.

Properties of the policy:

- If an address is contained in multiple enabled regions, then the region with the highest priority defines the access rights.
- If an address is contained in no region then the default region defines the access rights.
- The region defining the access rights checks in the access matrix if the Subject has access to the Object with respect to the desired Operation. In case the access is denied the MPU throws an access violation exception.

The TOE shall meet the requirement “Subset access control (FDP_ACC.1)” as specified below.

FDP_ACC.1 Subset access control

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1 The TSF shall enforce the Memory Access Control Policy on all Subjects, all Objects and all Operations.

The TOE shall meet the requirement “Security attribute based access control (FDP_ACF.1)” as specified below.

FDP_ACF.1 Security attribute based access control

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialization

FDP_ACF.1.1 The TSF shall enforce the Memory Access Control Policy to objects based on the following: As specified in the definition of the memory access control policy .

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:
As specified in the definition of the memory access control policy.

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none.

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: none.

The TOE shall meet the requirement “Static attribute initialisation (FMT_MSA.3)” as specified below.

FMT_MSA.3 Static attribute initialization

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 security roles

FMT_MSA.3.1 The TSF shall enforce the Memory Access Control Policy to provide restrictive[1] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the privilege level to specify alternative initial values to override the default values when an object or information is created.

The TOE shall meet the requirement “Management of security attributes (FMT_MSA.1)” as specified below:

FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control]
FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

FMT_MSA.1.1 The TSF shall enforce the Memory Access Control Policy to restrict the ability to modify any security attributes[2] to the privilege level.

The TOE shall meet the requirement “Specification of management functions (FMT_SMF.1)” as specified below:

FMT_SMF.1 Specification of management functions

Hierarchical to: No other components

Dependencies: No dependencies

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions: The privilege level shall be able to access the configuration registers of the MPU.

[1] The static definition of the access rules is documented in [7]

[2] editorially refined

7.3 Support of Cipher Schemes

The following additional specific security functionality is implemented in the TOE:

FCS_COP.1 Cryptographic operation requires a cryptographic operation to be performed in accordance with a specified algorithm and with a cryptographic key of specified sizes. The specified algorithm and cryptographic key sizes can be based on an assigned standard; dependencies are discussed in Section 7.6.1.1.

The following additional specific security functionality is implemented in the TOE:

- Advanced Encryption Standard (AES)
- Triple Data Encryption Standard (3DES)
- Elliptic Curve Cryptography (EC)
- Rivest-Shamir-Adleman (RSA)[1]
- Hash functions (SHA-x)

General statements with regard to Elliptic Curves:

The EC library is delivered as object code and in this way integrated in the user software. The certification covers the standard NIST [DSS] and Brainpool [ECC] Elliptic Curves with key lengths of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 Bits. Note that there are numerous other curve types, being also secure in terms of side channel attacks on this TOE, which the user can optionally add in the composition certification process.

7.3.1 Triple-DES Operation

The DES Operation of the TOE shall meet the requirement “Cryptographic operation (FCS_COP.1)” as specified below.

FCS_COP.1/DES 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 management]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/DES The TSF shall perform <u>encryption</u> and <u>decryption</u> in accordance with a specified cryptographic algorithm <u>Triple Data Encryption Standard (3DES) in Electronic Codebook Mode (ECB) and in the Cipher Block Chaining Mode (CBC)</u> and with cryptographic key sizes of <u>2 x 56 or 3 x 56 bit</u> that meet the following standards: <u>[N38A], [N867]</u>

Note:The TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, no 3DES computation supported by hardware is possible and this SFR is not applicable.

FCS_COP.1/DES_SCL_1 Cryptographic operation

[1] In case a user deselects the RSA and/or EC library, the TOE provides basic HW-related routines for RSA and/or EC calculations. For a secure library implementation the user has to implement additional countermeasures.

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 management]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/DES_SCL_1 The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Triple Data Encryption Standard (3DES) in Electronic Codebook mode (ECB),the Cipher Block Chaining mode (CBC), Counter mode (CTR) mode and with cryptographic key sizes of 2 x 56 or 3 x 56 bit, that meet the following standards: [N867], [N38A]

Note:This TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, no 3DES computation supported by hardware is possible and this SFR is not applicable.

Note:The TOE can be delivered with an optional SCL library v2.01.011. If this library is not available then this SFR is not applicable.

FCS_COP.1/DES_SCL_2 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 management]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/DES_SCL _2 The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Triple Data Encryption Standard (3DES) in Electronic Codebook mode (ECB),the Cipher Block Chaining mode (CBC), Cipher Feedback mode (CFB) , Counter mode (CTR) mode and with cryptographic key sizes of 2 x 56 or 3 x 56 bit, that meet the following standards: [N867], [N38A]

Note:This TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, no 3DES computation supported by hardware is possible and this SFR is not applicable.

Note:The TOE can be delivered with the optional SCL library v2.02.01. If this libray is not available then this SFR is not applicable.

FCS_COP.1/DES_SCL_3 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 management]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/DES_SCL _3 The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Triple Data Encryption Standard (3DES) in Electronic Codebook mode (ECB), the Cipher Block Chaining mode (CBC), Cipher Feedback mode (CFB), Counter mode (CTR), CMAC mode and with cryptographic key sizes of 2 x 56 or 3 x 56 bit, that meet the following standards: [N867], [N38A], [N38B].

Note:This TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, no 3DES computation supported by hardware is possible and this SFR is not applicable.

Note:The TOE can be delivered with the optional SCL library v2.04.003. If this libray is not available then this SFR is not applicable.

FCS_COP.1/DES_PSL 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 management]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/DES_PSL The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Triple Data Encryption Standard (3DES) in Electronic Codebook Mode (ECB) and in the Cipher Block Chaining Mode (CBC) and with cryptographic key sizes of 2 x 56 or 3 x 56 bit, that meet the following standards: [N867], [N38A]

Note:The TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, then this SFR is not applicable.

Note:The TOE can be delivered with an optional PSL library. If no optional PSL library is available then this SFR is not applicable.

FCS_COP.1/DES_MAC_PSL 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 management]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/DES_MAC_PSL The TSF shall perform MAC calculation in accordance with a specified cryptographic algorithm Triple Data Encryption Standard (3DES) in CBC MAC mode and cryptographic key sizes of 2 x 56 or 3 x 56 bit that meet the following standards: [N867], [9797] with the following options/modifications:

- MAC algorithm 1
- Padding must be done by the caller
- An Initialization Vector (IV) must be given by the caller

Note: The TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, then this SFR is not applicable.

Note: The TOE can be delivered with an optional PSL library. If no optional PSL library is available then this SFR is not applicable.

7.3.2 AES Operation

The AES Operation of the TOE shall meet the requirement "Cryptographic operation (FCS_COP.1)" as 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 encryption and decryption in accordance with a specified cryptographic algorithm : Advanced Encryption Standard (AES) in Electronic Codebook Mode (ECB) and in the Cipher Block Chaining Mode (CBC) and cryptographic key sizes of 128 bit or 192 bit or 256 bit that meet the following standards: [N197], [N38A]

Note: The TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, no AES computation supported by hardware is possible and this SFR is not applicable.

FCS_COP.1/AES_SCL_1 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_SCL_1 The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Advanced Encryption Standard (AES) in Electronic Codebook mode (ECB), Cipher Block Chaining mode (CBC), CTR(counter) mode and cryptographic key sizes of 128 bit or 192 bit or 256 bit that meet the following standards: [N197], [N38A]

Note: This TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, no AES computation supported by hardware is possible and this SFR is not applicable.

Note: The TOE can be delivered with the optional SCL library v2.01.011. If this libray is not available then this SFR is not applicable.

FCS_COP.1/AES_SCL_2 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_SCL _2 The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Advanced Encryption Standard (AES) in Electronic Codebook mode (ECB), Cipher Block Chaining mode (CBC), Cipher Feedback mode (CFB), CTR(counter) mode and cryptographic key sizes of 128 bit or 192 bit or 256 bit that meet the following standards: [N197], [N38A].

Note:This TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, no AES computation supported by hardware is possible and this SFR is not applicable.

Note:The TOE can be delivered with the optional SCL library v2.02.010. If this libray is not available then this SFR is not applicable.

FCS_COP.1/AES_SCL_3 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_SCL_3 The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm Advanced Encryption Standard (AES) in Electronic Codebook mode (ECB), Cipher Block Chaining mode (CBC), Cipher Feedback mode (CFB), CTR(counter) mode, CMAC mode and cryptographic key sizes of 128 bit or 192 bit or 256 bit that meet the following standards: [N197], [N38A], [N38B]

Note:This TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, no AES computation supported by hardware is possible and this SFR is not applicable.

Note:The TOE can be delivered with the optional SCL library v2.04.003. If this libray is not available then this SFR is not applicable.

FCS_COP.1/AES_PSL 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_PSL a (CBC) The TSF shall perform encryption and decryption in accordance with specified cryptographic algorithm Advanced Encryption Standard (AES) in Electronic Codebook Mode (ECB) and in the Cipher Block Chaining Mode and cryptographic key sizes of 128 bit or 192 bit or 256 bit that meet the following standards: [N197], [N38A]

Note:The TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, then this SFR is not applicable.

Note:The TOE can be delivered with an optional PSL library. If no optional PSL library is available then this SFR is not applicable.

FCS_COP.1/AES_MAC_PSL_1 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_MAC_PSL_1 The TSF shall perform MAC calculation in accordance with a specified cryptographic algorithm Advanced Encryption Standard (AES) in CBC MAC mode and cryptographic key sizes of 128 bit or 192 bit or 256 bit that meet the following standards: [9797], [N197] with the following options/modifications:

- MAC algorithm 1
- Padding must be done by the caller
- An Initialization Vector (IV) must be given by the caller

Note:The TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, then this SFR is not applicable.

Note:The TOE can be delivered with the optional PSL library v4.00.09 and v4.00.10. If none of those optional PSL libraries is available then this SFR is not applicable.

FCS_COP.1/AES_MAC_PSL_2 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_MAC_PSL_2 The TSF shall perform MAC calculation in accordance with a specified cryptographic algorithm Advanced Encryption Standard (AES) in

CBC MAC mode and CMAC mode and cryptographic key sizes of 128 bit or 192 bit or 256 bit that meet the following standards: [9797], [N197], [N38B] with the following options/modifications:

- MAC algorithm 1
- Padding must be done by the caller
- An Initialization Vector (IV) must be given by the caller

Note: The TOE can be delivered with the SCP accessible or blocked. In case the SCP is blocked, then this SFR is not applicable.

Note: The TOE can be delivered with the optional PSL library v5.00.06. If thislibrary is not available than this SFR is not applicable

7.3.3 Rivest-Shamir-Adleman (RSA) operation

The Modular Arithmetic Operation of the TOE shall meet the requirement “Cryptographic operation (FCS_COP.1)” as specified below.

