IC Platform of
FeliCa Contactless Smartcard

CXD9861 / MB94RS402

Security Target
(Public Version)

Version 5, Level 3
Nov 22, 2006

Developed and provided by

System Micro Division, FUJITSU Ltd. and
SONY FeliCa Business Center

Document History

Version	Level	Description	Investigation	Approval
5	1	First release Author: Sep28, 2006, Y.Takasaki	Sep 28, 2006 M.Nakajima	Sep 28, 2006 T.Teramoto
5	2	Correction after internal review Author: Nov 13, 2006, Y.Takasaki	Nov 13, 2006 M.Nakajima	Nov 13, 2006 T.Teramoto
5	3	Correction after internal review Author: Nov 22, 2006, Y.Takasaki	Nov 22, 2006 H. Mori	Nov 22, 2006 T.Teramoto

Table of Contents

List of Figure

List of Table

1 ST Introduction

1.1 ST Identification

Title: IC Platform of FeliCa Contactless Smartcard CXD9861/ MB94RS402, Security Target (Public Version)

Date: 2006-11-22

Version: 5, Level: 3

Authors: FRAM Product Design Dept. System Micro Div. LSI Group Fujitsu ltd.

TOE identity

The TOE includes below;

- Smartcard Integrated Circuit "CXD9861/MB94RS402, FR00 001" ("FR00 001" means TOE version.)
- HAL-API version 22.0
- DRNG Library version 22.0

This Security Target has been built with the CC version2.3.

References for all evaluation evidences are listed in Section 9.3 References.

1.2 ST Overview

This document is the public version of Security Target for Smartcard Integrated Circuit CXD9861 (Fujitsu's original naming MB94RS402; replaced by customer's naming CXD9861), which is developed and produced by Fujitsu. The issued TOE in this Security Target consists of Hardware (IC Chip) in CXD9861 and IC Dedicated Software implemented in CXD9861. This software consists in HAL-API and DRNG Library.

This Security Target is produced in the frame of the security evaluation and certification of the CXD9861. These are conducted under the French IT Security Evaluation and Certification scheme, with the work of the DCSSI as Certification body and of CEACI / Thales as Evaluation laboratory (also called ITSEF).

The goal of this Security Target is to specify the functional and assurance requirements that are applicable to the CXD9861 as a Smartcard Integrated Circuit for smartcard applications.

CXD9861 is developed as Platform of Contactless Smartcard for communication purpose, which carries the application system for transportation and finance. It is in conformity with ISO/IEC18092 Passive Communication Mode of Contactless communication interface. CXD9861 also carries functions such as Timer, ROM, SRAM, FRAM, DES, Contactless RF interface, centring on the F^2MC-8FX CPU core. Furthermore, Non-volatile Ferroelectric Random Access Memory (FRAM) makes low power consumption and high-speed communication possible. Ideally suited to contactless and rapid transactions, the FRAM greatly improves the performances of the overall system.

In general, a Smartcard is usually seen as a credit card-sized card having a non-volatile memory and a processing unit embedded within it. Due to the build-in IC chip, Smartcard is capable to contain/process more information than magnetic card. For this reason, the use of Smartcard can be various from transportation (electronic ticket, commuter pass), finance (e-money, credit card), ID authentication (identification, company's ID card, management of working hours), to Marketing services (point card, shopping card, amusement card). In such a context, since IC chip stores the private / financial information that requires the protection against leakage or tampering, the demand for security becomes higher. Also, as Smartcard can be exposed to unspecified people in various circumstances, the attacker may possess powerful attack ability. Responding to such risks, Smartcard has to be equipped with the security functions against high-level attacks.

For this purpose, CXD9861 has to assure the confidentiality and integrity of stored data in IC chip and of the data transported by Reader/Writer device. In addition, stability and accuracy are required on the behavior of the IC chip. All this must also be carefully considered and assured throughout the whole development process of the IC chip (designing, production, test), as well as its delivery. For all above stages, the secure environment must be maintained in order to assure the confidentiality and integrity of IC chip and the data stored inside it.

The main objectives of this Security Target are to:

- Describe the Target of Evaluation (TOE) as a product and position it in the life cycle of the smartcard. The ST includes the development and the production phase of the IC Chip with its dedicated software, without the smartcard embedded software development phase, smartcard production phase and followings. These information is explained in Section2.
- Describe the security environment of the TOE including the assets to be protected and the threats to be countered by the TOE and by the operational environment during the development and production phases. We define the following assets and threats in Section3.
    - Assets; User data, the smartcard embedded software, pre-personalization data, design data, IC dedicated software, TSF data, initialization data, test and characterization related data.
    - Threats; side channel attacks (e.g. SPA, DPA, SEMA, DEMA), fault attack, probing, manipulation, malfunction, modification and exploitation of the TOE.
- Describe the security objectives for the TOE and for its environment in terms of integrity and confidentiality of application data and programs, protection of the TOE and associated documentation during the development and production phases. These information is explained in Section4
- Specify the security requirements that include the TOE security functional requirements and the TOE security assurance requirements in Section5.
- Describe the TSF in Section6, based on the threats, the objectives, the TOE security functional requirements and the TOE security assurance requirements.
- Describe PP claims in Section7. This Security Target conforms to [BSI-PP-0002].
- Describe rationale between threats, assumption, policy, objectives, security requirements and TSF, in order to show consistency in our security policy.

1.3 CC Conformance

This Security Target has been built with the CC version2.3, for Information Technology Security Evaluation (refer to [CC/1], [CC/2], [CC/3]).

- Part2 extended; Security Functional Requirements
- Part3 conformant; Security Assurance Requirements

Furthermore, this Security Target claims conformance to Protection Profile; BSI-PP-002 version1.0 (refer to [BSI-PP-0002]).

The level of Assurance is EAL4 augmented.

The EAL4 is augmented by taking the following components:

- ADV_IMP.2
- ALC_DVS.2
- AVA_MSU.3
- AVA_VLA.4

The minimum Strength Of Function of TOE Security Function is SOF-high.
(Strength of Functions High)

2 TOE Description

2.1 TOE Definition

The TOE is a single-chip microcomputer CXD9861, developed as Platform of Contactless Smartcard for communication purpose.

CXD9861 consists of Hardware (IC Chip) and IC Dedicated Software. This software consists in HAL-API and DRNG Library. Also, the TOE includes the Test features that are used at the IC testing during a phase of wafer manufacturing and are not available to user phases after IC testing.
Nevertheless, Smartcard Embedded Software (OS/Applications) is outside the scope of the TOE.

(Figure 1) demonstrates Block Diagram of CXD9861 as following.

