PUBLIC

Infineon Technologies AG
Chipcard and Security

Evaluation Documentation
Security Target Lite

M7793 A12

including optional Software Libraries
RSA – EC – Toolbox

Version 2.0

Date 2011-09-12

Author Steffen Heinkel

REVISION HISTORY

1.0	2011-07-26: draft version
1.1	2011-09-05: added description of Security Functions
2.0	2011-09-12: version to be provided to BSI

TABLE OF CONTENTS

1 Security Target Lite Introduction (ASE_INT)

1.1 Security Target Lite and Target of Evaluation Reference

The title of this document is Security Target Lite (ST Lite). The Security Target Lite comprises the Infineon Technologies SmartCard IC (Security Controller) M7793 A12 with optional RSA v1.02.010, EC v1.02.010 and Toolbox v1.02.010 libraries and with specific IC-dedicated software.

The target of evaluation (TOE) M7793 A12 is described in the following sections. The Security Target Lite has the revision 2.0 and is dated 2011-09-12.

The Target of Evaluation (TOE) is a smartcard IC (Security Controller) as listed in Table 1: Identification and its blocked derivatives. The design step is A12.

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

The Protection Profile and the Security Target Lite are built in compliance to Common Criteria v3.1.

The ST Lite takes into account all relevant current final interpretations.

The targeted certificate is EAL4+.

Table 1: Identification

	Version	Date	Registration
Security Target	this version	see cover page	M7793 A12
Target of Evaluation	A12		M7793 A12
			with Flash Loader V3.92.009
			and Flash Loader patch version V3.93.003
			and RMS V7790b0118
			and STS V77.05.0d.06
			and STS Patch V7206
			and SAM V25b01
			and Overall Patch v7032
			and optional SW:
			RSA2048 V1.02.010
			RSA4096 V1.02.010
			EC V1.02.010
			Toolbox V1.02.010
			and guidance documentation
Guidance Documentation	Edition	2010-11	SLx 70 Family Hardware Reference Manual
		2011-06	SLx 70 Family Production and Personalization User's Manual
		2011-01	SLE 70 Family Programmer's Reference User's Manual
		2011-06	SLx 77CFXxxxP (M7793) Security Guidelines User's manual
		2011-07-07	SLx 70 Family SLx 77 Product Group (L90FL Technology) Errata Sheet
		2011-04	SLE77 Asymmetric Crypto Library for Crypto@2304T RSA/ECC/Toolbox User Interface (optional)
		2010-03	Crypto@2304T User Manual
Evaluation based on Protection Profile	1.0	15.06.2007	Security IC Platform Protection Profile PP0035
Common Criteria	Version 3.1 Revision 3	2009-July	Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model CCMB-2009-07-001 Part 2: Security functional requirements CCMB-2009-07-002 Part 3: Security Assurance Components CCMB-2009-07-003

A customer can identify the TOE and its configuration (for details see chapter 2.2.7) using the Non-ISO ATR in combination with firmware functions. The TOE answers the Non-ISO-ATR with a Chip Identification Mode (CIM). This CIM outputs a chip identifier byte, design step and firmware versions. The RMS base version and configuration of the TOE (memory size and available peripherals) can be obtained by dedicated RMS functions. The Flash Loader offers a function to extract its version.

1.2 Target of Evaluation Overview

The TOE comprises the Infineon Technologies SmartCard IC (Security Controller) M7793 A12 with specific IC-dedicated software and optional RSA, EC and Toolbox libraries.

This Security Target Lite (ST Lite) describes the TOE known as the Infineon Technologies AG security controller group as listed in Table 1 and gives a summary product description.

The TOE is a member of the Security Controller family SLE70 and meets high requirements in terms of performance and security.

The TOE provides a 16-bit CPU-architecture and is compatible to the Intel 80251 architecture. The major components of the core system are the non-standard CPU, the MMU (Memory Management Unit) and MED (Memory Encryption/Decryption Unit). The TOE implements a 16-MByte linear addressable memory space, a simple scaleable Memory Management concept and a scaleable stack size. The flexible memory concept consists of ROM and SOLID FLASH™[1]. For the SOLID FLASH™ the Unified Channel Programming (UCP) memory technology is used.

The RMS library providing some functionality via an API to the Smartcard Embedded Software contains, for example, SOLID FLASH™ service routines. The service algorithm provides functionality for the tearing-safe writing to the SOLID FLASH™. The STS firmware is used for test purposes during startup and the Flash Loader allows downloading of user software to the NVM during the manufacturing process. The STS resides in a dedicated test ROM area, that is part of the TOE.

The symmetric coprocessor (SCP) combines both AES and triple DES with dual-key or triple-key hardware acceleration. The asymmetric crypto coprocessor, called Crypto2304T in the following, supports RSA-2048 bit (4096-bit with CRT) and Elliptic Curve (EC) cryptography, for example.

The software part of the TOE consists of the cryptographic libraries RSA and EC and the supporting Toolbox and Base libraries. If RSA or EC or Toolbox is part of the shipment, the Base Library is 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 the generation of RSA key pairs (RsaKeyGen), RSA signature verification (RsaVerify), RSA signature generation (RsaSign) and RSA modulus recalculation (RsaModulus). 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. Following the BSI[2] recommendations, 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 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 verification, ECDSA key generation and Elliptic Curve Diffie-Hellman key agreement. The EC library is delivered as object code. The certification covers the standard NIST [14] and Brainpool [15] Elliptic Curves with key lengths of 192 to 521 bits. Other types of elliptic curves can be added by the user during a composite certification process.

The Toolbox library provides long integer and modular arithmetic operations. It does not support any security-relevant policy or function.

---

[1] SOLID FLASH™ is an Infineon Trade Mark and stands for Flash EEPROM technology.

[2] BSI Bundesamt für Sicherheit in der Informationstechnik – Federal office for information security.

The Base Library provides the low-level interface to the asymmetric cryptographic coprocessor for the cryptographic libraries and has no user interface. It does not support any security relevant policy or function.

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

2 Target of Evaluation Description

The TOE description helps the reader 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.

2.1 TOE Definition

The TOE consists of smartcard 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 (L90FL). This TOE is intended to be used in smartcards and for its previous use as a development platform for smartcard 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.