FCS_COP.1/RSA 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/RSA The TSF shall perform encryption, decryption, signature generation and verification in accordance with a specified cryptographic algorithm Rivest-Shamir-Adleman (RSA) and cryptographic key sizes of 1024 - 4096 bit that meet the following standards:

Encryption:
According to section 5.1.1 RSAEP in PKCS v2.2,
without 5.1.1(1).

Decryption (with or without CRT):
According to section 5.1.2 RSADP in PKCS v2.2
for u = 2, i.e., without any (r_i, d_i, t_i), i >2, therefore without 5.1.2(2.b)(ii)&(v), without 5.1.2(1), 5.1.2(2.a) only supported up to $n < 2^{2048}$.

Signature Generation (with or without CRT): According to section 5.2.1 RSASP1 in PKCS v2.2
for u = 2, i.e., without any (r_i, d_i, t_i), i >2,
therefore without 5.2.1(2.b) (ii)&(v), without 5.1.2(1),
5.2.1(2.a) only supported up to $n < 2^{2048}$.

Signature Verification:
According to section 5.2.2 RSAVP1 in PKCS v2.2,
without 5.2.2(1).

Note: This TOE can be delivered with the Crypto2304T coprocessor accessible or blocked. In case the Crypto2304T is blocked, no RSA computation supported by hardware is possible and this SFR is not applicable.

Note: The TOE can be delivered with an optional RSA library. Any optional RSA library contains the RSA algorithms stated above. Any optional RSA library needs an accessible Crypto2304T. If no optional RSA library is available then this SFR is not applicable.

FCS_COP.1/RSA_PSL 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/RSA_PSL The TSF shall perform encryption, decryption, signature generation and verification in accordance with a specified cryptographic algorithm Rivest-Shamir-Adleman (RSA) and cryptographic key sizes of 1024 - 4096 bit that meet the following standards:

Encryption:
According to section 5.1.1 RSAEP in PKCS v2.2,
without 5.1.1(1).

Decryption (with or without CRT):
According to section 5.1.2 RSADP in PKCS v2.2
for u = 2, i.e., without any (r_i, d_i, t_i), i >2, therefore without 5.1.2(2.b) (ii)&(v), without 5.1.2(1), 5.1.2.(2.a), only supported up to n < 2^{2048}

Signature Generation (with or without CRT): According to section 5.2.1 RSASP1 in PKCS v2.2
for u = 2, i.e., without any (r_i, d_i, t_i), i >2,
therefore without 5.2.1(2.b) (ii)&(v), without 5.2.1(1),
5.2.1(2.a) only supported up to n < 2^{2048}

Signature Verification:
According to section 5.2.2 RSAVP1 in PKCS v2.2
without 5.2.2(1).

Note: This TOE can be delivered with the Crypto2304T coprocessor accessible or blocked. In case the Crypto2304T is blocked, no RSA computation supported by hardware is possible and this SFR is not applicable.

Note: The TOE can be delivered with an optional PSL library. In case no PSL library is available then this SFR is not applicable.

7.3.4 Elliptic Curve DSA (ECDSA) operation

The Modular Arithmetic Operation of the TOE shall meet the requirement “Cryptographic operation (FCS_COP.1)” as specified below.

FCS_COP.1/ECDSA 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/ECDSA The TSF shall perform signature generation and signature verification in accordance with a specified cryptographic algorithm ECDSA and cryptographic key sizes 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits that meet the following standard:

Signature Generation:
According to section 7.3 in ANSI X9.62 – 2005
Not implemented is step d) and e) thereof.
The output of step e) has to be provided as input to our function by the caller.
Deviation of step c) and f):
The jumps to step a) were substituted by a return of
the function with an error code, the jumps are emulated by another call to our function.

Signature Verification:
According to section 7.4.1 in ANSI X9.62–2005
Not implemented is step b) and c) thereof.
The output of step c) has to be provided as input to our function by the caller.
Deviation of step d):
Beside noted calculation, our algorithm adds a random multiple of BasepointerOrder n to the calculated values u1 and u2.

Note: This TOE can be delivered with the Crypto2304T coprocessor accessible or blocked. In case the Crypto2304T is blocked, no ECC computation supported by hardware is possible and this SFR is not applicable.

Note: The TOE can be delivered with an optional ECC library. Any optional ECC library contains the ECC algorithms stated above. If no optional ECC library is available then this SFR is not applicable.

7.3.5 Elliptic Curve (EC) key generation

The key generation for the EC shall meet the requirement “Cryptographic key generation (FCS_CKM.1)”

FCS_CKM.1/EC Cryptographic key generation

Hierarchical to: No other components.

Dependencies: FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/EC The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm Elliptic Curve EC specified in ANSI X9.62-2005 and specified cryptographic key sizes 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits that meet the following standard:

ECDSA Key Generation:
According to the appendix A4.3 in ANSI X9.62-2005
the cofactor h is not supported.

Note:This TOE can be delivered with the Crypto2304T coprocessor accessible or blocked. In case the Crypto2304T is blocked, no ECC computation supported by hardware is possible and this SFR is not applicable.

Note:The TOE can be delivered with an optional ECC library. Any optional ECC library contains the ECC algorithms stated above. If no optional ECC library is available then this SFR is not applicable.

7.3.6 Elliptic Curve Diffie-Hellman (ECDH) key agreement

The Modular Arithmetic Operation of the TOE shall meet the requirement "Cryptographic operation(FCS_COP.1)" as specified below.

FCS_COP.1/ECDH 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/ECDH The TSF shall perform elliptic curve Diffie-Hellman key agreement in accordance with a specified cryptographic algorithm ECDH and cryptographic key sizes of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits that meet the following standard:
According to section 5.4.1 in ANSI X9.63 – 2001: Unlike section 5.4.1.3 our, implementation not only returns the x-coordinate of the shared secret, but rather the x-coordinate and y-coordinate.

Note:The certification covers the standard NIST [DSS] and Brainpool [ECC] Elliptic Curves with key lengths of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 Bits. Other types of elliptic curves can be added by the user during a composite certification process.

Note:This TOE can be delivered with the Crypto2304T coprocessor accessible or blocked. In case the Crypto2304T is blocked, no ECC computation supported by hardware is possible and this SFR is not applicable.

Note:The TOE can be delivered with an optional ECC library. Any optional ECC library contains the ECC algorithms stated above. If no optional ECC library is available then this SFR is not applicable.

FCS_COP.1/ECDH_PSL 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/ECDH_PSL The TSF shall perform elliptic curve Diffie-Hellman key agreement in accordance with a specified cryptographic algorithm ECDH and cryptographic key sizes of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 bits that meet the following standard: According to section 5.4.1 in ANSI X9.63 – 2001: Unlike section 5.4.1.3 our, implementation not only returns the x-coordinate of the shared secret, but rather the x-coordinate and y-coordinate.

Note:The certification covers the standard NIST [DSS] and Brainpool [ECC] Elliptic Curves with key lengths of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 Bits. Other types of elliptic curves can be added by the user during a composite certification process.

Note:For easy integration of EC functions into the user's operating system and/or application, the library contains single cryptographic functions respectively primitives which are compliant to the standard. The primitives are referenced above. Therefore, the library supports the user to develop an application representing the standard if required.

Note:This TOE can be delivered with the Crypto2304T coprocessor accessible or blocked. In case the Crypto2304T is blocked, no ECC computation supported by hardware is possible and this SFR is not applicable.

Note:The TOE can be delivered with an optional PSL library. Any PSL library contains a special interface to the algorithms stated above.If no optional PSL library is available then this SFR is not applicable.

7.3.7 Hash function

The TOE shall meet the requirement “Cryptographic operation – SHA (FCS_COP.1/SHA)” as specified below.

FCS_COP.1/SHA 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/ SHA The TSF shall perform hashing in accordance with a specified cryptographic algorithm SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 and cryptographic key sizes none that meet the following FIPS 180-4 [SHS].

Note:The TOE can be delivered with the optional HCL library. The optional HCL library contains the hash algorithms stated above. If the optional HCLlibrary is not delivered then this SFR is not applicable.

Note:This SFR claims countermeasures against SPA template attacks

Note:The SHA-1 algorithm shall only be used for session key derivation

FCS_COP.1/SHA_PSL 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/ SHA_PSL The TSF shall perform hashing in accordance with a specified cryptographic algorithm SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 and cryptographic key sizes none that meet the following FIPS 180-4 [SHS].

Note:The TOE can be delivered with the optional PSL library v5.00.06. If the optional PSL library v5.00.06 is not available then this SFR is not applicable.

Note:The SHA-1 algorithm shall only be used for session key derivation

7.4 Data Integrity

The TOE shall meet the requirement “Stored data integrity monitoring (FDP_SDI.1)” as specified below:

FDP_SDI.1 Stored data integrity monitoring

Hierarchical to: No other components

Dependencies: No dependencies

FDP_SDI.1.1 The TSF shall monitor user data stored in containers controlled by the TSF for inconsistencies between stored data and corresponding EDC on all objects, based on the following attributes: EDC value for RAM and ROM and ECC value for the SOLID FLASH™ NVM and verification of stored data in the SOLID FLASH™ NVM.

The TOE shall meet the requirement "Stored data integrity monitoring and action (FDP_SDI.2)" as specified below:

FDP_SDI.2 Stored data integrity monitoring and action

Hierarchical to: FDP_SDI.1 stored data integrity monitoring

Dependencies: No dependencies

FDP_SDI.2.1 The TSF shall monitor user data stored in containers controlled by the TSF for data integrity and one- and/or more-bit-errors on all objects, based on the following attributes: corresponding EDC value for RAM and ROM and error correction ECC for the SOLID FLASH™ NVM.

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall correct 1 bit errors in the SOLID FLASH™ NVM automatically and inform the user about more bit errors.

7.5 TOE Security Assurance Requirements

The evaluation assurance level is EAL5 augmented with ALC_DVS.2 and AVA_VAN.5. In the following table, the security assurance requirements are given. The augmentation of the assurance components compared to the Protection Profile [1] is expressed with bold letters.

Table 17 Assurance components

Aspect	Acronym	Description	Refinement
Development	ADV_ARC.1	Security Architecture Description	in PP [1]
	ADV_FSP.5	**Complete semiformal functional specification with additional error information**	in ST
	ADV_IMP.1	Implementation representation of the TSF	in PP [1]
	ADV_INT.2	**Well-structured internals**	
	ADV_TDS.4	**Semi-formal modular design**	
Guidance Documents	AGD_OPE.1	Operational user guidance	in PP [1]
	AGD_PRE.1	Preparative procedures	in PP [1]
Life-Cycle Support	ALC_CMC.4	Production support, acceptance procedures and automation	in PP [1]
	ALC_CMS.5	**Development tools CM coverage**	in ST

	ALC_DEL.1	Delivery procedures	in PP [1]
	ALC_DVS.2	Sufficiency of security measures	in PP [1]
	ALC_LCD.1	Developer defined life-cycle model	
	ALC_TAT.2	**Compliance with implementation standards**	in ST
Security Target Evaluation	ASE_CCL.1	Conformance claims	
	ASE_ECD.1	Extended components definition	
	ASE_INT.1	ST introduction	
	ASE_OBJ.2	Security objectives	
	ASE_REQ.2	Derived security requirements	
	ASE_SPD.1	Security problem definition	
	ASE_TSS.1	TOE summary specification	
Tests	ATE_COV.2	Analysis of coverage	in PP [1]
	ATE_DPT.3	**Testing: modular design**	in ST
	ATE_FUN.1	Functional testing	
	ATE_IND.2	Independent testing - sample	
Vulnerability Assessment	AVA_VAN.5	Advanced methodical vulnerability analysis	in PP [1]

7.5.1 Refinements

Some refinements are taken unchanged from the PP [1]. In some cases a clarification is necessary. In Table 19 an overview is given where the refinement is done.