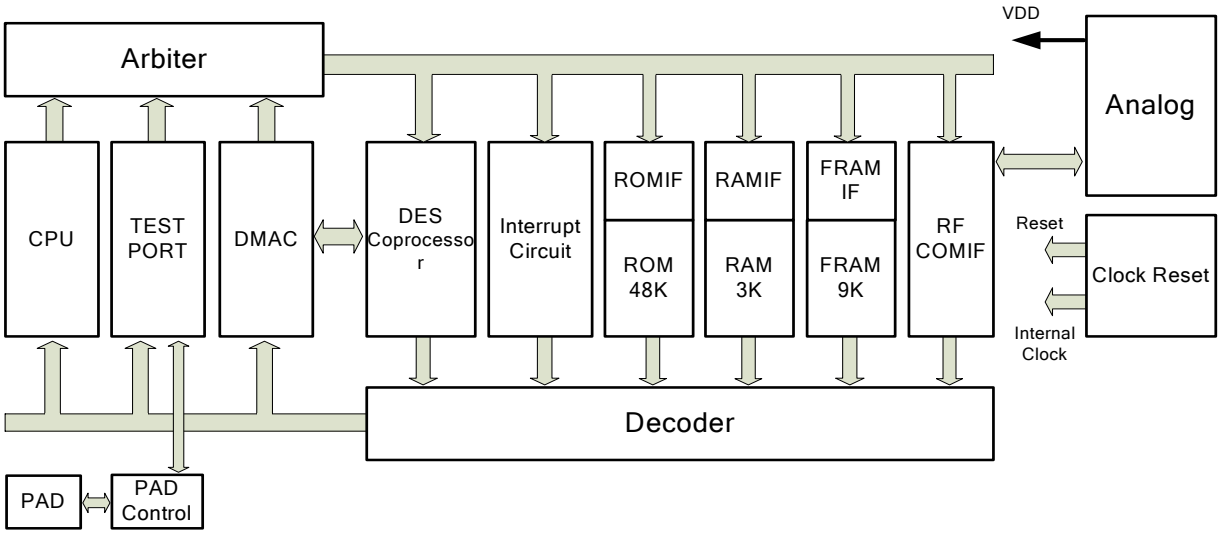

Figure 1 Internal Block Diagram of CXD9861

2.1.1 Hardware Description

Hardware of CXD9861 is a single-chip microcomputer that is developed for the Smartcard Integrated circuit with Contactless communication function. It is intended to be used for platform of application system, which carries the application system for transportation and finance. Centering on F^2MC8FX-CPU Core, CXD9861 consists of ROM, SRAM, FRAM, DES Co-processor, DMA controller, analog circuit (refers to Figure 1, Block Diagram).

CPU：F^2MC-8FX：

- 8-bit CISC
- Maximum CPU operating frequency of 6.78 MHz (during normal operation)
- Fixed-length 8-bit instructions (basic instructions), 1 instruction per cycle (minimum)
- Linear accessibility to 64-KB memory space

Memory：

- 48KB ROM
- 3KB SRAM
- 9KB FRAM nonvolatile memory

DES Coprocessor：

- Conformance to FIPS PUB 46-3
- Supporting the ECB and CBC modes and encryption and decryption.
- Supporting 1-key DES processing and 2-key triple-DES processing

DMA controller：

- Operable in coordination with the contactless communication logic circuit, supporting data transmission and data reception functions (the maximum data transfer length is of 256 bytes)
- Equipped with a circuit to support CRC calculation of Memory contents during transferring.

Analog circuit：

- Communication protocol: ISO/IEC 18092 Passive Communication Mode
- Carrier frequency: 13.56 MHz
- Supporting the communication speed: 212 kbps and 424 kbps
- Extracting the power from the carrier and supplying power and power-on reset to the chip
- ASK modulation and demodulation used for transmission and reception
- Sensors for detecting limit of outside of operating condition, which induces malfunction of the TOE hardware.

2.1.2 TOE Software Description

The TOE includes HAL-API and DRNG Library, as IC Dedicated Software. They are supplied by user in secure procedure and are stored in ROM. However, Smartcard Embedded Software (OS/Applications) is outside the scope of the TOE.

2.1.2.1 Hardware Abstraction Layer Application Program Interface (HAL-API) Description

HAL-API (; Hardware Abstraction Layer Application Program Interface) is, as IC Dedicated Software of CXD9861, a part of the TOE component, which is developed by IC Developer. Also, stored on ROM, HAL-API contains the function for Smartcard Embedded Software to facilitate the use of Hardware.

2.1.2.2 Deterministic Random Number Generator (DRNG) Library

DRNG Library is, as part of IC Dedicated Software of CXD9861, a security function of the TOE and contains a function to operate generation of random numbers.
DRNG Library generates deterministic random numbers conformed to AIS20, functionality class K3, strength of mechanism:high and algorithm is conformed to ANSIX9.42-2001 Annex C.2.

2.1.3 TOE Test Features

TOE Test features include IC Dedicated Test Software, Test circuits implemented in the TOE as parts of the TOE.
The IC Dedicated Test Software includes the Test-ROM and test programs, which are only used before IC Delivery and are not available to user phases.

2.1.4 Interface of TOE

The TOE interface is defined as followings:

- CXD9861's contactless RF communication antenna, which performs the communication with outside.
- CXD9861's HAL-API, which is used by Smartcard Embedded Software. (Details are in HAL-API function specification)
- CXD9861's DRNG library, which is used by Smartcard Embedded Software. (Details are in DRNG library specification)
- Assuming the TOE attack described in Section 3.3, Chip surface of TOE can be TOE interface.

2.2 Smartcard Product Life Cycle

Life Cycle of Smartcard products can be categorised to 7 phases as followings (refers to Figure 2 and [BSI-PP-0002, Section8.1.1]). Fujitsu is responsible for IC development and IC manufacturing in phase2 and phase3, including TOE traceability, interface / data exchange process between phase1 and phase2 and TOE delivery from phase3 to phase4.
Phase1, phase4 and following phases are outside of the scope of responsibility for Fujitsu.

Phase1: Smartcard Embedded Software Development

The Phase1 is outside the scope of TOE, and is managed by Smartcard Embedded Software developer.

Development of Smartcard Embedded Software (including pre-personalisation data) is performed on this phase. IC sensitive information, software and tools are delivered from IC developer to Smartcard Embedded Software developer.

Phase2: IC Development

In the Phase2, the IC developer is responsible for IC Design, IC Dedicated Software development, Mask manufacturing and Test program development.
Also, in this phase, IC sensitive information, software and tools are provided to Smartcard Embedded Software developer (on Phase1) by IC developer. Then, IC developer receives Smartcard Embedded Software and the pre-personalization data from Smartcard Embedded Software developer (on Phase1).

In Mask manufacturing, photomask databases generated by IC developer in Fujitsu are delivered to Mask manufacturer, which is subcontractor of Fujitsu. Then, photomasks are manufactured in the secure environment and delivered to IC manufacturer (on Phase3) by secure procedure.

Phase3: IC Manufacturing

In the Phase3, IC Manufacturer is responsible for IC manufacturing, IC Testing, injecting pre-personalization data, bump assembling and dicing, and TOE delivery.
Also, in this phase, the photomask is delivered from Mask manufacturer by secure procedure and wafer fabrication starts for manufacturing secure products.