Figure 1 shows the block diagram of the M7793:
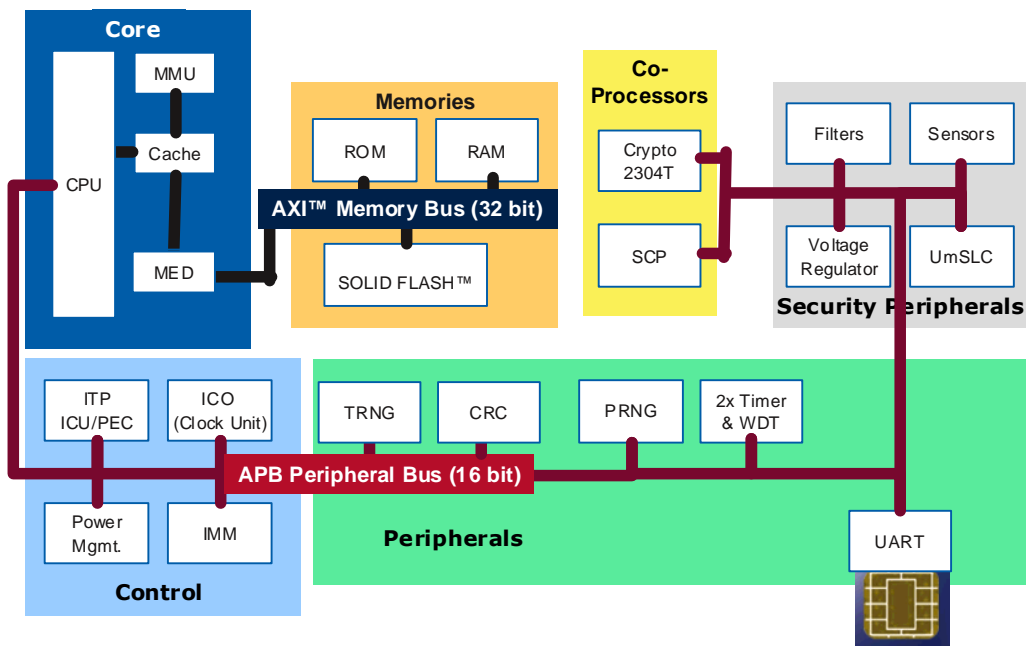
Figure 1: Block diagram of the TOE

The TOE consists of a core system, memories, coprocessors, peripherals, security modules and analog peripherals. The major components of the core system are the non-standard CPU, the MMU (Memory Management Unit) and MED (Memory Encryption/Decryption Unit). The coprocessor block contains the processors for RSA/EC and 3DES/AES processing, while the peripheral block contains the random number generation and the external interfaces service. The

peripheral block also contains also the timers and a watchdog. All data of the memory block is encrypted, RAM and ROM are equipped with an error detection code and the SOLID FLASH™ is equipped with an error correction code (ECC). Security modules manage the alarms. Alarms may be triggered when the environmental conditions are outside the specified operational range.

The CPU accesses memory via the integrated Memory Encryption and Decryption unit (MED). The access rights of the application to the memories can be controlled with the memory management unit (MMU). Errors in RAM and ROM are automatically detected (EDC, Error Detection Code) in terms of the SOLID FLASH™ 1-Bit-errors are also corrected (ECC, Error Correction Code).

The controller of this TOE stores both code and data in a linear 16-Mbyte memory space, allowing direct access without the need to swap memory segments in and out of memory using a memory management unit.

The cache is a high-speed memory buffer located between the CPU and (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 TRNG (True Random Number Gerenator) is specially designed for smartcard applications. The TRNG fulfils the requirements of the functionality class P2 of the AIS31 and produces genuine random numbers which then can be used directly or as seed for the PRNG (Pseudo Random Number gerenator). The PRNG is not in the scope of the evaluation.

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

The Clock Unit (CLKU) supplies the clocks for all components of the TOE. The Clock Unit can work in an internal and external clock mode. When operating in the internal clock mode the system frequency is derived from an internal DCO, whereas in external clock mode, the system clock is derived from an externally applied interface clock according to a defined dependency.

The Crypto2304T coprocessor allows calculation of asymmetric algorithms like RSA and Elliptic Curve (EC). The Crypto2304T is optimized for security and low power consumption.

The Symmetric Cryptographic Processor (SCP) allows calculation of dual-key or triple-key triple-DES and AES. The SCP is optimised for security and low power consumption. The SCP module computes the complete DES algorithm within a few clock cycles and is designed to counter attacks like DPA, EMA and DFA.

The STS (self-test software), RMS (Resource Management System), Service Algorithm Minimal (SAM) and Flash Loader together compose the TOE firmware stored in the ROM. All mandatory functions for internal testing, production usage and start-up behavior (STS), and also the RMS and SAM functions are grouped together in a common privilege level. These privilege levels are protected by a hardwired Memory Management Unit (MMU) setting.

The user software has to be implemented in NVM. The user can choose, whether the software is loaded into the SOLID FLASH™ by Infineon Technologies AG or by the user

The TOE uses Special Function Registers (SFRs). These SFRs are used for general purposes and chip configuration; they are located in SOLID FLASH™ in a configuration area page.

The bus system comprises two separate bus entities: a memory bus supporting the AXI™ protocol (Advanced eXtensible Interface) and an APB™ (Advanced Peripheral Bus) for high-speed communication with the peripherals.

An intelligent shielding algorithm finishes the upper layers above security critical signals and wires, finally providing the so called "I^2-shield".

The following is a list of features provided by the TOE:

- 24-bit linear addressing

- Up to 16 Mbytes of addressable memory

- Register-based architecture (registers can be accessed as bytes, words (2 bytes), and doublewords (4 bytes))

- 2-stage instruction pipeline

- Extensive set of powerful instructions, including 16- and 32-bit arithmetic and logic instructions

- Cache with single-cycle access searching

- 16-bit ALU

2.2 Scope of the TOE

The TOE comprises three parts:

1. Hardware of the smartcard security controller

2. Associated firmware and software

3. Documents

The hardware configuration options and configuration methods are described in Section 1.1.

The second part of this TOE includes the associated firmware and software required for operation and cryptographic support.

The documents as described in Section 2.2.4 and listed in Table 1, are supplied for user guidance. In the following description, the term "manufacturer" stands for Infineon Technologies AG, the manufacturer of the TOE. The Smartcard Embedded Software or user software is not part of the TOE.