Two refinements from the PP [1] have to be discussed here in the Security Target, as the assurance level is increased.

Life cycle support (ALC_CMS, ALC_TAT)

The refinement from the PP [1] can be applied even at the chosen assurance level EAL 5 augmented with ALC_CMS.5 and ALC_TAT.2. The assurance package ALC_CMS.4 is extended to ALC_CMS.5 with aspects regarding the configuration control system for the TOE. The assurance package ALC_TAT.1 is extended to ALC_TAT.2 with aspects regarding the implementation standards for the TOE. The refinements are not touched.

Functional Specification (ADV_FSP)

The refinement from the PP [1] can be applied even at the chosen assurance level EAL 5 augmented with ADV_FSP.5. The assurance package ADV_FSP.4 is extended to ADV_FSP.5 with aspects regarding the descriptive level. The level is increased from informal to semi-formal with informal description. The refinement is not touched from this measure.

For details of the refinement see PP [1].

Tests (ATE_DPT.3)

The refinement from the PP [1] can be applied even at the chosen assurance level EAL 5 augmented with ATE_DPT.3. The assurance package ATE_DPT.2 is augmented to ATE_DPT.3 relating to the requirements of the assurance level EAL 5. The refinement is not touched.

7.6 Security Requirements Rationale

7.6.1 Rationale for the Security Functional Requirements

The security functional requirements rationale of the TOE are defined and described in PP [1] section 6.3 for the following security functional requirements: FDP_ITT.1, FDP_IFC.1, FPT_ITT.1, FPT_PHP.3, FPT_FLS.1, FRU_FLT.2, FMT_LIM.1, FMT_LIM.2, FCS_RNG.1/HW, FCS_RNG.1/PSL and FAU_SAS.1.

The security functional requirements FPT_TST.2, FDP_ACC.1, FDP_ACF.1, FMT_MSA.1, FMT_MSA.3, FMT_SMF.1, FCS_COP.1, FCS_CKM.1, FDP_SDI.1 and FDP_SDI.2 are defined in the following description:

Table 18 Rational for additional SFR in the ST

Objective	TOE Security Functional Requirements
O.Add-Functions (optional)	FCS_COP.1/DES (optional) FCS_COP.1/DES_SCL_1 (optional) FCS_COP.1/DES_SCL_2 (optional) FCS_COP.1/DES_SCL_3 (optional) FCS_COP.1/DES_PSL (optional) FCS_COP.1/DES_MAC_PSL (optional) FCS_COP.1/AES (optional) FCS_COP.1/AES_SCL_1 (optional) FCS_COP.1/AES_SCL_2 (optional) FCS_COP.1/AES_SCL_3 (optional) FCS_COP.1/AES_PSL (optional) FCS_COP.1/AES_MAC_PSL_1 (optional) FCS_COP.1/AES_MAC_PSL_2 (optional) FCS_COP.1/RSA(optional) FCS_COP.1/RSA_PSL (optional) FCS_COP.1/ECDSA (optional) FCS_COP.1/ECDH (optional) FCS_COP.1/ECDH_PSL (optional) FCS_CKM.1/EC (optional) FCS_COP.1/SHA (optional) FCS_COP.1/SHA_PSL (optional)
O.Phys-Manipulation	FPT_TST.2
O.Mem-Access	FDP_ACC.1 FDP_ACF.1 FMT_MSA.3 FMT_MSA.1

	FMT_SMF.1
O.Malfunction	FDP_SDI.1 FDP_SDI.2

The table above gives an overview, how the security functional requirements are combined to meet the security objectives. The detailed justification is given in the following:

The justification related to the security objective “Additional Specific Security Functionality (O.Add-Functions)” is as follows:

The security functional requirement(s) “Cryptographic operation (FCS_COP.1)” exactly requires those functions to be implemented which are demanded by O.Add-Functions. FCS_CKM.1/EC supports the generation of EC keys needed for this cryptographic operations. Therefore, FCS_COP.1/RSA, FCS_COP.1/RSA_PSL, FCS_COP.1/ECDSA, FCS_COP.1/ECDH, FCS_COP.1/ECDH_PSL and FCS_CKM/EC are suitable to meet the security objective. The use of the supporting Base library has no impact on any security functional requirement nor does its use generate additional requirements.
Nevertheless, the developer of the Smartcard Embedded Software must ensure that the additional functions are used as specified and that the User Data processed by these functions are protected as defined for the application context. These issues are addressed by the specific security functional requirements:

- [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.

All these requirements have to be fulfilled to support OE.Resp-Appl for FCS_COP.1/DES, FCS_COP.1/DES_SCL_1, FCS_COP.1/DES_SCL_2, FCS_COP.1/DES_SCL_3,FCS_COP.1/DES_PSL, FCS_COP.1/DES_MAC_PSL and for FCS_COP.1/AES, FCS_COP.1/AES_SCL_1, FCS_COP.1/AES_SCL_2, FCS_COP.1/AES_SCL_3, FCS_COP.1, FCS_COP.1/AES_PSL, FCS_COP.1/AES_MAC_PSL_1, FCS_COP.1/AES_MAC_PSL_2. For the FCS_COP.1/RSA, FCS_COP.1/RSA _PSL and FCS_COP.1/ECDSA , and FCS_COP.1/ECDH, FCS_COP.1/ECDH_PSL and the FCS_CKM.1/EC are optional, since they are fulfilled by the TOE or may be fulfilled by the environment as the user can generate keys externally additionally.

The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced define how to implement the specific security functionality. However, key-dependent functions could be implemented in the Smartcard Embedded Software.

The usage of cryptographic algorithms requires the use of appropriate keys. Otherwise these cryptographic functions do not provide security. The keys have to be unique with a very high probability, and must have a certain cryptographic strength etc. In case of a key import into the TOE (which is usually after TOE delivery) it has to be ensured that quality and confidentiality are maintained. Keys for 3DES and AES are provided by the environment, the keys for RSA and EC algorithms can be provided either by the TOE or the environment.

In this ST the objectives for the environment OE.Plat-Appl and OE.Resp-Appl have been clarified. The Smartcard Embedded Software defines the use of the cryptographic functions FCS_COP.1 provided by the TOE. The requirements for the environment FDP_ITC.1, FDP_ITC.2, FCS_CKM.1 and FCS_CKM.4 support an appropriate key management. These security requirements are suitable to meet OE.Resp-Appl.

The justification of the security objective and the additional requirements (both for the TOE and its environment) show that they do not contradict to the rationale already given in the Protection Profile for the assumptions, policy and threats defined there.

The security functional component Subset TOE security testing (FPT_TST.2) has been newly created (Common Criteria Part 2 extended). This component allows that particular parts of the security mechanisms and functions provided by the TOE can be tested after TOE Delivery. This security functional component is used instead of the functional component FPT_TST.1 from Common Criteria Part 2. For the user it is important to know which security functions or mechanisms can be tested. The functional component FPT_TST.1 does not mandate to explicitly specify the security functions being tested. In addition, FPT_TST.1 requires verification of the integrity of TSF data and stored TSF executable code which might violate the security policy.

The tested security enforcing functions are SF_DPM Device Phase Management and SF_PMA Protection against modifying attacks.

The security functional requirement FPT_TST.2 will detect attempts to conduce a physical manipulation on the monitoring functions of the TOE. The objective of FPT_TST.2 is O.Phys-Manipulation. The physical manipulation will be tried to overcome security enforcing functions.

The security functional requirement “Subset access control (FDP_ACC.1)” with the related Security Function Policy (SFP) “Memory Access Control Policy” exactly require the implementation of an area based memory access control as required by O.Mem-Access. The related TOE security functional requirements FDP_ACC.1, FDP_ACF.1, FMT_MSA.3, FMT_MSA.1 and FMT_SMF.1 cover this security objective. The implementation of these functional requirements is represented by the dedicated privilege level concept.

The justification of the security objective and the additional requirements show that they do not contradict to the rationale already given in the Protection Profile for the assumptions, policy and threats defined there. Moreover, these additional security functional requirements cover the requirements by [3] user data protection of chapter 11 which are not refined by the PP [1].

Nevertheless, the developer of the Smartcard Embedded Software must ensure that the additional functions are used as specified and that the User Data processed by these functions are protected as defined for the application context. The TOE only provides the tool to implement the policy defined in the context of the application.

The justification related to the security objective “Protection against Malfunction due to Environmental Stress (O.Malfunction)” is as follows:

The security functional requirement “Stored data integrity monitoring (FDP_SDI.1)” requires the implementation of an Error Detection (EDC) algorithm which detects integrity errors of the data stored in RAM, ROM and SOLID FLASH™ NVM (in the SOLID FLASH™ NVM more bit errors are detected). By this the malfunction of the TOE using corrupt data is prevented. Therefore FDP_SDI.1 is suitable to meet the security objective.

The security functional requirement “Stored data integrity monitoring and action (FDP_SDI.2)” requires the implementation of an integrity observation and correction which is implemented by the Error Detection (EDC) and Error Correction (ECC) measures. The EDC is present in RAM and ROM of the TOE while the ECC is realized in the SOLID FLASH™ NVM. These measures detect and inform about one and more bit errors. In case of the SOLID FLASH™ NVM 1 bit errors of the data are corrected automatically. By the ECC mechanisms it is prevented that the TOE uses corrupt data. Therefore FDP_SDI.2 is suitable to meet the security objective.

The CC part 2 defines the component FIA_SOS.2, which is similar to FCS_RNG.1, as follows:

FIA_SOS.2 TSF Generation of secrets

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_SOS.2.1 The TSF shall provide a mechanism to generate secrets that meet [assignment: *defined quality metric*].

FIA_SOS.2.2 The TSF shall be able to enforce the use of TSF generated secrets for [assignment: *list of TSF functions*].

The CC part 2, annex G.3 [3], states: "This family defines requirements for mechanisms that enforce defined quality metrics on provided secrets, and generate secrets to satisfy the defined metric". Even the operation in the element FIA_SOS.2.2 allows listing the TSF functions using the generated secrets. Because all applications discussed in annex G.3 are related to authentication, the component FIA_SOS.2 is also intended for authentication purposes while the term "secret" is not limited to authentication data (cf. CC part 2, paragraphs 39-42). Paragraph 685 in the CC part 2 [3] recommends to use the component FCS_CKM.1 to address random number generation. However, this may hide the nature of the secrets used for key generation and does not allow describing random number generation for other cryptographic methods (e.g., challenges, padding), authentication (e.g., password seeds), or other purposes (e.g., blinding as a countermeasure against side channel attacks). The component FCS_RNG addresses general RNG, the use of which includes but is not limited to cryptographic mechanisms. FCS_RNG allows specifying requirements for the generation of random numbers including necessary information for the intended use. These details describe the quality of the generated data where other security services rely on. Thus by using FCS_RNG a ST or PP author is able to express a coherent set of SFRs that include or use the generation of random numbers as a security service.