After the wafer fabrication, IC testing is performed in order to assure conformance with the device specification. During the IC testing, TOE identification data (Chip manufacturing information) and pre-personalisation data that includes customer's confidential data are injected into the TOE during the IC testing. After the IC testing, bump assembling and wafer dicing are performed in this phase.

At the end of the Phase3, Test features (including IC dedicated test software, test circuits) are deactivated, in order to avoid that attacker exploits the TOE.

After Phase3, TOE (IC chips) is delivered to Smartcard Product manufacturer (Phase4 and followings) by trusted delivery and verification procedure. Fujitsu is responsible for managing from Phase2 to TOE Delivery after Phase3.

Phase4 and Phase5: Smartcard production

These phases are outside the scope of TOE. In these phases, the smartcard is produced at a smartcard manufacturing facility.
These phases include IC packaging, testing module, and incorporation of module into the plastic card body, and the IC Packaging Manufacturer and the Smartcard Product Manufacturer are responsible for those things.

Phase6: Smartcard personalization

This phase is the final step necessary to prepare the smartcard for issue to users consists of personalisation of smartcard.
The Personaliser is responsible for the above things.

Phase7: End-user

This phase is the end-user phase where the smartcard is issued to end-users for operational deployment.
The end-user phase contains also the end of life process of the smartcard, which is critical aspect in the life cycle.
The Smartcard Issuer is responsible for the above things.

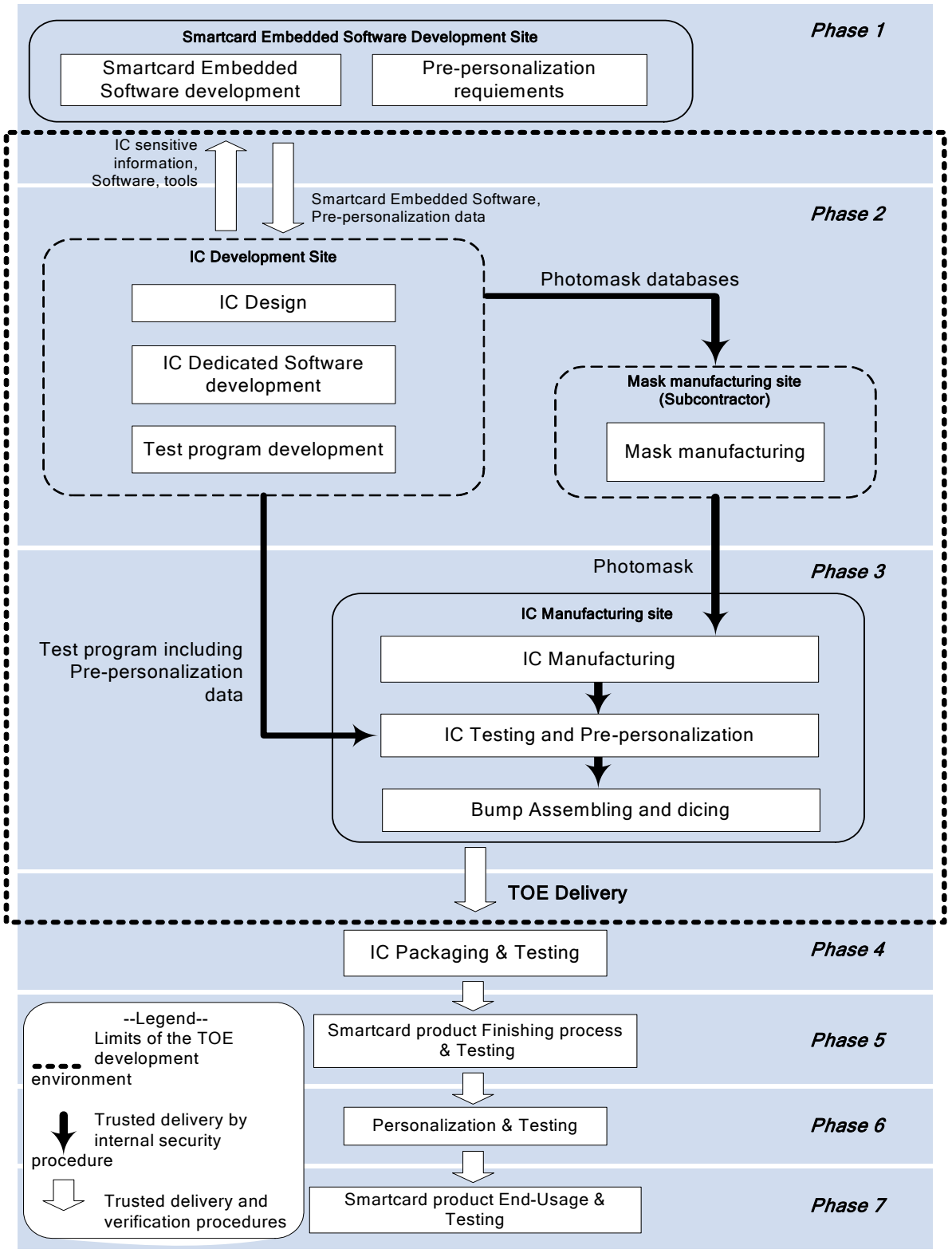

Figure 2 Smartcard product life cycle

2.3 TOE Environment

The TOE environment related with TOE development and production corresponds to Phase2 and Phase3 (refer to Section 2.2). Then there are four identified sites in the TOE life cycle, as evaluation scope (refer to Figure 2, as below).

- IC development site in pshase2
- Mask manufacturing site in phase2
- IC Manufacturing site in phase3

2.3.1 Environment of IC Development Site

In IC development site, Operations, as shown in Figure 2, are performed in secure environment. The site is controlled by the following secure procedures.

- To assure security, the environment in which the development takes place shall be made secured with controllable accesses having traceability. Furthermore, it is important that all authorized personnel involved fully understand the importance and the rigid implementation of defined security procedures.
- The development begins with the TOE's specification. All parties in contact with sensitive information are required to abide by Non-Disclosure agreement.
- Network in these sites is stand-alone. Therefore, network system of these sites is disconnected to Fujitsu-Wan by physical and logical procedures.
- The engineer uses a secure computer system (preventing unauthorized access) to make his design simulation, circuit performance verifications, generation of test specification, test pattern and test program and generation of TOE's IC photomask databases.
- Sensitive documents, databases on tapes, diskettes, and printed circuit layout information are stored in appropriate locked cupboards/safe. Of paramount importance also is the disposal of unwanted data (complete electronic erasure) and documents (e.g. shredding).

The data, which shall maintain integrity and confidentiality, includes followings;

- Smartcard embedded software and pre-personalization data
- Logical design data and physical design data
- Mask data
- IC dedicated software
- Test specification and Test data
- Test program
- Documentation

2.3.2 Environment of Mask Manufacturing Site

In Mask manufacturing site, Operations, as shown in Figure 2, are performed in secure environment. This site is controlled by the following secure procedures.