2.2.1 Hardware of the TOE

The hardware part of the TOE (see Figure 1) as defined in [1] comprises the following:

Core System

- Proprietary CPU implementation of the Intel[TM] MCS251 standard architecture from a functional perspective

- Memory Encryption/Decryption Unit (MED)

- Memory Management Unit (MMU)

Memories

- Read-Only Memory (ROM)

- Random Access Memory (RAM)

- SOLID FLASH™

Peripherals

- True Random Number Generator (TRNG)

- Pseudo Random Number Generator (PRNG)

- Watchdog and timers

- Universal Asynchronous Receiver/Transmitter (UART)

- Checksum module (CRC)

Control

- Dynamic Power Management

- Internal Clock Oscillator (ICO)

- Interrupt and Peripheral Event Channel Controller (ITP and PEC)

- Interface Management Module (IMM)

- User mode Security Life Control (UmSLC)

- Voltage regulator

Coprocessors

- Crypto2304T for asymmetric algorithms like RSA and EC (optionally blocked)

- Symmetric Crypto Coprocessor for AES and 3DES Standard

Security Peripherals

- Filters

- Sensors

Buses

- AXI^TM Memory Bus

- APB Peripheral Bus

2.2.2 Firmware and Software of the TOE

The entire firmware of the TOE consists of different parts, as described below:

One part comprises the RMS and SAM routines for SOLID FLASH™ programming, security functions test, and random number online testing (Resource Management System, IC Dedicated Support Software in PP [1]). The RMS and SAM routines are stored by Infineon Technologies AG in a reserved area of the normal user ROM.

The second part is the STS, consisting of test and initialization routines (Self Test Software, IC Dedicated Test Software in PP [1]). The STS routines are stored in a specially protected test ROM and are not accessible by user software.

The third part is the Flash Loader, a piece of software located in the user ROM. It supports download of user software or parts of it to SOLID FLASH™. After completion of the download the Flash Loader can be deactivated permanently by the user.

The optional software part of the TOE consists of the cryptographic libraries RSA and EC and the supporting Toolbox and Base libraries.

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 generation of RSA Key Pairs (RsaKeyGen), the RSA signature verification (RsaVerify), the RSA signature generation (RsaSign) and the RSA modulus recalculation (RsaModulus). 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[3] 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 [14] and Brainpool [15] Elliptic Curves with key lengths of 192 to 521 bits. Other types of elliptic curves can be added by the user during a composite certification process.

The toolbox library provides long integer and modular arithmetic operations. It does not support any security relevant policy or function.

The Base Library provides the low level interface to the asymmetric cryptographic coprocessor for the cryptographic libraries and has no user available interface. It does not support any security relevant policy or function.

2.2.3 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 includes the pads of the chip, particularly the contacted RES, I/O, CLK lines and supply lines VCC and GND. The communication meets ISO 7816/ETSI/EMV standards.

- The data-oriented I/O interface of the TOE is represented by the I/O pad.

- The interface to the firmware consists of special registers used for hardware configuration and control (Special Function Registers, SFR).

- The interface of the TOE to the operating system is covered by the RMS routines and by the instruction set of the TOE.

- The interface of the TOE to the test routines is formed by the STS test routine call, i.e. entry to test mode (STS-TM entry).

- The interface to the RSA calculations is defined by the RSA library

- The interface to the EC calculations is defined by the EC library

- The interface to the Toolbox basic arithmetic functions is defined by the Toolbox library.

---

[3] CRT Chinese Reminder Theorem

Note that the interfaces of cryptographic libraries (RSA and EC) and the toolbox library are optional, as these depend on the procurement order.

2.2.4 Guidance Documentation

The guidance documentation consists of:

- SLx 70 Family Hardware Reference Manual

- SLx 70 Family Production and Personalization User's Manual

- SLE 70 Family Programmer's Reference User's Manual

- SLx 70 Family SLx 77 Product Group (L90FL Technology) Errata Sheet

  These documents contain the description of all interfaces of the software to the hardware relevant for programming the TOE.

- SLx 77CFXxxxP (M7793) Security Guidelines User's manual: This document provides secure coding guidance to the application writer.

- SLE77 Asymmetric Crypto Library for Crypto@2304T RSA/ECC/Toolbox User Interface (optional): User Interface, contains all interfaces of the cryptographic RSA- and EC libraries, as well as of the Toolbox library. This document is only delivered to the user in case the RSA library and/or the EC library and/or the Toolbox library is/are part of the delivered TOE.

- Crypto@2304T User Manual, describing the architecture of cryptographic coprocessor on register level. It also provides a functional description of the register architecture, instruction set and gives programming guidance.

Finally the certification report may contain an overview of recommendations to a software developer regarding the secure use of the TOE.

2.2.5 Forms of Delivery

The TOE can be delivered in the form of complete modules, as plain wafers in an IC case (e.g. DSO20) or in bare dies. The delivery can therefore be at the end of phase 3 or at the end of phase 4 which may also include pre-personalization steps according to [1]. In any case the testing of the TOE is finished and the extended test features are removed. From a security policy point of view the different forms of delivery do not have any impact.

The delivery to the software developer (phase 2 ➔ phase 1) contains the development package, which is delivered in electronic form. It contains the documents as described above, the development and debugging tools.
Part of the software delivery is the Flash Loader program, provided by Infineon Technologies AG, running on the TOE and controlling the download of user software onto the TOE via the UART interface. 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.

2.2.6 Production sites

The TOE may be handled at different production sites but the silicon is produced in Dresden only, as listed below. To distinguish the different production sites of various products in the field, the site is coded in the Chip Ident Mode data. The exact coding of the chip identification data is described in [7].

The delivery measures are described in the ALC_DVS aspect.

Table 2: Production site in chip identification

Production Site	Chip Identification
Dresden	Bits 7:4 of batch byte number 06: 0010

2.2.7 TOE Configuration

This TOE is represented by various configurations called products. All are derived from the same configurable hardware M7793. 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 degree of freedom for configuring the TOE is predefined by Infineon Technologies AG.

RMS functions allow a customer to extract the present hardware configuration. The original Chip Identifier Byte, which was valid before blocking, can also be extracted. Depending on the frequency configuration two original Chip Identifier Bytes are available 0xEF and 0xE3.