7.6.1.1 Dependencies of Security Functional Requirements

The dependence of security functional requirements are defined and described in PP [1] section 6.3.2 for the following security functional requirements: FDP_ITT.1, FDP_IFC.1, FPT_ITT.1, FPT_PHP.3, FPT_FLS.1, FRU_FLT.2, FMT_LIM.1, FMT_LIM.2, FCS_RNG.1/HW, FCS_RNG.1/PSL and FAU_SAS.1.

The dependence of security functional requirements for the security functional requirements FPT_TST.2, FDP_ACC.1, FDP_ACF.1, FMT_MSA.1, FMT_MSA.3, FMT_SMF.1, FCS_COP.1, FCS_CKM.1, FDP_SDI.1 and FDP_SDI.2 are defined in the following description.

Table 19 Dependency for cryptographic operation requirement

Security Functional Requirement	Dependencies	Fulfilled by security requirements
FCS_COP.1/DES FCS_COP.1/DES_SCL_1 FCS_COP.1/DES_SCL_2 FCS_COP.1/DES_SCL_3 FCS_COP.1/DES_PSL FCS_COP.1/DES_MAC_PSL	FCS_CKM.1	Yes, see comment
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) FCS_CKM.4	Yes, see comment

FCS_COP.1/AES FCS_COP.1/AES_SCL_1 FCS_COP.1/AES_SCL_2 FCS_COP.1/AES_SCL_3 FCS_COP.1/AES_PSL FCS_COP.1/AES_MAC_PSL_1 FCS_COP.1/AES_MAC_PSL_2	FCS_CKM.1	Yes, see comment
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) FCS_CKM.4	Yes, see comment
FCS_COP.1/RSA FCS_COP.1/RSA_PSL	FCS_CKM.1	Yes, see comment
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) FCS_CKM.4	Yes, see comment
FCS_COP.1/ECDSA	FCS_CKM.1	Yes, see comment
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) FCS_CKM.4	Yes, see comment
FCS_CKM.1/EC	FCS_CKM.2 or FCS_COP.1	Yes
	FCS_CKM.4	Yes, see comment
FCS_COP.1/ECDH FCS_COP.1/ECDH_PSL	FCS_CKM.1	Yes, see comment
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) FCS_CKM.4	Yes, see comment
FCS_COP.1/SHA FCS_COP.1/SHA_PSL	FCS_CKM.1, FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1), FCS_CKM.4	Not required, see comment
FPT_TST.2	None	See comment
FDP_ACC.1	FDP_ACF.1	Yes
FDP_ACF.1	FDP_ACC.1 FMT_MSA.3	Yes Yes
FMT_MSA.3	FMT_MSA.1 FMT_SMR.1	Yes Not required, see comment
FMT_MSA.1	FDP_ACC.1 or FDP_IFC.1 FMT_SMR.1 FMT_SMF.1	Yes See comment Yes
FMT_SMF.1	None	N/A
FDP_SDI.1	None	N/A
FDP_SDI.2	None	N/A

Note:The dependency FMT_SMR.1 introduced by the two components FMT_MSA.1 and FMT_MSA.3 is considered to be satisfied because the access control specified for the intended TOE is not role-based but enforced for each subject. Therefore, there is no need to identify roles in form of a security functional requirement FMT_SMR.1.

Comment: The security functional requirement "Cryptographic operation (FCS_COP.1)" met by the TOE, has the following dependencies:

- [FDP_ITC.1 Import of user data without security attributes, or

- FDP_ITC.2 Import of user data with security attributes]
- FCS_CKM.1 Cryptographic key generation
- FCS_CKM.4 Cryptographic key destruction.

The security functional requirement “Cryptographic key management (FCS_CKM)” met by TOE, has the following dependencies:

- [FCS_CKM.2 Cryptographic key distribution, or
- FCS_COP.1 Cryptographic operation]
- FCS_CKM.4 Cryptographic key destruction.

These requirements all address the appropriate management of cryptographic keys used by the specified cryptographic function and are not part of the PP [1]. Most requirements concerning key management shall be fulfilled by the environment since the Smartcard Embedded Software is designed for a specific application context and uses the cryptographic functions provided by the TOE.
For the security functional requirement FCS_COP.1/DES, FCS_COP.1/DES_SCL_1, FCS_COP.1/DES_SCL_2, FCS_COP.1/DES_SCL_3,FCS_COP.1/DES_PSL, FCS_COP.1/DES_MAC_PSL and FCS_COP.1/AES, FCS_COP.1/AES_SCL_1, FCS_COP.1/AES_SCL_2, FCS_COP.1/AES_SCL_3, FCS_COP.1/AES_PSL, FCS_COP.1/AES_MAC_PSL_1, FCS_COP.1/AES_MAC_PSL_2 the respective dependencies FCS_CKM.1, FCS_CKM.4 and FDP_ITC.1 or FDP_ITC.2 have to be fulfilled by the environment. That mean, that the environment shall meet the requirements FCS_CKM.1 and FCS_CKM.4 as defined in [3], section 10.1 and shall meet the requirements FDP_ITC.1 or FDP_ITC.2 as defined in [3], section 11.7.

For the security functional requirement FCS_COP.1/RSA, FCS_COP.1/RSA_PSL, FCS_COP.1/ECDSA, and FCS_COP.1/ECDH, FCS_COP.1/ECDH_PSL, the respective dependencies FCS_CKM.4 and FDP_ITC.1 or FDP_ITC.2 have to be fulfilled by the environment. That mean, that the environment shall meet the requirements FDP_ITC.1 or FDP_ITC.2 as defined in [3], section 11.7.

For the security functional requirement FCS_COP.1/RSA, FCS_COP.1/RSA_PSL, FCS_COP.1/ECDSA, and FCS_COP.1/ECDH, the respective dependency FCS_CKM.1 has to be fulfilled by the TOE with the security functional requirement FCS_CKM.1/EC (for FCS_COP.1/ECDSA and FCS_COP.1/ECDH) as defined in section 7.1.4. Additionally the requirement FCS_CKM.1 can be fulfilled by the environment as defined in [3], section 10.1.

For the security functional requirement FCS_COP.1/RSA , FCS_CKM.1 has to be fulfilled by the environment.

For the security functional requirement FCS_COP.1/ECDH_PSL, the respective dependency FCS_CKM.1 does not apply, because the PSL does not provide a key generation operation for elliptic curves.

For the security functional requirement FCS_CKM.1/EC the respective dependency FCS_COP.1 is fulfilled by the TOE. The respective dependency FCS_CKM.4 has to be fulfilled by the environment. That means, the environment shall meet the requirement FCS_CKM.4 as defined in [3], section 10.1.

For the security functional requirement FCS_COP.1/SHA and FCS_COP.1/SHA_PSL the respective dependencies are not applicable, because no keys are involved.

The cryptographic libraries RSA and EC are delivery options. If one of the libraries RSA, EC are delivered, the asymmetric Base Lib is automatically part of it. Therefore the user may choose a free combination of these libraries. In case of deselecting one or several of these libraries the TOE does not provide the respective functionality Additional Specific Security Functionality Rivest-Shamir-Adleman Cryptography (RSA) and/or Elliptic Curve Cryptography (EC). The asymmetric Base Library is no directly accessible cryptographic library and provides no additional specific security functionality.

End of comment.

7.6.2 Rationale of the Assurance Requirements

The chosen assurance level EAL5 and the augmentation with the requirements ALC_DVS.2 and AVA_VAN.5 were chosen in order to meet the assurance expectations explained in the following paragraphs. In Table 17 the different assurance levels are shown as well as the augmentations. The augmentations are in compliance with the Protection Profile.

An assurance level EAL5 with the augmentations ALC_DVS.2 and AVA_VAN.5 are required for this type of TOE since it is intended to defend against **highly sophisticated attacks** without protective environment. This evaluation assurance package was selected to permit a developer to gain maximum assurance from positive security engineering based on good commercial practices. In order to provide a meaningful level of assurance that the TOE provides an adequate level of defence against such attacks, the evaluators should have access to all information regarding the TOE including the TSF internals, the low level design and source code including the testing of the modular design. Additionally the mandatory technical document “Application of Attack Potential to Smartcards” [10] shall be taken as a basis for the vulnerability analysis of the TOE.

ALC_DVS.2 Sufficiency of security measures

Development security is concerned with physical, procedural, personnel and other technical measures that may be used in the development environment to protect the TOE.

In the particular case of a Security IC the TOE is developed and produced within a complex and distributed industrial process which must especially be protected. Details about the implementation, (e.g. from design, test and development tools as well as Initialization Data) may make such attacks easier. Therefore, in the case of a Security IC, maintaining the confidentiality of the design is very important.

This assurance component is a higher hierarchical component to EAL5 (which only requires ALC_DVS.1). ALC_DVS.2 has no dependencies.

AVA_VAN.5 Advanced methodical vulnerability analysis

Due to the intended use of the TOE, it must be shown to be highly resistant to penetration attacks. This assurance requirement is achieved by the AVA_VAN.5 component.

Independent vulnerability analysis is based on highly detailed technical information. The main intent of the evaluator analysis is to determine that the TOE is resistant to penetration attacks performed by an attacker possessing high attack potential.

AVA_VAN.5 has dependencies to ADV_ARC.1 “Security architecture description”, ADV_FSP.2 “Security enforcing functional specification”, ADV_TDS.3 “Basic modular design”, ADV_IMP.1 “Implementation representation of the TSF”, AGD_OPE.1 “Operational user guidance”, and AGD_PRE.1 “Preparative procedures”.

All these dependencies are satisfied by EAL5.

It has to be assumed that attackers with high attack potential try to attack Security ICs like smart cards used for digital signature applications or payment systems. Therefore, specifically AVA_VAN.5 was chosen in order to assure that even these attackers cannot successfully attack the TOE.

8 TOE Summary Specification (ASE_TSS)

The product overview is given in section 2.1. In the following the Security Features are described and the relation to the security functional requirements is shown.

The TOE is equipped with following Security Features to meet the security functional requirements:

SF_DPM	Device Phase Management
SF_PS	Protection against Snooping
SF_PMA	Protection against Modification Attacks
SF_PLA	Protection against Logical Attacks
SF_CS	Cryptographic Support

The following description of the Security Features is a complete representation of the TSF.