- Photomasks are manufactured in subcontractor of Fujitsu by subcontractor's secure procedure. They are transported and worked on in a secure environment with accountability and traceability of all (good and bad) products.
- During the transfer of sensitive data electronically, procedures shall be established to ensure that the data arrive only at the destination and are not accessible at intermediate stages.

The data, which shall maintain integrity and confidentiality, includes followings;

- Photomask databases

- Photomask (Reticle)

2.3.3 Environment of IC Manufacturing Site

In IC Manufacturing site, Operations, as shown in Figure 2, are performed in secure environment. This site is controlled by the following secure procedures.

- By access control system, secure environment is established in the IC manufacturing environment. Also, transfer of wafer between each process is operated by secure procedures. Furthermore the only authorized persons can work for secure products. The all authorized persons fully understand the importance and the rigid wafer processing of defined security procedures.
- After fabrication, IC testing is operated in secure environment, same as IC manufacturing. Also, test program and test tools are securely protected and operated by the only authorized staff. Furthermore, TOE identification data and the pre-personalization data are injected to TOE in secure environment during the IC testing.
- After IC testing, bump assembling and wafer dicing are performed in secure environment, same as IC manufacturing. After that, TOE is packed up and is delivered to the smartcard product manufacturer by trusted delivery and verification procedure. Non-functional and bad ICs, which are separated from functional and good ICs, are discarded in a controlled accountable manner.
- On the each process in IC manufacturing site, tracking system maintains traceability of the TOE.

The data, which shall maintain integrity and confidentiality, includes followings;

- Product
- Test program, and test and characterisation related data
- Initialisation Data and Pre-personalisation Data

2.4 TOE Intended Usage

2.4.1 Smartcard Intended Usage

Contactless Smartcard for communication purpose, which includes CXD9861, is intended to be used for financial settlement, transportation, personal identification and distribution service. For such purposes, Smartcard stores personal/monetary information, which requires the protection against leakage and tampering. Following list demonstrates the possible use of Smartcard.

- Finance/Settlement purposes: E-money card on pre-paid format and as credit card are imaginable

- Transportation purposes: E-transportation tickets/pass on pre-paid format are imaginable.

- Identification purposes: Personal identification/Company identification/Entry or Exit control are imaginable.

- Marketing service purposes: Point card/shopping card/amusement card are imaginable

For the uses above, Smartcard is expected to provide the security features and to be used effectively in various purposes to improve the service for users.

2.4.2 TOE IT Security features

In order to assure the confidentiality and integrity of TOE, CXD9861 carries several functions as followings:
- DES function secure against the Side-Channel Attack/DFA (Differential Fault Attack)/Timing Attack
- Sensor against the malfunction induced by environmental stress (temperature/frequency/voltage)
- DRNG Library which supports generation of deterministic random numbers.
- Hardware Security Function against Physical Manipulation/Physical Probing

2.4.3 TOE Related Users

Followings are possible personnel, who are related to Life Cycle of Smartcard (refers to Section2.2 and [BSI-PP-0002, Section8.1.1]).

TOE User
- Smartcard Embedded Software developer

TOE Administrator
- IC Developer
- IC Manufacturer
- IC Packaging Manufacturer
- Smartcard Product manufacturer
- Personaliser
- Smartcard Issuer

3 TOE Security Environment

3.1 Description of Assets

This section defines the primary and secondary assets to be protected by the TOE. The following assets are derived from [BSI-PP-0002 section3.1].

Assets regarding the Threats

The primary assets to be protected

- The User Data:
  This includes especially Cryptographic keys, personalisation data and other data generated and used by the Smartcard Embedded Software.

- The Smartcard Embedded Software, comprising followings;

  - Hard-coded Smartcard Embedded Software (stored in ROM)

  - Soft-coded Smartcard Embedded Software (stored in FRAM)

The further primary assets to be protected

- The correct operation of TOE (including its Random Number Generator).

  In particular this means that Smartcard Embedded Software is correctly being executed which includes the correct operation of the TOE's functions.

- The random numbers generated by the TOE.

The secondary assets include logical design data, physical design data, IC Dedicated Software and TSF data that are data created and used by the TSF.

In addition, the following will also contain information about the TOE.

- Initialisation Data and Pre-personalisation Data, specific development aids, test and characterisation related data, material for software development support, and photomasks.

Such information and the ability to perform manipulations assist in threatening the above primary assets.

Note that there are many ways to manipulate or disclose the User Data. (i) An attacker may manipulate the Smartcard Embedded Software or the TOE. (ii) An attacker may cause malfunctions of the TOE or abuse Test Features provided by the TOE. Such attacks usually require design information of the TOE to be obtained.
Therefore, the design information is a secondary asset. They pertain to all information about (i) the circuitry of the IC (hardware including the physical memories), (ii) the IC Dedicated Software and (iii) the TSF data.

[1] The pre-personalization data includes confidential data of customer.

Assets regarding the Organisational Security Policy P.Process-TOE

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, Smartcard Embedded Software, Initialisation Data and Pre-personalisation Data,
- Specific development aids,
- Test and characterisation related data,
- Material for software development support, and
- Photomasks and products in any form,

as long as they are generated, stored, or processed by the TOE Manufacturer.
Explanations can be found in [BSI-PP-0002, Section 8.1.2]

Assets regarding the Assumption A.Process-Card

The information and material produced and/or processed by the Smartcard Embedded Software Developer in Phase 1 and by the Card Manufacturer can be grouped as follows:

- The Smart Card Embedded Software including specifications, implementation and related documentation,
- Pre-personalisation and personalisation data including specifications of formats and memory areas, test related data,
- The User Data and related documentation, and
- Material for software development support,

as long as they are not under the control of the TOE Manufacturer.

3.2 Assumptions

In this section, the assumptions are described according to [BSI-PP-0002, Section 3.2].

The intended usage of the TOE is twofold, depending on the Life Cycle Phase:
(i) The Smartcard Embedded Software developer uses it as a platform for the smartcard software being developed. (ii) The Card Manufacturer (and the end-user) uses it as a part of the Smartcard. The Smartcard is used in a terminal that supplies the card (with power and clock) and (at least) mediates the communication with the Smartcard Embedded Software.

The following Assumptions are derived from [BSI-PP-0002, Section 3.2].

Appropriate “Protection during Packaging, Finishing and Personalisation (A. Process-Card)” must be ensured after TOE Delivery up to the end of Phase 6, as well as during the delivery to Phase 7 as specified below.

A.Process-Card Protection during Packaging, Finishing and Personalisation

It is assumed that security procedures are used after delivery of the TOE by the TOE Manufacturer up to delivery to the end-user 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).