Two methods are available to customers to configure the TOE:

1. To order a configuration, which is defined and offered by Infineon Technologies.
2. To apply the Bill-Per-Use (BPU) method for the TOE. This method enables a customer to use tailored products of the TOE within the TOE´s configuration options

BPU allows a customer to block chips on demand at the customer´s premises. Customers, who intend to use this feature receive the TOEs in a predefined configuration, e.g. no blocking applies. The blocking information is part of a chip configuration area. The blocking information can be modified by customers using specific APDUs. Once final blocking is done, further modifications are disabled.

The BPU software part is only present on predefined products, which have been ordered with the BPU option. In all other cases this software is not present on the product.

2.2.8 TOE initialization with Customer Software

Several options are available to initialize the TOE with customer software:

Table 3: Options to initialize the TOE with customer software

	User software implementation options	Availability of Flash Loader
1.	User software is initialized by the customer.	Flash Loader required.
2	User software is initialized by Infineon Tech.	Flash Loader availability to customer is optional. This means, that the TOE can be delivered without flashloader. In this case the flashloader is not available.

3 Conformance Claims (ASE_CCL)

3.1 CC Conformance Claim

This Security Target Lite (ST Lite) 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 Lite is claimed for:
Common Criteria part 2 extended and Common Criteria part 3 conformant.

3.2 PP Claim

This Security Target Lite 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[4] (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 4: Augmentations of the assurance level of the TOE

Assurance Class	Assurance components	Description
Test depth	**ATE_DPT.2**	Testing: Security enforcing modules
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 Lite does not claim conformance to a package of [1].

The assurance level for the TOE is EAL4 augmented with the components ALC_DVS.2, ATE_DPT.2 and AVA_VAN.5.

---

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

3.4 Conformance Rationale

This Security Target Lite claims strict conformance to [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 [1], the security problem definition is enhanced by adding a threat, an organization security policy and an augmented assumption. Including these add-ons, the security problem definition of this Security Target Lite is consistent with the statement of the security problem definition in [1], as the Security Target Lite claimed strict conformance to the [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 T.Mem-Access, 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 Lite remains conformant to CC [2], claim 482 as the possibility to introduce additional restrictions is given.

- The Security Target Lite 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, and due to the memory access control - O.Mem-Access, 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 Lite remains conformant to CC [2], claim 482 as the possibility to introduce additional restrictions is given.

- The Security Target Lite 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 Lite are consistent with the statement of the security objectives in the PP [1], as the Security Target Lite claimed strict conformance to the PP [1].

All security functional requirements defined in the PP [1] are included and completely defined in this ST Lite. 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 Lite:

- 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])

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

All assignments and selections of the security functional requirements are done in the PP [1] and in this Security Target Lite in section 7.2.

The Assurance Requirements of the TOE obtain the Evaluation Assurance Level 4 augmented with the assurance components ALC_DVS.2, ATE_DPT.2 and AVA_VAN.5 for the TOE.

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 5: 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 6: 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 organisational 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 organisational 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 7: 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 [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)

Note 1:
The cryptographic libraries RSA and EC and the Toolbox library are delivery options. . If one of the libraries RSA, EC or Toolbox are delivered, the 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 Toolbox and Base Library are no cryptographic libraries and provide no additional specific security functionality.

End of note.

Note 2:

The Crypto2304T can be blocked. The blocking depends on the user's choice prior to the production of the hardware. No accessibility of the Crypto2304T 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 Crypto2304T

End of note.

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 8: 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 [1] section 3.4.

5 Security objectives (ASE_OBJ)

This section shows the subjects and objects, which 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 of the TOE

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

Table 9: 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 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)

Note 3:
The cryptographic libraries RSA and EC and the Toolbox library are delivery options. If one of the libraries RSA, EC or Toolbox is delivered, the 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 Toolbox and Base Library are no cryptographic libraries and provide no additional specific security functionality.

End of note.

Note 4:

The Crypto2304T can be blocked. The blocking depends on the user's choice prior to the production of the hardware. No accessibility of the Crypto2304T 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 Crypto2304T

End of note.

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 10: 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 [1] section 4.2 and 4.3. The table below lists the security objectives.

Table 11: Security objectives for the environment according to [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 realised 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 12: Security Objective Rational

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 organisational 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 [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 [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 realised 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 objective O.RND corresponds directly to the description of the threat T.RND. Therefore T.RND is covered by O.RND.

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 FCS_RNG, FMT_LIM and FAU_SAS are defined and described in PP [1] section 5. The component FPT_TST.2 is defined in the following.

6.1 Component "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 (438,439).

Component levelling

```
┌──────────────────────────────────────────────┐      ┌─────┐
│  FPT_TST     TSF self test                     │─────<│  1  │
└──────────────────────────────────────────────┘      └─────┘
                                                       ┌─────┐
                                                       │  2  │
                                                       └─────┘
```

FPT_TST.1:   The component FPT_TST.1 is defined in [3] section 15.14 (440, 441,442).

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 conditions

- management 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].

7 Security Requirements (ASE_REQ)

For this section [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 [1] section 6.1 and in the following description.

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

Table 13: 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
FCS_RNG.1	"Quality metric for random numbers"	No

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

Table 14: 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 initialisation"
FMT_SMF.1	"Specification of Management functions"
FCS_COP.1	"Cryptographic support"
FCS_CKM.1	"Cryptographic key management"
FDP_SDI.1	"Stored data integrity monitoring
FDP_SDI.2	"Stored data integrity monitoring and action"

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

The above marked extended components FMT_LIM.1 and FMT_LIM.2 are introduced in [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 here. This family describes the functional requirements for random number generation used for cryptographic purposes.

FCS_RNG.1	Random Number Generation
Hierarchical to:	No other components
Dependencies:	No dependencies
FCS_RNG.1	Generation of random numbers requires that random numbers meet a defined quality metric.
FCS_RNG.1.1	The TSF shall provide a physical random number generator that implements total failure test of the random source *and a continuous RNG test according to:* *National Institute of Standards and Technology, Security Requirements for Cryptographic Modules, Federal Information Processing Standards Publication (FIPS) 140-2, 1999.*
FCS_RNG.1.2	The TSF shall provide random numbers that meet *the functionality class P2 with SOF-high of [6].*

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 dependencies
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.*

Note, that the TOE can be clearly identified by the Chip Identification Mode (CIM) and dedicated RMS functions. The CIM outputs the chip identification, design step and firmware versions. Dedicated RMS functions allow a customer to extract the present hardware configuration and the original Chip Identifier Byte, which was valid before blocking.