8.1 SF_DPM: Device Phase Management

The life cycle of the TOE is split-up in several phases. Chip development and production (phase 2, 3, 4) and final use (phase 4-7) is a rough split-up from TOE point of view. These phases are implemented in the TOE as test mode (phase 3) and user mode (phase 4-7).
In addition a chip identification mode exists which is active in all phases. The chip identification data (O.Identification) is stored in a in the not changeable configuration page area and non-volatile memory. In the same area further TOE configuration data is stored. In addition, user initialization data can be stored in the non-volatile memory during the production phase as well. During this first data programming, the TOE is still in the secure environment and in Test Mode.
The covered security functional requirement is FAU_SAS.1 “Audit storage”.

During start-up of the TOE the decision for one of the operation modes is taken dependent on phase identifiers. The decision of accessing a certain mode is defined as phase entry protection. The phases follow also a defined and protected sequence. The sequence of the phases is protected by means of authentication.
The covered security functional requirements are FMT_LIM.1 “Limited capabilities” and FMT_LIM.2 “Limited availability”.

During the production phase (phase 3 and 4) or after the delivery to the customer (phase 5 or phase 6), the TOE provides the possibility to download a user specific encryption key and user code and data into the empty (erased) SOLID FLASH™ NVM memory area as specified by the associated control information of the Flash Loader software. After finishing the load operation, the Flash Loader can be permanently deactivated, so that no further load operation with the Flash Loader is possible. These procedures are defined as phase operation limitation.
The covered security functional requirement is FMT_LIM.2 “Limited availability”.

During operation within a phase the accesses to memories are granted by the MPU controlled access rights and related levels.
The covered security functional requirements are FDP_ACC.1 “Subset access control”, FDP_ACF.1 “Security attribute based access control” and FMT_MSA.1 “Management of security attributes”.

In addition, during each start-up of the TOE the address ranges and access rights are initialized by the Boot Software (BOS) with predefined values.
The covered security functional requirement is FMT_MSA.3 “Static attribute initialisation”.

The TOE clearly defines access rights and levels in conjunction with the appropriate key management in dependency of the firmware or software to be executed.
The covered security functional requirement is FMT_SMF.1 “Specification of Management functions”.

Each operation phase is protected by means of authentication and encryption. The covered security functional requirements are FPT_ITT.1 “Basic internal TSF data transfer protection” and FDP_IFC.1 “Subset information flow control”. If any comparison of the authentication code fails a direct security reset is performed. The covered security functional requirements is FPT_FLS.1 (”Failure with preservation of secure state”).

The **SF_DPM** “Device Phase Management” covers the security functional requirements FPT_FLS.1, FAU_SAS.1, FMT_LIM.1, FMT_LIM.2, FDP_ACC.1, FDP_ACF.1, FMT_MSA.1, FMT_MSA.3, FMT_SMF.1, FPT_ITT.1 and FDP_IFC.1.

8.2 SF_PS: Protection against Snooping

Several mechanisms protect the TOE against snooping the design or the user data during operation and even if it is out of operation (power down).

The entire design is kept in a non standard way to prevent attacks using standard analysis methods. Important parts of the chip are especially designed to counter leakage or side channel attacks like DPA/SPA or EMA/DEMA. Therefore, even the physical data gaining is difficult to perform, since timing and current consumption is independent of the processed data. In the design a number of components are automatically synthesized and mixed up to disguise an attacker and to make an analysis more difficult. The covered security functional requirement is FPT_PHP.3 “Resistance to physical attack”.

A further protective design method used is secure wiring. All security critical wires have been identified and protected by special routing measures against probing. Additionally the wires are embedded into shield lines and used as normal signal lines for operation of the chip to prevent successful probing. This measurement is called “security optimized wiring”. The covered security functional requirements are FPT_PHP.3 “Resistance to physical attack”, FPT_ITT.1 “Basic internal TSF data transfer protection”, FPT_FLS.1 “Failure with preservation of secure state” and FDP_ITT.1 “Basic internal transfer protection”.

All contents of the memories RAM, ROM and SOLID FLASH™ NVM of the TOE are encrypted on chip to protect them against data analysis. The external Flash-memory is not encrypted and not a part of the security functional requirements. In addition the data transferred over the memory bus to and from (bi-directional encryption) the CPU, Co-processor (Crypto2304T and SCP), the special SFRs and the peripheral devices (CRC, RNG and Timer) are transported encrypted with an automatically dynamic key change. The encryption of the memory content is done by the MED using a proprietary cryptographic algorithm and a complex key management providing protection against cryptographic analysis attacks. This means that the SOLID FLASH™ NVM, RAM, ROM and the bus are encrypted with module dedicated and dynamic keys. The only key remaining static over the product life cycle is the specific ROM key changing from mask to mask. All security relevant transfer of addresses or data via the peripheral bus is dynamically masked and thus protected against readout and analysis. The function Trash Register Writes can be activated by the user to hide the fact if a register has been written. The covered security functional requirements are FDP_IFC.1 “Subset information flow control“, FPT_PHP.3 “Resistance to physical attack”, FPT_ITT.1 “Basic internal TSF data transfer protection, FPT_FLS.1 “Failure with preservation of secure state” and FDP_ITT.1 “Basic internal transfer protection”.

The **SF_PS** “Protection against Snooping” covers the security functional requirements FPT_PHP.3, FDP_IFC.1, FPT_ITT.1, FPT_FLS.1 and FDP_ITT.1.

8.3 SF_PMA: Protection against Modifying Attacks

The TOE is equipped with an error detection code (EDC) for protecting RAM and ROM and an ECC, which is realized in the SOLID FLASH™ NVM. Thus introduced failures are securely detected and, in terms of single bit errors in the SOLID FLASH™ NVM also automatically corrected (FDP_SDI.2). For SOLID FLASH™ NVM in case of more than one bit errors and for RAM in case of any bit errors detected, a security alarm is triggered.
In order to prevent accidental bit faults during production in the ROM, over the data stored in ROM an EDC value is calculated (FDP_SDI.1).

The covered security functional requirements are FRU_FLT.2 "Limited fault tolerance", FDP_PHP.3 "Resistance to physical attack", FDP_SDI.1 "Stored data integrity monitoring" and FDP_SDI.2 "Stored data integrity monitoring and action".

If a user tears the card resulting in a power off situation during an SOLID FLASH™ NVM programming operation or if other perturbation is applied, no data or content loss occurs and the TOE restarts power on. The NVM tearing save write functionality covers FDP_SDI.1 "Stored data integrity monitoring" as the new data to be programmed are checked for integrity and correct programming before the page with the old data becomes valid.

The covered security functional requirement are FPT_PHP.3 "Resistance to physical attack", since these measures make it difficult to manipulate the write process of the NVM, FPT_FLS.1 "Failure with preservation of secure state"and FDP_SDI.1 "Stored data integrity monitoring".

In the case that a physical manipulation or a physical probing attack is detected, the processing of the TOE is immediately stopped and the TOE enters a secure state called security reset.

A shielding algorithm finishes the upper layers above security critical signals and wires, finally providing the so called "security optimized wiring".

The covered security functional requirements are FPT_FLS.1 "Failure with preservation of secure state", FPT_PHP.3 "Resistance to physical attack" and FPT_TST.2 "Subset TOE security testing".

As physical effects or manipulative attacks may also address the program flow of the user software, two watchdog timers each with a check point register function are implemented. This feature allows the user to check the correct processing time and the integrity of the program flow of the user software.
The Instruction Stream Signature Checking (ISS) calculates a hash about all executed instructions and automatically checks the correctness of this hash value. If the code execution follows an illegal path an alarm is triggered.
Another measure against modifying and perturbation respectively differential fault attacks (DFA) is the implementation of backward calculation in the SCP. By this induced errors are discovered.

The covered security functional requirements are FPT_FLS.1 "Failure with preservation of secure state", FDP_IFC.1 "Subset information flow control", FPT_ITT.1 "Basic internal transfer protection", FDP_ITT.1 "Basic internal transfer protection" and FPT_PHP.3 "Resistance to physical attack".

During start up, the TOE performs various configurations and subsystem tests. After the TOE startup has finished, the operating system or application can call the User Mode Security Life Control (UMSLC) test provided by the Resource Management System. The UMSLC checks the alarm lines and/or the different security functions and sensors for correct operation. The test can be triggered by user software during normal operation. As attempts to modify the security features will be detected from the test, the covered security functional requirement is FPT_TST.2 "Subset TOE security testing".

The correct function of the TOE is only given in the specified range of the environmental operating parameters. To prevent an attack exploiting that circumstance the TOE is equipped with a temperature sensor, glitch sensor and backside light detection. The TOE falls into the defined secure state in case of a specified range violation. The defined secure state causes the chip internal reset process. Note that the specified range checking can only work when the TOE is running and can not prevent reverse engineering.
The covered security functional requirements are FRU_FLT.2 "Limited fault tolerance" and FPT_FLS.1 "Failure with preservation of secure state".

The **SF_PMA** "Protection against Modifying Attacks" covers the security functional requirements FPT_PHP.3, FDP_IFC.1, FPT_ITT.1, FDP_ITT.1, FPT_TST.2, FDP_SDI.1, FDP_SDI.2, FRU_FLT.2 and FPT_FLS.1.

8.4 SF_PLA: Protection against Logical Attacks

The memory model of the TOE provides two distinct, independent levels called the privileged and non-privilege level and the possibility to define up to eight memory regions with different access rights enforced by the Management Protection Unit (MPU). This gives the user software the possibility to define different access rights for the regions 0 to 7 for privilege or non-privilege level. In the case of an access violation the MPU will trigger a trap. The policy of setting up the MPU and specifying the memory ranges for the regions (0 to 7) is defined from the user software.
The covered security functional requirements are FDP_ACC.1 "Subset access control", FDP_ACF.1 "Security attribute based access control", FMT_MSA.1 "Management of security attributes", FMT_MSA.3 "Static attribute initialisation" and FMT_SMF.1 "Specification of Management functions".

All memories present on the TOE (NVM, ROM, RAM) are encrypted using individual keys assigned by complex key management. In case of security critical error a security alarm is generated and the TOE ends up in a secure state.
The covered security functional requirements are FDP_ACF.1 "Security attribute based access control" and FPT_FLS.1 "Failure with preservation of secure state".

The **SF_PLA** "Protection against Logical Attacks" covers the security functional requirements FDP_ACC.1, FDP_ACF.1, FMT_MSA.1, FMT_MSA.3, FPT_FLS.1 and FMT_SMF.1.

8.5 SF_CS: Cryptographic Support

The TOE is equipped an asymmetric and a symmetric hardware accelerators to support the standard symmetric and asymmetric cryptographic operations. This security function is introduced to include the cryptographic operation in the scope of the evaluation as the cryptographic function respectively mathematic algorithm itself is not used from the TOE security policy. The components are a co-processor supporting the DES and AES algorithms and a co-processor and software modules to support RSA cryptography, EC signature generation and verification, ECDH key agreement and EC public key calculation and testing. Additionally the TOE is equipped with a True Random Number Generator for the generation of random numbers.