This means that the Phases after TOE Delivery (refer to *Sections 2.2*) are assumed to be protected appropriately. For a preliminary list of assets to be protected, see *Section3.1*

The developer of the Smartcard Embedded Software must ensure the appropriate “Usage of Hardware Platform (A.Plat-Appl)” while developing this software in Phase1 as specified below.

A.Plat-Appl Usage of Hardware Platform

The Smartcard Embedded Software is designed so that the requirements from the following documents are met:

(i) LSI specification;[SPC], HAL-API function specification;[HAL], DRNG library specification;[DRN] of CXD9861 and the hardware application notes,
(ii) Findings of the TOE evaluation reports relevant for the Smartcard Embedded Software.

Note1: “Security Requirements” is described in Section4 of [HAL]. It is required that software developer shall develop the smartcard embedded software according to the “Security Requirements”, in order to maintain TSF operation securely.

Note2: note that particular requirements for the Smartcard Embedded Software are often not clear before considering a specific attack scenario during vulnerability analysis of the smartcard integrated circuit

(AVA_VLA). Therefore, such results from the TOE evaluation (as contained in the Evaluation Technical Report (ETR)) must be given to the developer of the Smartcard Embedded Software in an appropriate and authorised form and be taken into account during the evaluation of the software. This may also hold for additional tests being required for the combination of hardware and software. The TOE evaluation must be completed before evaluation of the Smartcard Embedded Software can be completed. The TOE evaluation can be conducted before and independent from the evaluation of the Smartcard Embedded Software.

The developer of the Smartcard Embedded Software must ensure the appropriate "Treatment of User Data (A.Resp-Appl)" while developing this software in Phase1 as specified below.

A.Resp-Appl Treatment of User Data

All User Data are owned by Smartcard Embedded Software. Therefore, it must be assumed that security relevant User Data (especially cryptographic keys) are treated by the Smartcard Embedded Software as defined for the specific application context.

Examples of embedded software security concerns are given in [BSI-PP-0002, Section8.2]

Additional Assumptions

The following Assumption is derived from [SSVG Augmentation].

The developer of Smartcard Embedded Software must ensure the appropriate "Usage of key-dependent Functions (A.Key-Function)" while developing this software in Phase1 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 comprise 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 and (ii) the processing of User Data including cryptographic keys.

3.3 Threats

In this section, the threats are described according to [BSI-PP-0002, Section 3.3].

The TOE has the following high-level security concerns:

SC1 manipulation of User Data and of the Smartcard Embedded Software (while being executed / processed and while being stored in the TOE's memories) and

SC2 disclosure of User Data and of the Smartcard Embedded Software (while being processed and while being stored in the TOE's memories).

SC3 deficiency of random numbers.

These high-level security concerns are refined below by defining threats as required by Common Criteria. Note that manipulation of the TOE is only a means to threaten User Data or the Smartcard Embedded Software and is not a success for the attacker in itself.

The Smartcard Embedded Software must contribute to averting the threats: At least it must not undermine the security provided by the TOE. For detail refer to the assumptions regarding the Smartcard Embedded Software specified in Section3.2.

These security concerns are derived from considering the end-usage phase (Phase 7) since

- Phase1 and the Phases from TOE Delivery up to the end of Phase 6 are covered by assumptions and

- the development and production environment starting with Phase 2 up to TOE Delivery are covered by an organisational security policy.

The TOE's countermeasures are designed to avert the threats described below. Nevertheless, they may be effective in earlier phases (Phases 4 to 6).

The TOE is exposed to different types of influences or interactions with its outer world. Some of them may result from using the TOE only but others may also indicate an attack. The different types of influences or interactions are visualised in Figure 3, which is modified [BSI-PP-0002, Section 3.3, Figure 8] to meet this TOE.

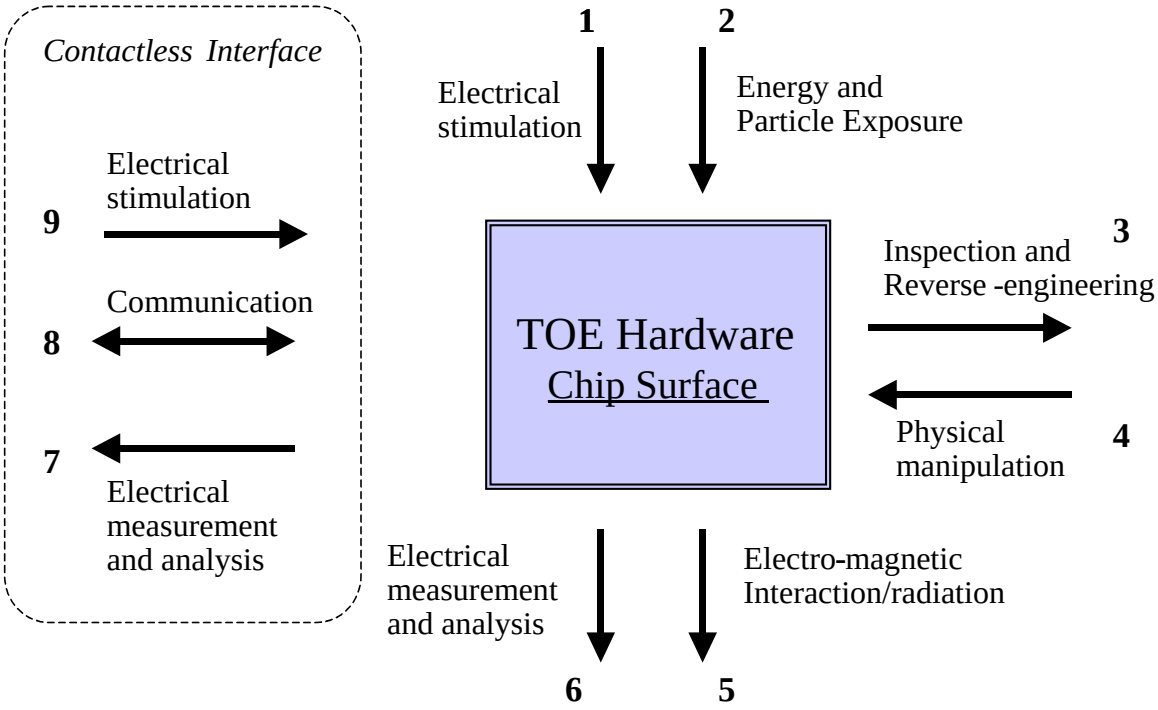

Figure 3 Attack Model for the TOE [2]

An interaction with the TOE can be done through the contactless interface (Number 7 – 9 in Figure 3). Influences or interactions with the TOE also occur through the chip surface (Number 1 – 6 in Figure 3). In Number 1 and 6, galvanic contacts are used. In Number 2 and 5, the influence (arrow directed to the chip) or the measurement (arrow starts from the chip) does not require a contact. Number 3 and 4 refer to specific situations where the TOE and its functional behaviour is not only influenced but definite changes are made by applying mechanical, chemical and other methods (such as 1, 2). Many attacks require a prior inspection and some reverse engineering (Number 3).