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:	FPT_AMT.1 Abstract machine testing
FPT_TST.2.1	The TSF shall run a suite of self tests *at the request of the authorised user* to demonstrate the correct operation of the *alarm lines and/or following environmental sensor mechanisms:*

- *CORE – CPU related alarms*

- *SCP - Symmetric Cryptographic Co-Processor*

- *Temperature alarm*

- *AXI – Memory Bus*

- *NVM_MISS – NVM illegal addressing alarm*

- *FSE – Internal Frequency Sensor alarm*

- *Light – Light sensitive alarm*

- *WDT - Watch Dog Timer related alarms*

- *SW – Software triggered alarm*

- *TRNG – True Random Number Generator*

- *Glitch sensor alarm*

- *Backside light detection (BLD) - alarm*

- *RAM/ROM EDC or SOLID FLASH™ ECC*

7.1.3 Memory access control

Usage of multiple applications in one Smartcard often requires code and data separation in order to prevent one application from accessing code and/or data of another application. For this reason the TOE provides Area based Memory Access Control. The underlying memory management unit (MMU) is documented in section 4 of [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)":

Memory Access Control Policy

> *The TOE shall control read, write, delete and execute accesses of software running at the privilege levels as defined below. Any access is controlled, regardless whether the access is on code or data or a jump on any other privilege level outside the current one.*

> The memory model provides distinct, independent privilege levels separated from each other in the virtual address space. These levels are referred to as the Infineon Technologies (IFX) level, operating system 1 and 2 levels (OS1, OS2), shared application level, and application 1 and 2 levels. A pseudo-level is the "current" level, which is simply the level on which code is currently being executed. The access rights are controlled by the MMU and related to the privilege level as depicted in following diagram:

Current level	0
Reserved	1
IFX level	2
OS1	3
OS2	4
Shared application	5
Application 1	6
Application 2	7
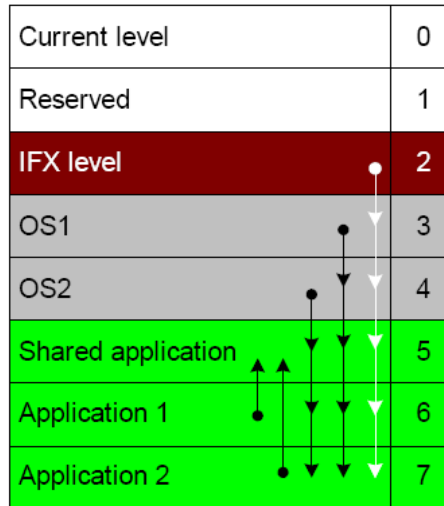
Figure 2: Privilege Levels of the TOE

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 (software running at the defined and assigned privilege levels), all objects (data including code stored in memories) and all the operations defined in the Memory Access Control Policy, i.e. privilege levels*.

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 initialisation
FDP_ACF.1.1	The TSF shall enforce the *Memory Access Control Policy* to objects based on the following: *Subject*: - *software running at the IFX, OS1 and OS2 privilege levels required to securely operate the chip. This includes also privilege levels running interrupt routines.* - *software running at the privilege levels containing the application software* *Object*: - *data including code stored in memories* *Attributes*: - *the memory area where the access is performed to and/or* - *the operation to be performed*.

FDP_ACF.1.2   The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *evaluate the corresponding permission control information of the relevant memory range before, during or after the access so that accesses to be denied can not be utilised by the subject attempting to perform the operation*.

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 initialisation

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 *well defined[5]* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2   The TSF shall allow *any subject, provided that the Memory Access Control Policy is enforced and the necessary access is therefore allowed[6],* 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 *change default, modify or delete* the security attributes *permission control information to the software running on the privilege levels.*

---

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

[6] The Smartcard Embedded Software is intended to set the memory access control policy

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*: access the configuration registers of the MMU.*

7.1.4 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 7.3.1.1.

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)[7]

Triple-DES Operation

The 3DES Operation of the TOE shall meet the requirement "Cryptographic operation (FCS_COP.1)" as specified below.

> **FCS_COP.1/3DES** 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.1.1/3DES The TSF shall perform *encryption and decryption* in accordance to a specified cryptographic algorithm *Triple Data Encryption Standard (3DES)* in *Electronic Codebook Mode (ECB), Cipher Block Chaining Mode (CBC), Blinding Feedback Mode (BLD)* and in the *Cipher Feedback Mode (CFB)8* with cryptographic key sizes *of 2 x 56 bit or 3 x 56 bit,* that meet the following *standards*
>
> *National Institute of Standards and Technology (NIST), Technology Administration, U.S. Department of Data Encryption Standard (DES), NIST Special Publication 800-67, Version 1.1*

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

---

[7] 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.

[8] The CFB is also called Recrypt Mode.

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 to a specified cryptographic algorithm *Advanced Encryption Standard (AES)* and cryptographic key sizes of *128 bit or 192 bit or 256 bit* that meet the following standards: *U.S. Department of Commerce, National Institute of Standards and Technology, Information Technology Laboratory (ITL),* *Advanced Encryption Standard (AES), FIPS PUB 197*

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 and decryption* in accordance with a specified cryptographic algorithm *Rivest-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 v2.1 RFC3447, without 5.1.1.1.* *Decryption (with or without CRT):* *According to section 5.1.2 RSADP in PKCS v2.1 RFC3447* *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 v2.1 RFC3447* *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 v2.1 RFC3447, without 5.2.2.1.*

Rivest-Shamir-Adleman (RSA) key generation

The key generation for the RSA shall meet the requirement "Cryptographic key generation (FCS_CKM.1)"

FCS_CKM.1/RSA   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
                     FMT_MSA.2 Secure security attributes

FCS_CKM.1.1/RSA   The TSF shall generate cryptographic keys in accordance with a
                  specified cryptographic key generation algorithm *rsagen1 (PKCS
                  v2.1 RFC3447)* and specified cryptographic key sizes of *1024 – 4096
                  bits* that meet the following *standard:*

                  According to section 3.2(2) in PKCS v2.1 RFC3447,
                  for u=2, i.e., without any (r_i, d_i, t_i), i > 2.
                  For p x q < 2^{2048} additionally according to section 3.2(1).

Note 5:
For easy integration of RSA 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.
End of note.