8.5.1 3DES encryption

The TOE supports the encryption and decryption in accordance with the specified cryptographic algorithm Triple Data Encryption Standard (3DES) in the Electronic Codebook Mode (ECB), Cipher Block Chaining Mode (CBC), Cipher Feedback Mode (CFB), Counter Mode (CTR) and CMAC mode and with cryptographic key sizes of 112 and 168 bit meeting the standard: [N867], [N38A], [N38B].

The covered security functional requirements are FCS_COP.1/DES, FCS_COP.1/DES_SCL_1, FCS_COP.1/DES_SCL_2, FCS_COP.1/DES_SCL_3, and FCS_COP.1/DES_ PSL

This SFR is implemented in 3 ways:

1. By directly programming the hardware registers of the symmetric coprocessor.
2. By using the interface of the optional SCL. This library contains additional countermeasures.
3. By using the interface of the optional PSL. This library uses the SCL library to access the symmetric coprocessor.

8.5.2 3DES MAC

The TSF supports MAC calculation with the cryptographic algorithm Triple Data Encryption Standard (3DES) in CBC MAC mode and cryptographic key sizes of 2 x 56 or 3 x 56 bit according to the standards: [N867], [9797] with the following options/modifications:

- MAC algorithm 1
- Padding must be done by the caller
- An Initialization Vector (IV) must be given by the caller
- The covered security functional requirements are FCS_COP.1/DES_MAC_PSL

8.5.3 AES encryption

The TSF supports the encryption and decryption in accordance with the specified cryptographic algorithm Advanced Encryption Standard (AES)) in the Electronic Codebook Mode (ECB), Cipher Block Chaining Mode (CBC), Cipher Feedback Mode (CFB), CTR (Counter) Mode and CMAC mode and cryptographic key sizes of 128 bit or 192 bit or 256 bit according to the standard: [N197], [N38A], [N38B].

The covered security functional requirement is FCS_COP.1/AES, FCS_COP.1/AES_SCL_1, FCS_COP.1/AES_SCL_2, FCS_COP.1/AES_SCL_3,FCS_COP.1/AES_PSL.

This TSF is implemented in 3 ways:

1. By directly programming the hardware registers of the symmetric coprocessor.
2. By using the interface of the optional SCL. This library contains additional countermeasures.
3. By using the interface of the optional PSL. This library uses the SCL library to access the symmetric coprocessor.

8.5.4 AES MAC

The TSF supports MAC calculation with the cryptographic algorithm Advanced Encryption Standard (AES) in CBC MAC mode and CMAC mode and cryptographic key sizes of 128 bit or 192 bit or 256 bit according to the standards: [N197],[N38B], [9797] with the following options/modifications:

- MAC algorithm 1
- Padding must be done by the caller
- An Initialization Vector (IV) must be given by the caller
- The covered security functional requirements are FCS_COP.1/AES_MAC_PSL_1, FCS_COP.1/AES_MAC_PSL_2

8.5.5 RSA

8.5.5.1 Encryption, Decryption, Signature Generation and Verification

The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithmRivest-Shamir-Adleman (RSA) and cryptographic key sizes 1024 - 4096 bits that meet the following standards:

Encryption:
According to section 5.1.1 RSAEP in PKCS v2.2, without 5.1.1(1).

Decryption (with or without CRT):
According to section 5.1.2 RSADP in PKCS v2.2
for u = 2, i.e., without any (r_i, d_i, t_i), i >2, therefore without 5.1.2(2.b) (ii)&(v), without 5.1.2(1),

5.1.2(2.a) only supported up to n < 2^{2048}.

Signature Generation (with or without CRT):
According to section 5.2.1 RSASP1 in PKCS v2.2
for u = 2, i.e., without any (r_i, d_i, t_i), i >2,
therefore without 5.2.1(2.b) (ii)&(v), without 5.2.1(1),
5.2.1(2.a) only supported up to n < 2^{2048}.

Signature Verification:
According to section 5.2.2 RSAVP1 in PKCS v2.2,
without 5.2.2(1).

The covered security functional requirement is FCS_COP.1/RSA, FCS_COP.1/RSA_PSL.

8.5.6 Elliptic Curves

The certification covers the standard NIST [DSS] and Brainpool [ECC] Elliptic Curves with key lengths of 160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521 Bits. Note that there exists numerous other curve types, being also secure in terms of side channel attacks on this TOE, which can the user optionally add in the composition certification process.

8.5.6.1 Signature Generation and Verification

The TSF shall perform signature generation and signature verification in accordance with a specified cryptographic algorithm ECDSA and cryptographic key sizes *160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521* bits that meet the following standard:

Signature Generation:

3. According to section 7.3 in ANSI X9.62 – 2005:
   Not implemented is step d) and e) thereof.
   The output of step e) has to be provided as input to our function by the caller.
   Deviation of step c) and f):
   The jumps to step a) were substituted by a return of the function with an error code, the jumps are emulated by another call to our function.

Signature Verification:

4. According to section 7.4.1 in ANSI X9.62–2005:
   Not implemented is step b) and c) thereof.

The output of step c) has to be provided as input to our function by the caller.
Deviation of step d):
Beside noted calculation, our algorithm adds a random multiple of the group order n to the calculated values u1 and u2.

The covered security functional requirement is FCS_COP.1/ECDSA.

8.5.6.2 Asymmetric Key Generation

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm Elliptic Curve EC specified in ANSI X9.62-1998 and specified cryptographic key sizes *160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521* bits that meet the following standard:

ECDSA Key Generation:

5. According to the appendix A4.3 in ANSI X9.62-2005 the cofactor h is not supported.

The covered security functional requirement is FCS_CKM.1/EC.

8.5.6.3 Asymmetric Key Agreement

The TSF shall perform elliptic curve Diffie-Hellman key agreement in accordance with a specified cryptographic algorithm ECDH and cryptographic key sizes *160, 163, 192, 224, 233, 256, 283, 320, 384, 409, 512 or 521* bits that meet the following standard:

6. According to section 5.4.1 in ANSI X9.63 -2001 Unlike section 5.4.1.3 our implementation not only returns the x-coordinate of the shared secret, but rather the x-coordinate and y-coordinate.
7.

The covered security functional requirement is FCS_COP.1/ECDH, FCS_COP.1/ECDH_PSL.

8.5.7 Asymmetric Base Library

The asymmetric Base library provides the low level interface to the asymmetric cryptographic coprocessor and has no user available interface. The asymmetric Base library does not provide any security functionality, implements no security mechanism, and does not provide additional specific security functionality. The asymmetric Base library does not cover security functional requirements.

8.5.8 Symmetric Crypto Library (SCL)

The symmetric crypto Library provides an interface to the SCP for AES and 3DES operations. The SCL contains additional software countermeasures to harden the restance against side channel and fault attacks. The SCL consists of three files “AES.lib”, “DES.lib” and “cipher.lib”. Those library files will only distributes together.
The covered security functional requirements are FCS_COP.1/DES_SCL_1, FCS_COP.1/DES_SCL_2, FCS_COP.1/DES_SCL_3, FCS_COP.1/AES_SCL_1, FCS_COP.1/AES_SCL_2, FCS_COP.1/AES_SCL_3.

8.5.1 Hash Crypto Library (HCL)

The hash crypto Library provides an interface to SHA-1 and SHA-2 hash operations. The HCL contains additional software countermeasures to harden the restance against single side

channel template attacks. The HCL consists of the files “HCL97-CPU-L90-hash.lib” and “HCL97-CPU-L90-sha.lib”
The covered security functional requirements are FCS_COP.1/SHA.

8.5.2 Platform Support Layer (PSL)

The Platform Support Layer (PSL) library is used to provide a standardized interface to the hardware, directly or via the RSA, ECC and SCL lib
rary. The provided interfaces are syntactically similar to Windows NT device driver calls. The PSL provides as additional cryptographic operations a MAC calculation with AES and 3DES keys.
The covered security functional requirements are FCS_COP.1/DES_PSL, FCS_COP.1/DES_MAC_PSL, FCS_COP.1/AES_PSL, FCS_COP.1/AES_MAC_PSL_1, FCS_COP.1/AES_MAC_PSL_2, FCS_COP.1/RSA_PSL, FCS_COP.1/ECDH_PSL, FCS_COP.1/SHA_PSL, FCS_RNG.1/PSL.

8.5.3 TRNG

Random data is essential for cryptography as well as for security mechanisms. The TOE is equipped with a physical True Random Number Generator (TRNG, FCS_RNG.1/HW and FCS_RNG.1/PSL). The random data can be used from the Smartcard Embedded Software and is also used from the security features of the TOE, like masking. The TRNG implements also self testing features. The TRNG fulfils the requirements from the functionality class PTG.2 of [6].
The covered security functional requirement is FCS_RNG.1/HW and FCS_RNG.1/PSL “Quality metric for random numbers”, FPT_PHP.3 “Resistance to physical attack”, FDP_ITT.1 “Basic internal transfer protection”, FPT_ITT.1 “Basic internal TSF data transfer protection, FDP_IFC.1 “Subset information flow control“, FPT_TST.2 “Subset TOE security testing“ and FPT_FLS.1“Failure with preservation of secure state”.

The **SF_CS** “Cryptographic Support” covers the security functional requirements FCS_COP.1/DES, FCS_COP.1/DES_SCL_1, FCS_COP.1/DES_SCL_2, FCS_COP.1/DES_SCL_3, FCS_COP.1/DES_PSL, FCS_COP.1/DES_MAC_PSL, FCS_COP.1/AES, FCS_COP.1/AES_SCL_1, FCS_COP.1/AES_SCL_2, FCS_COP.1/AES_SCL_3, FCS_COP.1/AES_PSL, FCS_COP.1/AES_MAC_PSL_1, FCS_COP.1/AES_MAC_PSL_2,FCS_COP.1/RSA, FCS_COP.1/RSA_PSL, FCS_COP.1/ECDSA, FCS_COP.1/ECDH, FCS_COP.1/ECDH_PSL, FCS_CKM.1/EC, FPT_PHP.3, FDP_ITT.1, FPT_ITT.1, FPT_FLS.1 ,FCS_RNG.1/HW and FCS_RNG.1/PSL, FDP_IFC.1.