Examples for specific attacks are given in [BSI-PP-0002, Section 8.3]

[2] Figure 3 was modified [BSI-PP-0002, Section3.3 Figure 8] to meet the TOE.

The Threats derived from SC1 ～ SC3

The TOE shall avert the threat “Inherent Information Leakage (T.Leak-Inherent)” as specified below.

T.Leak-Inherent Inherent Information Leakage

An attacker may exploit information that is leaked from the TOE during usage of the Smartcard in order to disclose confidential data (User Data or TSF data).

No direct contact with the Smartcard internals is required here. Leakage may occur through emanations, variations in power consumption, I/O characteristics, clock frequency, or by changes in processing time requirements. Some examples are the Differential and Simple Power Attack (DPA, SPA), Differential and Simple Electro Magnetic Attack (DEMA, SEMA)[3]. These leakages may be interpreted as a covert channel transmission but are more closely related to measurement of operating parameters, which may be derived either from direct (contact) measurements (Numbers 6 and 7 in Figure 3) or measurement of emanations (Number 5 in Figure 3) and can then be related to the specific operation being performed.

The TOE shall avert the threat “Physical Probing (T.Phys-Probing)” as specified below.

T.Phys-Probing Physical Probing

An attacker may perform physical probing of the TOE in order to:

(i) Disclose User Data,
(ii) Disclose/reconstruct the Smartcard Embedded Software or
(iii) Disclose other critical operational information especially TSF data.

Physical probing requires direct interaction with the Smartcard Integrated Circuit internals (Numbers 5 and 6 in Figure 3). Techniques commonly employed in IC failure analysis and IC reverse engineering efforts may be used. Before that hardware security mechanisms and layout characteristics need to be identified (Number 3 in Figure 3). Determination of software design including treatment of User Data may also be a pre-requisite.

This pertains to “measurements” using galvanic contacts or any type of charge interaction whereas manipulations are considered under the threat “Physical Manipulation (T.Phys-Manipulation)”. The threats “Inherent Information Leakage (T.Leak-Inherent)” and “Forced Information Leakage (T.Leak-Forced)“ may use physical probing but require complex signal processing in addition.

[3] Differential and Simple Electro Magnetic Analysis (DEMA, SEMA) were added to this sentence to meet the TOE.

The TOE shall avert the threat “Malfunction due to Environmental Stress (T.Malfunction)” as specified below.

T.Malfunction Malfunction due to Environmental Stress

An attacker may cause a malfunction of TSF or of the Smartcard Embedded Software by applying environmental stress in order to:

(i) Deactivate or modify security features or functions of the TOE or
(ii) Deactivate or modify security functions of the Smartcard Embedded Software. This may be achieved by operating the Smartcard outside the normal operating conditions (Numbers 1, 2 and 9 in Figure 3).

To exploit this an attacker needs information about the functional operation.

The TOE shall avert the threat “Physical Manipulation (T.Phys-Manipulation)” as specified below.

T.Phys-Manipulation Physical Manipulation

An attacker may physically modify the Smartcard in order to:

(i) Modify security features or functions of the TOE,
(ii) Modify security functions of the Smartcard Embedded Software or
(iii) Modify User Data.

The modification may be achieved through techniques commonly employed in IC failure analysis (Numbers 1, 2 and 4 in Figure 3) and IC reverse engineering efforts (Number 3 in Figure 3). The modification may result in the deactivation of a security function. Before that hardware security mechanisms and layout characteristics need to be identified. Determination of software design including treatment of User Data may also be a pre-requisite. Changes of circuitry or data can be permanent or temporary.

In contrast to malfunctions (refer to T.Malfunction) the attacker requires to gather significant knowledge about the TOE’s internal construction here (Number 3 in Figure 3).

The TOE shall avert the threat “Forced Information Leakage (T.Leak-Forced)” as specified below.

T.Leak-Forced Forced Information Leakage

An attacker may exploit information that is leaked from the TOE during usage of the Smartcard in order to disclose confidential data (User Data or TSF data) even if the information leakage is not inherent but caused by the attacker.

This threat pertains to attacks where methods described in “Malfunction due to Environmental Stress” (refer to T.Malfunction) and/or “Physical Manipulation” (refer to T.Phys-Manipulation) are used to cause leakage from signals (Numbers 5, 6, 7 and 8 in Figure 3) that normally do not contain significant information about secrets.

The TOE shall avert the threat “Abuse of Functionality (T.Abuse-Func)” as specified below.

T.Abuse-Func Abuse of Functionality

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

(i) Disclose or manipulate User Data,
(ii) Manipulate (explore, bypass, deactivate or change) security features or functions of the TOE or of the Smartcard Embedded Software or
(iii) Enable an attack.

The TOE shall avert the threat “Deficiency of Random Numbers (T.RND)” as specified below.

T.RND Deficiency of Random Numbers

An attacker may predict or obtain information about random numbers generated by the TOE for instance because of a lack of entropy of the random numbers provided.

An attacker may gather information about the produced random numbers which might be a problem because they may be used for instance to generate cryptographic keys.

Here the attacker is expected to take advantage of statistical properties of the random numbers generated by the TOE without specific knowledge about the TOE’s generator. Malfunctions or premature ageing are also considered which may assist in getting information about random numbers.

Also, an attacker may crack the cryptographic keys by analyzing the rules of leaked random numbers during the wireless communication of Smartcard.[4]

[4] This sentence was added to [BSI-PP-0002, Section 3.3] to meet the TOE.

Additional Threats

The following additional threat assumes attack against exploiting or modifying FRAM data.

T.Memory-Integrity Memory Integrity of FRAM

An attacker with high motivation and resources as well as a good knowledge of IC process (and of embedded software development), can try to exploit FRAM sensitivity to environmental conditions in order to modify FRAM contents (both TSF and user data).

T.Memory-Access Memory Access to FRAM

An attacker with high motivation and resources as well as a good knowledge of embedded software development, can try to bypass the memory control access policy and gain illegal access to the user data and TSF data, which are stored in protected FRAM area.

3.4 Organizational Security Policies

The following policy is derived from [BSI-PP-0002, Section 3.4].

The IC Developer/Manufacturer must apply the policy “Protection during TOE Development and Production (P.Process-TOE)” as specified below.

P.Process-TOE Protection during TOE Development and Production

The TOE Manufacturer must ensure that the development and production of the Smartcard Integrated Circuit (Phase 2 up to TOE Delivery, refer to Section2.2) is secure so that no information is unintentionally made available for the operational phase of the TOE. For example, the confidentiality and integrity of design information and test data shall be guaranteed; access to samples, development tools and other material shall be restricted to authorised persons only; scrap will be destroyed etc. This not only pertains to the TOE but also to all information and material exchanged with the developer of the Smartcard Embedded Software and therefore especially to the Smartcard Embedded Software itself. This includes the delivery (exchange) procedures for phase 1 and the phases after TOE Delivery as far as they can be controlled by the TOE Manufacturer.