Note 6:
The TOE can be delivered with or without the RSA library. In case a user deselects the library the
TOE does not provide the Additional Specific Security Functionality Rivest-Shamir-Adleman
Cryptography (RSA) realized with the security functional requirements FCS_COP.1/RSA and
FCS_CKM.1/RSA. In case of a blocked Crypto2304T no cryptographic libraries are delivered.
End of note.

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 *192 - 521 bits* that meet the following
                  standard:

                  Signature Generation:
                  1. 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.

2. According to sections 6.2 (6.2.2. + 6.2.3) in ISO/IEC 15946-2:2002
   Not implemented is section 6.2.1:
   The output of 5.4.2 has to be provided by the caller as input to the
   function.

Signature Verification:
1. 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.

2. According to sections 6.4 (6.4.1. + 6.4.3 + 6.4.4) in ISO/IEC
   15946-2:2002
   Not implemented is section 6.4.2:
   The output of 5.4.2 has to be provided by the caller as input to the
   function.

Note 7:
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.
End of note.

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 FMT_MSA.2 Secure security attributes
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 *ISO/IEC 15946-1:2002* and specified cryptographic key sizes *192 - 521 bits* that meet the following *standard:*

ECDSA Key Generation:
1. According to the appendix A4.3 in ANSI X9.62-2005

> *Optional cofactor h is not supported.*
> *2. According to section 6.1 (not 6.1.1) in ISO/IEC 15946-1:2002*

Note 8:
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.
End of note.

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 *192 - 521 bits* that meet the following *standard:*

1. 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.

2. According to sections 8.4.2.1, 8.4.2.2, 8.4.2.3, and 8.4.2.4 in ISO/IEC 15946-3:2002:
   The function enables the operations described in the four sections.

Note 9:
The certification covers the standard NIST [14] and Brainpool [15] Elliptic Curves with key lengths of 192 to 521 Bits. Other types of elliptic curves can be added by the user during a composite certification process.
End of note.

Note 10:
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.
End of note.

Note 11:
The TOE can be delivered without the EC library. In case a user deselects the library the TOE does not provide the Additional Specific Security Functionality Elliptic Curve Cryptography realised with the security functional requirements FCS_COP.1/ECSA, FCS_COP.1/ECDH and

FCS_CKM.1/EC. In case of a blocked Crypto2304T no cryptographic libraries are delivered.
End of note.

Note 12:
The EC primitives allow the selection of various curves. The selection of the curves depends on the user.
End of note.

In case of a blocked Crypto2304T coprocessor no cryptographic libraries are delivered.

7.1.5 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* based on the following attributes: *EDC value for ROM and RAM smart parity for the cache and ECC value for the SOLID FLASH™ and verification of stored data in SOLID FLASH™.*

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 ROM and RAM, smart parity for Cache and error correction ECC for the SOLID FLASH™.*
FDP_SDI.2.2	Upon detection of a data integrity error, the TSF shall *correct 1 bit errors in the SOLID FLASH™ automatically and inform the user about other bit errors.*

7.2 TOE Security Assurance Requirements

The evaluation assurance level is EAL 4 augmented with ALC_DVS.2, ATE_DPT.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 15: Assurance components

Aspect	Acronym	Description	Refinement
Development	ADV_ARC.1	Security Architecture Description	[1]
	ADV_FSP.4	Complete functional specification	[1]
	ADV_IMP.1	Implementation representation of the TSF	[1]
	ADV_TDS.3	Basic modular design	
Guidance Documents	AGD_OPE.1	Operational user guidance	[1]
	AGD_PRE.1	Preparative procedures	[1]
Life-Cycle Support	ALC_CMC.4	Production support, acceptance procedures and automation	[1]
	ALC_CMS.4	Problem tracking CM coverage	[1]
	ALC_DEL.1	Delivery procedures	[1]
	ALC_DVS.2	Identification of security measures	[1]
	ALC_LCD.1	Developer defined life-cycle model	
	ALC_TAT.1	Well-defined development tools	
Security Target Lite 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	[1]
	ATE_DPT.2	Testing: security enforcing modules	
	ATE_FUN.1	Functional testing	
	ATE_IND.2	Independent testing - sample	
Vulnerability Assessment	AVA_VAN.5	Advanced methodical vulnerability testing	[1]

7.2.1 Refinements

Some refinements are taken unchanged from [1] Table 15 provides an overview.

7.3 Security Requirements Rationale

7.3.1 Rationale for the Security Functional Requirements

The security functional requirements rationale of the TOE are defined and described in [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, 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 16: Rational for additional SFR in the ST Lite

Objective	TOE Security Functional Requirements
O.Add-Functions	- FCS_COP.1/3DES „Cryptographic operation"
	- FCS_COP.1/AES „Cryptographic operation"
	- FCS_COP.1/RSA „Cryptographic operation"
	- FCS_COP.1/ECDSA „Cryptographic operation"
	- FCS_COP.1/ECDH „Cryptographic operation"
	- FCS_CKM.1/RSA „Cryptographic key generation "
	- FCS_CKM.1/EC „Cryptographic key generation"
O.Phys-Manipulation	- FPT_TST.2 „ Subset TOE security testing "
O.Mem-Access	- FDP_ACC.1 "Subset access control"
	- FDP_ACF.1 "Security attribute based access control"
	- FMT_MSA.3 "Static attribute initialisation"
	- FMT_MSA.1 "Management of security attributes"
	- FMT_SMF.1 "Specification of Management Functions"
O.Malfunction	- FDP_SDI.1 „Stored data integrity monitoring"
	- FDP_SDI.2 „Stored data integrity monitoring and action"

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/RSA supports the generation of RSA keys, FCS_CKM.1/EC supports the generation of EC keys needed for this cryptographic operations. Therefore, FCS_COP.1/RSA, FCS_COP.1/ECDSA, FCS_COP.1/ECDH and FCS_CKM.1/RSA and FCS_CKM/EC are suitable to meet the security objective.

The use of the supporting libraries Toolbox and Base 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,

- FMT_MSA.2 Secure security attributes.

All these requirements have to be fulfilled to support OE.Resp-Appl for FCS_COP.1/3DES (3DES algorithm) and for FCS_COP.1/AES (AES algorithm). For the FCS_COP.1/RSA (RSA algorithm) and FCS_COP.1/ECDSA and FCS_COP.1/ECDH (both EC algorithms) the FCS_CKM.1/RSA and 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. Keys for RSA and EC algorithms can be provided either by the TOE or the environment.