8.6 Assignment of Security Functional Requirements to TOE’s Security Functionality

The justification and overview of the mapping between security functional requirements (SFR) and the TOE’s security functionality (SF) is given in sections the sections above. The results are shown in Table 20. The security functional requirements are addressed by at least one relating security feature.
The various functional requirements are often covered manifold. As described above the requirements ensure that the TOE is checked for correct operating conditions and if a not correctable failure occurs that a stored secure state is achieved, accompanied by data integrity monitoring and actions to maintain the integrity although failures occurred. An overview is given in following table:

Table 20 Mapping of SFR and SF

SFR	SF_DPM	SF_PS	SF_PMA	SF_PLA	SF_CS
FAU_SAS.1	X				
FMT_LIM.1	X				

FMT_LIM.2	X				
FDP_ACC.1	X			X	
FDP_ACF.1	X			X	
FPT_PHP.3		X	X		X
FDP_ITT.1		X	X		X
FDP_SDI.1			X		
FDP_SDI.2			X		
FDP_IFC.1	X	X	X		X
FMT_MSA.1	X			X	
FMT_MSA.3	X			X	
FMT_SMF.1	X			X	
FRU_FLT.2			X		
FPT_ITT.1	X	X	X		X
FPT_TST.2			X		
FPT_FLS.1	X	X	X	X	X
FCS_RNG.1/HW					X
FCS_RNG.1/PSL					X
FCS_COP.1/DES					X
FCS_COP.1/DES_SCL_1					X
FCS_COP.1/DES_SCL_2					X
FCS_COP.1/DES_SCL_3					X
FCS_COP.1/DES_PSL					X
FCS_COP.1/DES_MAC_PSL					X
FCS_COP.1/AES					X
FCS_COP.1/AES_SCL_1					X
FCS_COP.1/AES_SCL_2					X
FCS_COP.1/AES_SCL_3					X
FCS_COP.1/AES_PSL					X
FCS_COP.1/AES_MAC_PSL_1					X
FCS_COP.1/AES_MAC_PSL_2					X
FCS_COP.1/RSA					X
FCS_COP.1/RSA_PSL					X
FCS_COP.1/ ECDSA					X
FCS_COP.1/ECDH					X
FCS_COP.1/ECDH_PSL					X
FCS_COP.1/SHA					X
FCS_COP.1/SHA_PSL					X
FCS_CKM.1/EC					X

8.7 Security Requirements are internally Consistent

For this chapter the PP [1] section 6.3.4 can be applied completely.

In addition to the discussion in section 6.3 of PP [1] the security functional requirement FCS_COP.1 is introduced. The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced also protect the cryptographic algorithms implemented according to the security functional requirement FCS_COP.1. Therefore, these security functional requirements support the secure implementation and operation of FCS_COP.1.

As disturbing, manipulating during or forcing the results of the test checking the security functions after TOE delivery, this security functional requirement FPT_TST.2 has to be protected. An attacker could aim to switch off or disturb certain sensors or filters and preserve the detection of his manipulation by blocking the correct operation of FPT_TST.2. The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced also protect the security functional requirement FPT_TST.2. Therefore, the related security functional requirements support the secure implementation and operation of FPT_TST.2.

The requirement FPT_TST.2 allows testing of some security mechanisms by the Smartcard Embedded Software after delivery. In addition, the TOE provides an automated continuous user transparent testing of certain functions.

The implemented level concept represents the area based memory access protection enforced by the MPU. As an attacker could attempt to manipulate the privilege level definition as defined and present in the TOE, the functional requirement FDP_ACC.1 and the related other requirements have to be protected themselves. The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced also protect the area based memory access control function implemented according to the security functional requirement described in the security functional requirement FDP_ACC.1 with reference to the Memory Access Control Policy and details given in FDP_ACF.1. Therefore, those security functional requirements support the secure implementation and operation of FDP_ACF.1 with its dependent security functional requirements.

The requirement FDP_SDI.2.1 allows detection of integrity errors of data stored in memory. FDP_SDI.2.2 in addition allows correction of one bit errors or taking further action. Both meet the security objective O.Malfunction. The requirements FRU_FLT.2, FPT_FLS.1, and FDP_ACC.1 which also meet this objective are independent from FDP_SDI.2 since they deal with the observation of the correct operation of the TOE and not with the memory content directly.

1 9 References

[1] Security IC Platform Protection Profile, Version 1.0, 15.06.2007, BSI-PP-0035

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

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

[4] Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements; Version 3.1 Revision 5, September 2017, CCMB-2017-04--003

[5] ARMv7-M Architecture Reference Manual, ARM DDI 0403D ID021310, 12. February 2010, ARM Limited

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

[7] SLE97 M9900 Hardware Reference Manual, Revision 3.0, 2019-08-28

[10] Joint Interpretation Library, Application of Attack Potential to Smartcards, Version 2.9, January 2013

[11] SLx97 Platform Support Layer Library 32-bit Security Controller Programmer's Reference Manual, revision 5.4, 2018-07-06

[12] M9900 Errata Sheet, Rev.4.1, 2019-09-24 and M9905 M9906 Errata Sheet, Rev.3.1, 2019-06-05

[15] Anwendungshinweise und Interpretationen zum Schema (AIS), AIS31, Version 3, 2013-05-15, Bundesamt für Sicherheit in der Informationstechnik

[SHS] NIST: FIPS publication 180-4: Secure Hash Standard (SHS), August 2015

[DSS] NIST: FIPS publication 186-4: Digital Signature Standard (DSS), July 2013

[ECC] IETF: RFC 5639, Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation, March 2010, http://www.ietf.org/rfc/rfc5639.txt

[BSIG] Act on the Federal Office for Information Security (BSI-Gesetz - BSIG) of 14 August 2009, Bundesgesetzblatt I p. 2821

[9797] ISO/IEC 9797-1: 2011

[N867] National Institute of Standards and Technology (NIST), Technology Administration, U.S. Department of Commerce, NIST Special Publication 800-67, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, Revised January 2012, Revision 1

[N197] U.S. Department of Commerce, National Institute of Standards and Technology, Information Technology Laboratory (ITL), Advanced Encryption Standard (AES), FIPS PUB 197

[N38A] National Institute of Standards and Technology (NIST), Technology Administration, U.S. Department of Data Encryption Standard, NIST Special Publication 800-38A, Edition 2001

[N38B] National Institute of Standards and Technology (NIST), Technology Administration, U.S. Department of Data Encryption Standard, NIST Special Publication 800-38B, Edition 2005

[PKCS] PKCS #1: RSA Cryptography Standard, v2.2, October 27, 2012, RSA Laboratories

[X962] American National Standard for Financial Services ANS X9.62-2005, Public Key Cryptography for the Financial Services Industry, The Elliptic Curve Digital Signature Algorithm (ECDSA), November 16, 2005, American National Standards Institute

[X963] American National Standard for Financial Services X9.63-2001, Public Key Cryptograph for the Financial Services Industry: Key Agreement and Key Transport Using Elliptic Curve Cryptography, November 20, 2001, American National Standards Institute

[ST] Confidential Security Target M9900, M9905, M9906 including optional Software Libraries RSA-EC-SCL-HCL-PSL, v4.9, 2024-09-20, Infineon Technologies AG

10 Appendix

In the following tables, the hash signatures of the respective CL97 Crypto Library files are documented. For convenience purpose several hash values are referenced.

Table 21 Reference hash values of the FTL V1.01.0008 library

MD5	5abc1dca 0d92375d 3101a3cd de11faf8
SHA1	0201487a eb93b1a9 b766c02d 43a17c97 fe4c1106
SHA256	0438971c 5845d797 8d176578 b601812d e8d8e663 a09e3dc2 662b0999 7f473ea2

Table 22 Reference hash values of the CL97 v 2.05.005 Crypto libraries

Library	Hash Value
Cl97-LIB-base.lib:	
MD5	8a4a99be8204e99db9f3b409fae756f1
SHA1	dc8911a4ea23924d8dcc05429793e9d0c23504ba
SHA256	1e1e8e4b8d2ca53985e5257626db8b5c7f439b2cc88c35ee de3a34f521de8530
Cl97-LIB-ecc.lib	
MD5	e27681989fd2891e856baa03f7ac7d4a
SHA1	4a36bb058cb193db593b194ad8d6f29491afb996
SHA256	4172c0445e511e90b55f273985d90b4399ff9effb1608f18c 65812f1efbd7453
Cl97-LIB-2k.lib	
MD5	17b01b76d3353b3a6710c2ee19f1e8b0
SHA1	0011e3f4678ebecf614796d7ba697eb55ac0870d
SHA256	9dc4c346602ce6a9586ef62d4f1108b60a81e01c0e8753bf 6d4026a5ba6f59d4
Cl97-LIB-4k.lib	
MD5	d8c77f9f5a47dd587f219f73c66c82de
SHA1	f2f4169cc2331732d234848cf771c7e73341e4ae
SHA256	ac9866759a32cfca44377a88384a32bd27345cbacf2de30c 4af31aefd4872928
Cl97-LIB-toolbox.lib	
MD5	e85484a56c08343a3bfb8bc57a9f3691
SHA1	51e8329a4013f130c4ef31ca6dcd2fb311d7fc6c
SHA256	e383c7934627c6e4638b6e5b6b224913d6fc4e216431bc1 b51301a02b9b6bf25

Table 23 Reference hash values of the CL97 v2.07.003 Crypto libraries

Library	Hash Value
Cl97-LIB-base.lib:	
MD5	0581debbbc6bf992c1b979bcedb731c5
SHA1	d1afcc8ec1c898774f238568c7d8e159e4dd5f68
SHA256	02009f6c7b84b6e3d148dfa761143052720361c14babccc265aa8ce5a22a947a
Cl97-LIB-ecc.lib	
MD5	6542752d79576891580c2daa395ef66b
SHA1	097047756b24bf138b384357d6751ea8e33d8dff
SHA256	8f72c8ebdad3c99c59e9d115b284e6245122bb9ab38bd93da247c282c1526383
Cl97-LIB-2k.lib	
MD5	7ab7f68c7eb0e8a0c6ea97c3185d882c
SHA1	0e33b00961cba3d0be34352a6e7d9ddb6e12961e
SHA256	701ecd9bcd4cd828982e7a9db35820c2e4482b98263492b9072230a352d4d2e8
Cl97-LIB-4k.lib	
MD5	f5a36e6b9ff47d877c0e74e823e03d56
SHA1	bbb4fed8e9c37c180417602ac8395130b6c0fea5
SHA256	a8b6654f1302a9766ded5102b0ac6f93795bba1163885c44d0209df137b1fb7e
Cl97-LIB-toolbox.lib	
MD5	016053d0479897706ccd0638c39fd8f2
SHA1	ef499d2d497a5797e4ebf376b011d3ce41391d3b
SHA256	5efc01016edbfc3e5de35e7c58700a63f1c1bcbea35be277e7743659587aa22c

Table 24 Reference hash values of the SCL97 v2.01.011 crypto libraries

Scl97-SCP-v3-LIB-cipher.lib:	
MD5	a4c45e84dd9e2f651edf1ffaa077190d
SHA1	4cbda743be21b29de6e826112120ea11a10c2641
SHA256	998ea5e14a36ef20fc6c7c5d6c511adaac8dfa0a411bd2b96e1cbd9eee3596eb
Scl97-SCP-v3-LIB-des.lib:	
MD5	041abd8e8233e1b407d777b2734fbbd6
SHA1	ebfe549e8ac2092f1b03438f2ecb4995839b0c9a
SHA256	784ed8ca5b60ee0ac91df10b6871429a3db77e1166b2142116f4f0b61258d83d

Scl97-SCP-v3-LIB-aes.lib:	
MD5	fd061c43a23c3a256ee2aa89dbfc3d27
SHA1	c3516afe6cf16f635704ebbc751ec2763b8115bd
SHA256	57e6a9100d635d6df05241edb2874e3cbc2006927361ba4976044f7d996e48ae