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

Additional Policies

The following Policy is derived from [SSVG Augmentation].

The TOE provides specific security functionality that 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:

- Data Encryption Standard (DES)
- Triple Data Encryption Standard (3DES)

4 Security Objectives

4.1 Security Objectives for the TOE

In this section, the security objectives for TOE are described according to [BSI-PP-0002, Section 4.1].

The product supports the following high-level security goals:

SG1 maintain the integrity of User Data and of the Smartcard Embedded Software (when being executed / processed and when being stored in the TOE's memories) as well as

SG2 maintain the confidentiality of User Data and of the Smartcard Embedded Software (when being processed and when being stored in the TOE's memories).

SG3 provide random numbers.

These standard high-level security goals are refined below by defining security objectives, as required by the Common Criteria. Note that the integrity of the TOE is a means to reach these objectives.

The Security Objectives derived from SG1 ～ SG3

The TOE shall provide "Protection against Inherent Information Leakage (O.Leak-Inherent)" as specified below.

O.Leak-Inherent Protection against Inherent Information Leakage

The TOE must provide protection against disclosure of confidential data (User Data or TSF data) stored and/or processed in the Smartcard IC.

- by measurement and analysis of the shape and amplitude of signals (for example on the power, clock, or I/O lines) and
- by measurement and analysis of the time between events found by measuring signals (for instance on the power, clock, or I/O lines) and
- by measurement and analysis of the electro magnetic emanations of TOE.[5]

This objective pertains to measurements with subsequent complex signal processing whereas O.Phys-Probing is about direct measurements on elements on the chip surface. Details correspond to an analysis of attack scenarios that is not given here.

The TOE shall provide "Protection against Physical Probing (O.Phys-Probing)" as specified below.

[5] This sentence was added to [BSI-PP-0002, Section 4.1] to meet the TOE.

O.Phys-Probing Protection against Physical Probing

The TOE must provide protection against disclosure of User Data, against the disclosure/reconstruction of the Smartcard Embedded Software or against the disclosure of other critical operational information. This includes protection against

- Measuring through galvanic contacts which is direct physical probing on the chips surface except on pads being bonded (using standard tools for measuring voltage and current) or

- Measuring not using galvanic contacts but other types of physical interaction between charges (using tools used in solid-state physics research and IC failure analysis)

With a prior

- Reverse-engineering to understand the design and its properties and functions.

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

The TOE shall provide “Protection against Malfunctions (O.Malfunction)” as specified below.

O.Malfunction Protection against Malfunctions

The TOE must ensure its correct operation.

The TOE must prevent its operation outside the normal operating conditions where reliability and secure operation has not been proven or tested. This is to prevent errors. The environmental conditions may include voltage, clock frequency, temperature, or external energy fields.

Remark: A malfunction of the TOE may also be caused using a direct interaction with elements on the chip surface. This is considered as being a manipulation (refer to the objective O.Phys-Manipulation) provided that detailed knowledge about the TOE’s internal construction is required and the attack is performed in a controlled manner.

The TOE shall provide “Protection against Physical Manipulation (O.Phys-Manipulation)” as specified below.

O.Phys-Manipulation Protection against Physical Manipulation

The TOE must provide protection against manipulation of the TOE (including its software and TSF data), the Smartcard Embedded Software and the User Data. This includes protection against

- Reverse-engineering (understanding the design and its properties and functions),

- Manipulation of the hardware and any data, as well as

- Controlled manipulation of memory contents (User Data).

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

The TOE shall provide “Protection against Forced Information Leakage (O.Leak-Forced)“ as specified below:

O.Leak-Forced Protection against Forced Information Leakage

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

- by forcing a malfunction (refer to “Protection against Malfunction due to Environmental Stress (O.Malfunction)” and/or

- by a physical manipulation (refer to “Protection against Physical Manipulation (O.Phys-Manipulation)”.

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

The TOE shall provide “Protection against Abuse of Functionality (O.Abuse-Func)” as specified below.

O.Abuse-Func Protection against Abuse of Functionality

The TOE must prevent those functions of the TOE that may not be used after TOE Delivery can be abused in order to

(i) Disclose critical User Data,
(ii) Manipulate critical User Data of the Smartcard Embedded Software,
(iii) Manipulate Soft-coded Smartcard Embedded Software or
(iv) Bypass, deactivate, change or explore security features or functions of the TOE.

Details depend, for instance, on the capabilities of the Test Features provided by the IC Dedicated Test Software that are not specified here.

The TOE shall provide “TOE Identification (O.Identification)“ as specified below:

O.Identification TOE Identification

The TOE must provide means to store Initialisation Data and Pre-personalisation Data in its non-volatile memory. The Initialisation Data (or parts of them) are used for TOE identification.

The TOE shall provide “Random Numbers (O.RND)” as specified below.

O.RND Random Numbers

The TOE will ensure the cryptographic quality of random number generation. For instance, random numbers shall not be predictable and shall have sufficient entropy.

The TOE will ensure that no information about the produced random numbers is available to an attacker since they might be used for instance to generate cryptographic keys.

Additional Objectives 1

The following Policy is derived from [SSVG Augmentation].

The TOE shall provide “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:

- Data Encryption Standard (DES)
- Triple Data Encryption Standard (3DES)

Additional Objectives 2

The following additional threat objective assumes protection against that FRAM data is exploited or modified.

O.Memory-Integrity Memory Integrity of FRAM

The TOE must provide protection against exploitation of FRAM sensitivity to environmental conditions by an attacker with high motivation and resources as well as a good knowledge of IC process (and of embedded software development), in order to protect modification of FRAM contents (both TSF and user data).

O.Memory-Access Memory Access to FRAM

The TOE must provide protection against bypassing the memory control access policy and gaining illegal access to the user data and TSF data, which are stored in protected FRAM area, by an attacker with high motivation and resources as well as a good knowledge of embedded software development.

4.2 Security Objectives for Environment

In this section, the security objectives for Environment are described according to [BSI-PP-0002, Section 4.2]. Then some clarifications derived from “O.Add-Functions» are added according to [SSVG Augmentation].

Phase 1

The Smartcard Embedded Software shall provide “Usage of Hardware Platform (OE.Plat-Appl)” as specified below.

OE.Plat-Appl Usage of Hardware Platform

To ensure that the TOE is used in a secure manner the Smartcard Embedded Software shall be designed so that the requirements from the following documents are met:

(i) LSI specification;[SPC], HAL-API function specification;[HAL], DRNG library specification;[DRN] of CXD9861 and the TOE application notes,
(ii) findings of the TOE evaluation reports relevant for the Smartcard Embedded Software.

Note: Software developer shall develop the smartcard embedded software according to “Security Requirements” which is described in Section4 of [HAL].