In this ST Lite 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, FCS_CKM.4, and FMT_MSA.2 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 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_OSC Operating State Checking, SF_CRT Cryptographic Support and SF_TST Self Test.

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 the rationale already given in [1] 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 [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 and ROM. 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™. These measures detect and inform about one and more bit errors. In case of the SOLID FLASH™ 1 bit errors of the data are corrected automatically. The ECC mechanisms prevent the TOE from using corrupt data. Therefore FDP_SDI.2 is suitable to meet the security objective.

7.3.1.1 Dependencies of Security Functional Requirements

The dependencies of security functional requirements are defined and described in [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 and FAU_SAS.1.

The dependencies 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 17: Dependency for cryptographic operation requirement

Security Functional Requirement	Dependencies	Fulfilled by security requirements
FCS_COP.1/3DES	FCS_CKM.1	Yes, see comment 3
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) FCS_CKM.4	Yes, see comment 3
FCS_COP.1/AES	FCS_CKM.1	Yes, see comment 3
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) FCS_CKM.4	Yes, see comment 3
FCS_COP.1/RSA	FCS_CKM.1	Yes, see comment 3
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) FCS_CKM.4	Yes, see comment 3
FCS_CKM.1/RSA	FCS_CKM.2 or FCS_COP.1	Yes
	FMT_MSA.2 FCS_CKM.4	Yes, see comment 3
FCS_COP.1/ECDSA	FCS_CKM.1	Yes, see comment 3
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) FCS_CKM.4	Yes, see comment 3
FCS_CKM.1/EC	FCS_CKM.2 or FCS_COP.1	Yes
	FMT_MSA.2 FCS_CKM.4	Yes, see comment 3
FCS_COP.1/ECDH	FCS_CKM.1	Yes, see comment 3
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) FCS_CKM.4	Yes, see comment 3
FPT_TST.2	None	see comment 1
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 2
FMT_MSA.1	FDP_ACC.1 or FDP_IFC.1 FMT_SMR.1 FMT_SMF.1	Yes see comment 2 Yes
FMT_SMF.1	None	N/A
FDP_SDI.1	None	N/A
FDP_SDI.2	None	N/A

Comment 1:
The TOE is already a platform representing the lowest level in a Smartcard. There is no lower or »underlying abstract machine« used by the TOE which can be tested. Therefore, the former dependency to FPT_AMT.1 is fulfilled without further and by that dispensable. CC in the Revision 3 considered this and dropped this dependency. The requirement FPT_TST.2 is satisfied.
End of comment.

Comment 2:
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.
End of note comment.

Comment 3:
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

- FMT_MSA.2 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 [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.1/3DES and FCS_COP.1/AES 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 means, 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/ECDSA and FCS_COP.1/ECDH 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. The respective dependency FCS_CKM.1 has to be fulfilled by the TOE with the security functional requirement FCS_CKM.1/RSA (for FCS_COP.1/RSA) and 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_CKM.1/RSA and FCS_CKM.1/EC the respective dependency FCS_COP.1 is fulfilled by the TOE. The respective dependencies FMT_MSA.2 and FCS_CKM.4 have to be fulfilled by the environment. That means, the environment shall meet the

requirement FCS_CKM.4 as defined in [3], section 10.1 and the requirement FMT_MSA.2 as defined in [3], section 13.2.

The cryptographic libraries RSA and EC and the Toolbox library are delivery options. . If one of the libraries RSA, EC or Toolbox are delivered, the 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 Toolbox and Base Library are no cryptographic libraries and provide no additional specific security functionality.

 End of Comment.

7.3.2 Rationale of the Assurance Requirements

The chosen assurance level EAL4 and the augmentation with the requirements ALC_DVS.2, ATE_DPT.2 and AVA_VAN.5 were chosen in order to meet the assurance expectations explained in the following paragraphs. In Table 15 the different assurance levels are shown as well as the augmentations. The augmentations are in compliance with the Protection Profile.

An assurance level EAL4 with the augmentations ALC_DVS.2, ATE_DPT.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 [11] 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 EAL4 (which only requires ALC_DVS.1). ALC_DVS.2 has no dependencies.

ATE_DPT.2

This ST Lite claims strict conformance to [1], which requests the assurance class ATE_DPT.2.

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 AVA_VAN.5.

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 EAL4.

It has to be assumed that attackers with high attack potential try to attack Security ICs like smartcards 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. The Security Features are described below and the relation to the security functional requirements is shown.

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

Table 18: TOE Security Features

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

8.1 SF_DPM: Device Phase Management

The life cycle of the TOE is split up into several phases. Different operation modes help to protect the TOE during each phase of its lifecycle.

8.2 SF_PS: Protection against Snooping

The TOE uses various means to protect from snooping of memories and busses and prevents single stepping.

8.3 SF_PMA: Protection against Modifying Attacks

This TOE implements protection against modifying attacks of memories, alarm lines and sensors.

8.4 SF_PLA: Protection against Logical Attacks

Memory access of the TOE is controlled by a Memory Management Unit (MMU), which implements different priviledge levels. The MMU decides, whether access to a physical memory location is allowed based on the access rights of the privilege levels

8.5 SF_CS: Cryptographic Support

The TOE is equipped with an asymmetric and a symmetric hardware accelerator and also software modules to support several symmetric and asymmetric cryptographic operations. To provide random numbers according to meet FCS_RNG.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 the sections above. The results are shown in Table 19. The security functional requirements are addressed by at least one related security feature.

Table 19: 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	X	
FDP_ACF.1	X		X	X	
FPT_PHP.3	X	X	X	X	X
FDP_ITT.1	X	X	X	X	X
FDP_SDI.1			X		
FDP_SDI.2			X		
FDP_IFC.1		X	X	X	
FMT_MSA.1	X		X	X	
FMT_MSA.3	X		X	X	
FMT_SMF.1	X		X	X	
FRU_FLT.2			X		
FPT_ITT.1	X	X	X		X
FPT_TST.2			X		X
FPT_FLS.1		X	X	X	X
FCS_RNG.1					X
FCS_COP.1 /3DES					X
FCS_COP.1 /AES					X
FCS_COP.1 /RSA					X
FCS_COP.1 /ECDSA					X
FCS_COP.1 /ECDH					X
FCS_CKM.1 /RSA					X
FCS_CKM.1 /EC					X

8.7 Security Requirements are internally Consistent

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

In addition to the discussion of section 6.3 of [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.