Table 25 Reference hash values of the SCL97 v2.02.010 crypto libraries

Scl97-SCP-v3-LIB-cipher.lib:	
MD5	4d8cb3b84d95c386fb87ef6028d56404
SHA1	70f41981bd933db8f603b493c5fb035162e7758e
SHA256	848f57e48083a8b76c1e45de4dc006e3ed9b8bcf08a683d4d9466b56b924c6d2
Scl97-SCP-v3-LIB-des.lib:	
MD5	25b52ec314713ac21e9b3300011f0c6f
SHA1	6ff3713a41413b0fc1badef971c4ff55a6fd5d5b
SHA256	a633636604ab515251370fdfe28007ad2cb3baa190340c4a3eb88e2fcbf815b7
Scl97-SCP-v3-LIB-aes.lib:	
MD5	c8ca2dc013450d67d17d74deaeda79f0
SHA1	6bf7b1e082235ca5c0239ad070214a0337b89a13
SHA256	702653de911da2e3f5baa4ff7ff541a8ee43452dd2439373d55c876027125598

Table 26 Reference hash values of the SCL97 v2.04.003 crypto libraries

Scl97-SCP-v3-L90-cipher.lib:	
MD5	874995a6f7415d8da60e2afdfb4df5f0
SHA1	5c2b7aaa2b37c4275cdbc118a6abff4b55e02854
SHA256	ee78b3e7317947e1222707f7f1b012cbe6ad2efcaf455efa40937a64506899ae
Scl97-SCP-v3-L90-mac.lib:	
MD5	33d8a4c2a2396eaed10653c92bdfcdc6
SHA1	4597f5b6263a63fc57313e44d3a33f3485c3d14f
SHA256	ce94d6475e327aab96785c206ce08237ce334e174795bd07e0973d47f0ac8aaa
Scl97-SCP-v3-L90-des.lib:	
MD5	9d15777997d613d4ace3c6eae0314b05
SHA1	33e9f042611c9f6056f73d3dd7297b62c09ff705
SHA256	1ee72d08d96ec201fd44f7a5660595f0056961e6c7ddbd5985f193e01c00284e
Scl97-SCP-v3-L90-aes.lib:	
MD5	e5d105c9cacf024c7b7983ea9ae3cc75

SHA1	71608b626dc10d3f0931b3907dbd1bf5eb942d6a
SHA256	92c378ab9bee9f59c399475725015910535c0ff51e638067801e93cad7d4717f

Table 27 Reference hash values of the NRG libraries

Library/object file	Hash Value
NRGManagement-01.03.0927-M9900.lib	
MD5	874e529dabef8419e672c44a94600963
SHA1	9a2cfae606dc562b0d3b2dc6abcad1e11c7829f5
SHA256	95f2ed5f6a3001146c9fef36c539ac2844839af0bacb92e44a2da474e6d5aa8e
NRGReader-01.02.0800-M9900.lib	
MD5	5be6e2e9eb0f2a4847f7fc4bacf8126a
SHA1	f0cbc090657fa09d5f23e1f468e95c245f63f6fb
SHA256	16848631a68b3094e29790ee34bbb95af208a9985c8e111f46849fffc4ef3385

Table 28 PSL library v4.00.09

Psl90.lib	
MD5	7afa798cca7307789cb0611816f85998
SHA1	52f8e9acc0677c20e4b9826e0ee969e09cf9ef66
SHA256	81a2e6e9b8e8793ab6b1e11c9f11afc3a5debea65a6b1cd777da7f0fb25d31c8

Table 29 PSL library v4.00.10

Psl90.lib	
MD5	4263eb7321e170d89199593bf915faa5
SHA1	2f46f032e919991d93ea1fae9f9db3f096044d6a
SHA256	9451d17d6876d38b19613241220fdc0574b7d88319d2f32181b819d34ea3eef7

Table 30 PSL library v5.00.06

Psl90.lib	
MD5	b41ba56c9239124f10a241ed1ac775f3
SHA1	2c3cfc01e66ad307deb43d835db5a4d85a03286f
SHA256	ac256bd880d528b33ffe0594265378d3142b912357ea616b5e68a06eecd83572

Table 31 HCL library v1.01.003

HCL97-CPU-L90-hash.lib	
MD5	3d83d1294aa70fdf9b4b3883d617990a
SHA1	44db57a695407da941b351fd909d460fbfa10825
SHA256	3d8712eaf73fe89c83b978c8ba583329df2762086c7b8767515d3c3ef4560ede
HCL97-CPU-L90-sha.lib	
MD5	b76d81c778e8ecb7efd5d20ac8d8f011
SHA1	0bcf346dc501c685616bf4f18ed90ea0640d1699
SHA256	50a0117ae0392a3928735fb9d777dcd5dd551b2c296317185b50bf2e6c184b79

11 List of Abbreviations

AES	Advanced Encryption Standard
AIS31	“Anwendungshinweise und Interpretationen zu ITSEC und CC Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren”
API	Application Programming Interface
BOS	Boot Software
CC	Common Criteria
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
Crypto2304T	Asymmetric Cryptographic Processor
CRT	Chinese Reminder Theorem
DPA	Differential Power Analysis
DFA	Differential Failure Analysis
EC	Elliptic Curve
ECC	Error Correction Code
EDC	Error Detection Code
EDU	Error Detection Unit
GCIM	Generic Chip Identification Mode (BOS-CIM)
EEPROM	Electrically Erasable and Programmable Read Only Memory
EMA	Electro magnetic analysis
HW	Hardware
IC	Integrated Circuit
ID	Identification
IMM	Interface Management Module
I/O	Input/Output
MED	Memory Encryption and Decryption
MPU	Memory Protection Unit
NRG	ISO/IEC14443-3 Type A with CRYPTO1
O	Objective
OS	Operating system
PSL	Platform Support Layer
RAM	Random Access Memory
RMS	Resource Management System
RNG	Random Number Generator
ROM	Read Only Memory
RSA	Rives-Shamir-Adleman Algorithm
SCL	Symmetric Crypto Library

SCP	Symmetric Cryptographic Processor
SF	Security Feature
SFR	Special Function Register, as well as Security Functional Requirement
SPA	Simple power analysis
SW	Software
T	Threat
TM	Test Mode (BOS)
TOE	Target of Evaluation
TRNG	True Random Number Generator
TSF	TOE Security Functionality
UART	Universal Asynchronous Receiver/Transmitter
UM	User Mode (BOS)
UMSLC	User Mode Security Life Control
3DES	Triple DES Encryption Standard

12 Glossary

Boot System	Part of the firmware with routines for controlling the operating state and testing the TOE hardware
Central Processing Unit	Logic circuitry for digital information processing
Chip	Integrated Circuit]
Chip Identification Mode data	Data stored in the SOLID FLASH™ NVM containing the chip type, lot number (including the production site), die position on wafer and production week and data stored in the ROM containing the BOS version number
Chip Identification Mode	Operational status phase of the TOE, in which actions for identifying the individual chip by transmitting the Chip Identification Mode data take place
Controller	IC with integrated memory, CPU and peripheral devices
Crypto2304T	Cryptographic coprocessor for asymmetric cryptographic operations (RSA, Elliptic Curves)
Cyclic Redundancy Check	Process for calculating checksums for error detection
Electrically Erasable and Programmable Read Only Memory (SOLID FLASH™ NVM)	Non-volatile memory permitting electrical read and write operations
Firmware	Part of the software implemented as hardware
Hardware	Physically present part of a functional system (item)
Integrated Circuit	Component comprising several electronic circuits implemented in a highly miniaturized device using semiconductor technology
Memory Encryption and Decryption	Method of encoding/decoding data transfer between CPU and memory
Memory	Hardware part containing digital information (binary data)
Microprocessor	CPU with peripherals
Non-privilege level	Restricted (non Supervisor) mode of the CPU
Object	Physical or non-physical part of a system which contains information and is acted upon by subjects
Operating System	Software which implements the basic TOE actions necessary for operation
Privilege level	Supervisor mode of the CPU
Programmable Read Only Memory	Non-volatile memory which can be written once and then only permits read operations
Random Access Memory	Volatile memory which permits write and read operations
Random Number Generator	Hardware part for generating random numbers

Read Only Memory	Non-volatile memory which permits read operations only
Resource Management System	Part of the firmware containing SOLID FLASH™ NVM programming routines, AIS31 testbench etc.
Security Mechanism	Logic or algorithm which implements a specific security function in hardware or software
SCP	Symmetric cryptographic coprocessor for symmetric cryptographic operations (3DES, AES).
Security Function	Part(s) of the TOE used to implement part(s) of the security objectives
Security Target	Description of the intended state for countering threats
Smart Card	Plastic card in credit card format with built-in chip
Software	Information (non-physical part of the system) which is required to implement functionality in conjunction with the hardware (program code)
Subject	Entity, generally in the form of a person, who performs actions
Target of Evaluation	Product or system which is being subjected to an evaluation
Test Mode	Operational status phase of the TOE in which actions to test the TOE hardware take place
Threat	Action or event that might prejudice security
User	Person in contact with a TOE who makes use of its operational capability
User Mode	Operational status phase of the TOE in which actions intended for the user takes place
WLB	Wafer Level Ballgrid Array
WLP	Wafer Level Package

Revision History

Major changes since the last revision

Page or Reference	Description of change
4.6	Final version

Trademarks of Infineon Technologies AG
AURIX™, C166™, CanPAK™, CIPOS™, CoolGaN™, CoolMOS™, CoolSET™, CoolSiC™, CORECONTROL™, CROSSAVE™, DAVE™, DI-POL™, DrBlade™, EasyPIM™, EconoBRIDGE™, EconoDUAL™, EconoPACK™, EconoPIM™, EiceDRIVER™, eupec™, FCOS™, HITFET™, HybridPACK™, Infineon™, ISOFACE™, IsoPACK™,
i-Wafer™, MIPAQ™, ModSTACK™, my-d™, NovalithIC™, OmniTune™, OPTIGA™, OptiMOS™, ORIGA™, POWERCODE™, PRIMARION™, PrimePACK™, PrimeSTACK™, PROFET™, PRO-SIL™, RASIC™, REAL3™, ReverSave™, SatRIC™, SIEGET™, SIPMOS™, SmartLEWIS™, SOLID FLASH™, SPOC™, TEMPFET™, thinQ!™, TRENCHSTOP™, TriCore™.

Trademarks updated August 2015

Other Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2024-09-20

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Email: erratum@infineon.com

Document reference
Doc_Number

IMPORTANT NOTICE
The information contained in this application note is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this application note.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology delivery terms and conditions and prices please contact your nearest Infineon Technologies office (www.infineon.com).

WARNINGS
Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.

Edition 2024-09-20

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Email: erratum@infineon.com

Document reference
ifx1

IMPORTANT NOTICE
The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics („Beschaffenheitsgarantie") .

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology delivery terms and conditions and prices please contact your nearest Infineon Technologies office (www.infineon.com).

WARNINGS
Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.