The functional requirement FPT_TST.2 requires further protection to prevent manipulation of test results, while checking the security functions of the TOE..An attacker could aim to switch off or disturb certain sensors or filters and prevent the detection of distortion 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.

The implemented privilege level concept represents the area based memory access protection enforced by the MMU. As an attacker could attempt to manipulate the level concept as defined and present in the TOE, the functional requirement FDP_ACC.1 and the related other requirements have to be protected. The security functional requirements necessary 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.

9 References

9.1 Literature

[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 3 July 2009, CCMB-2009-07-001

[3]   Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements; Version 3.1 Revision 3 July 2009, CCMB-2009-07-002

[4]   Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements; Version 3.1 Revision 3 July 2009, CCMB-2009-07-003

[5]   Status report, List of all available user guidance

[6]   Functionality classes and evaluation methodology for physical random number generators AIS31, Version 1, 25.09.2001

[7]   SLx 70 Family Hardware Reference Manual , Infineon Technologies AG

[11]   Joint Interpretation Library, Application of Attack Potential to Smartcards, Version 2.7, April 2006

[12]   SLx 70 Family SLx 77 Product Group (L90FL Technology) Errata Sheet

[13]   SLE 70 Family Programmer's Reference User's Manual, Infineon Technologies AG,

[14]   NIST: FIPS publication 186-3: Digital Signature Standard (DSS), June 2009

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

[18]   SLx 70 Family Production and Personalization User's Manual

Note that the versions of these documents will be defined at the end of the evaluation and listed in the certification report.

10 Appendix

In Table 21 the hash signatures of the respective CL70 Crypto Library file are documented. For convenience purpose several hash values are referenced.

Table 20: Reference hash values of the CL70 Crypto Libraries

Library	Hash Value
Cl77-LIB-toolbox-XSMALL-HUGE.lib	
MD5	ad6957cb996ae91c497f36dd779966d7
SHA1	3456f98ea2f3ffb92cc6b4ce46da3e736f701a28
SHA256	9b4b23001cb400ea2bea9e8f34a870a5ff6541965ca568538c3adf0911b7229f
Cl77-LIB-base-XSMALL-HUGE.lib	
MD5	aad4d845f85995cb7398401f74c81436
SHA1	b2eb9a36a4b8b8c8c58b534ed33ad5ca79970a90
SHA256	bfebde15ec7856285a3ef5b47e2e124f33664f041c0ed08fb00793fd5c47841f
Cl77-LIB-2k-XSMALL-HUGE.lib	
MD5	173e460ee7a9b0aaefd601497228762b
SHA1	dc537705ef1f124ba05c57c103ab3d8fd7413059
SHA256	0aaf92505015f8d3ae902caf35c92234e2e2dfbba975ea7c893c7da23ba8d8db
Cl77-LIB-4k-XSMALL-HUGE.lib	
MD5	2d56f41542d202c8c76b37c02ea3b548
SHA1	6cf3d9c985775f2f6bb4ff58e4f022012c5d8c72
SHA256	374e4e6ded895b7b720045b16e62310245e3df53bffbcbcc71464345de728d23
Cl77-LIB-ecc-XSMALL-HUGE.lib	
MD5	5ba06504ff48113e4c4a8ed3e1cd651a
SHA1	1d900546d5613d8bd21ebf61dc2bed5e48584d91
SHA256	8064a397767f1d704b9713031b626b33cfe6b6a3c780de5d8435b091feb2511a

11 List of Abbreviations

AES	Advanced Encryption Standard
AIS31	"Anwendungshinweise und Interpretationen zu ITSEC und CC Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren"
APB™	Advanced Peripheral Bus
API	Application Programming Interface
ATR	Answer to Reset
AXI™	Advanced eXtensible Interface Bus Protocol
CC	Common Criteria
CI	Chip Identification Mode (STS-CI)
CIM	Chip Identification Mode (STS-CI), same as CI
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
EEPROM	Electrically Erasable and Programmable Read Only Memory
EMA	Electro magnetic analysis
Flash	Flash Memory
HW	Hardware
IC	Integrated Circuit
ICO	Internal Clock Oscillator
ID	Identification
IMM	Interface Management Module
ITP	Interrupt and Peripheral Event Channel Controller
I/O	Input/Output
IRAM	Internal Random Access Memory
ITSEC	Information Technology Security Evaluation Criteria
M	Mechanism
MED	Memory Encryption and Decryption
MMU	Memory Management Unit
O	Object
OS	Operating system

PEC	Peripheral Event Channel
PRNG	Pseudo Random Number Generator
PROM	Programmable Read Only Memory
RAM	Random Access Memory
RMS	Resource Management System
RNG	Random Number Generator
ROM	Read Only Memory
RSA	Rives-Shamir-Adleman Algorithm
SAM	Service Algorithm Minimal
SCP	Symmetric Cryptographic Processor
TSC	TOE Security Functions Control
TSF	TOE Security Functionality
UART	Universal Asynchronous Receiver/Transmitter
UM	User Mode (STS)
UMSLC	User mode Security Life Control
WDT	Watch Dog Timer
XRAM	eXtended Random Access Memory
3DES	Triple DES Encryption Standard

12 Glossary

Application Program/Data	Software which implements the actual TOE functionality provided for the user or the data required for that purpose
Central Processing Unit	Logic circuitry for digital information processing
Chip	Integrated Circuit]
Chip Identification Data	Data stored in the SOLID FLASH™ 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 STS version number
Chip Identification Mode	Operational status phase of the TOE, in which actions for identifying the individual chip by transmitting the Chip Identification Data take place
Controller	IC with integrated memory, CPU and peripheral devices
Cyclic Redundancy Check	Process for calculating checksums for error detection
Electrically Erasable and Programmable Read Only Memory (EEPROM)	Non-volatile memory permitting electrical read and write operations
End User	Person in contact with a TOE who makes use of its operational capability
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
Internal Random Access Memory	RAM integrated in the CPU
Mechanism	Logic or algorithm which implements a specific security function in hardware or software
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
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
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™ programming routines, AIS31 testbench etc.
Self Test Software	Part of the firmware with routines for controlling the operating state and testing the TOE hardware
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
Security Target Lite	Lite version of Security Target
SmartCard	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 Mode	Operational status phase of the TOE in which actions intended for the user takes place