Clarification of “Usage of Hardware Platform (OE.Plat-Appl)”

The TOE supports cipher schemes as additional specific security functionality (as in O.Add-Functions). 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 Smartcard Embedded Software shall provide “Treatment of User Data (OE.Resp-Appl)” as specified below.

OE.Resp-Appl Treatment of User Data

Security relevant User Data (especially cryptographic keys) are treated by the Smartcard Embedded Software as required by the security needs of the specific application context. For example the Smartcard Embedded Software will not disclose security relevant user data to unauthorised users or processes when communicating with a terminal.

Clarification of “Treatment of User Data (OE.Resp-Appl)”

The TOE supports cipher schemes as additional specific security functionality (as in O.Add-Functions).

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 practically 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.

Phase 2 up to TOE Delivery

The TOE Manufacturer shall ensure the “Protection during TOE Development and Production (OE.Process-TOE)” as specified below.

OE.Process-TOE Protection during TOE Development and Production

The TOE Manufacturer must ensure that the development and production of the Smartcard Integrated Circuit (Phases 2 and 3 up to TOE Delivery, refer to Section2.2) is secure so that no information is unintentionally made available for the operational phase of the TOE. For example, the confidentiality and integrity of design information and test data must be guaranteed, access to samples, development tools and other material must be restricted to authorised persons only, scrap must be destroyed. This not only pertains to the TOE but also to all information and material exchanged with the developer of the Smartcard Embedded Software and therefore especially to the Smartcard Embedded Software itself. This includes the delivery (exchange) procedures for Phase 1 and the Phases after TOE Delivery as far as they can be controlled by the TOE Manufacturer.

An accurate identification must be established for the TOE. This requires that each instantiation of the TOE carry this unique identification. In order to make this practical, electronic identification shall be possible.

TOE Delivery up to the end of Phase 6

Appropriate “Protection during Packaging, Finishing and Personalisation (OE.Process-Card)” must be ensured after TOE Delivery up to the end of Phases 6, as well as during the delivery to Phase 7 as specified below.

OE.Process-Card Protection during Packaging, Finishing and Personalisation

Security procedures shall be used after TOE Delivery up to delivery to the end-user to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorized use).

This means that Phases after TOE Delivery up to the end of Phase 6 (refer to Section2.2) must be protected appropriately.

5 IT Security Requirements

5.1 TOE Security Requirements

5.1.1 TOE Functional Requirements

Following Security Functional Requirements are derived from CC Part2.
As for FCS_RND.1, FMT_LIM.1, FMT_LIM.2, FAU_SAS.1, these requirements are extended CC Part2, which is newly added for Smartcard IC. The details are defined in [BSI-PP-0002, Section8.4～6].

The strength of function of SFRs conforms to FCS_RND.1.
Followings are the SFRs used in this Security Target.

Security Functional Requirement		Requirements
Component	Component name	
FRU_FLT.2	Limited fault tolerance	Addressing the robustness within some limit before active reaction takes place.
FPT_FLS.1	Failure with preservation of secure state	Correct detection of outside the scope of some limit
FPT_SEP.1	TSF domain separation	The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.
FPT_PHP.3	Resistance to physical attack	Countermeasures against physical attacks
FDP_ITT.1	Basic internal transfer protection	Protection against leakage. Enforcing the Data Processing Policy to protect against leakage. FDP_IFC.1 provides the data processing policy.
FDP_IFC.1	Subset information flow control	
FPT_ITT.1 (1)	Basic internal TSF data transfer protection	
FAU_SAS.1	Audit storage	Storage of TOE identification data and pre-personalization data
FMT_LIM.1	Limited capability	Countermeasures against the abuse of TOE functionality by using Test features after TOE delivery
FMT_LIM.2	Limited availability	
FCS_RND.1	Quality metric for random number	Generation of good quality random numbers
FCS_COP.1	Cryptographic operation	Implementation of DES co-processor

FDP_ACC.1	Subset access control	Protection against improper access to memory. Enforcing Controlled Access Memory Policy to control the access to memory
FDP_ACF.1	Security attribute based access control	
FMT_MSA.3	Static attributes initialization	
FMT_MSA.1	Management of security attributes	
FMT_SMR.1	Security role	
FMT_SMF.1	Specification of management functions	
FDP_SDI.2	Stored data integrity and monitoring	Monitoring user data stored in FRAM
FPT_ITT.3	TSF data integrity monitoring	Monitoring for integrity of TSF data read from or written to FRAM
FPT_ITT.1 (2)	Basic internal TSF data transfer protection	

Table 1 Security Functional Requirement

The following functional requirements and those interpretations are derived from [BSI-PP-0002, section 5.1.1]

Malfunctions

There are different ranges of operating conditions such as supply voltage, external frequency and temperature. The TOE can be operated within the limits visualised as the inner dashed rounded rectangle in Figure 4 and must operate correctly there.
The limits have been reduced to ensure correct operation. This is visualised by the outer dotted rounded rectangle in the Figure 4.

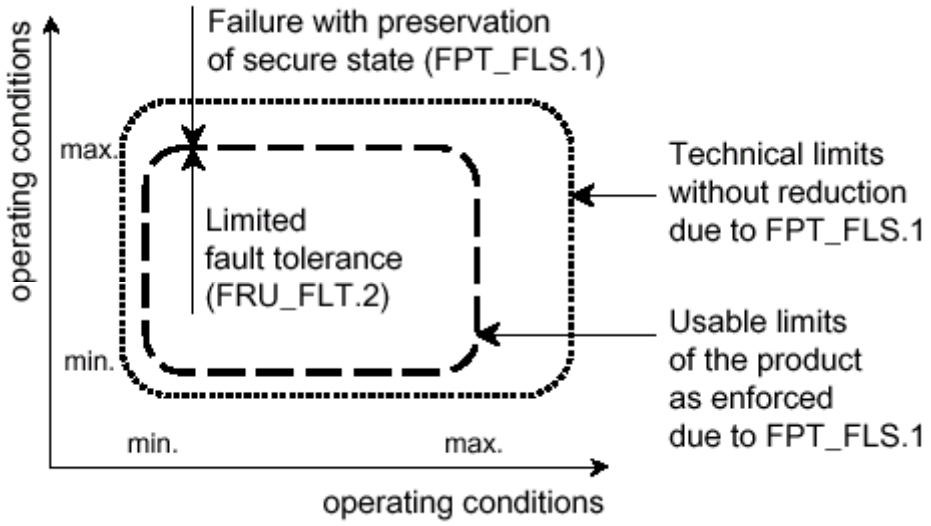

Figure 4 Paradigm regarding Operating Conditions

Figure 4 must not be understood as being two-dimensional and defining static limits only. Reality is multi-dimensional and includes a variety of timing aspects. Note that the limit of the operating conditions visualised by the inner dashed rounded rectangle in Figure 4 is not necessarily exactly reflected by the limits identified in the TOE's data sheet. Instead this limit marks the boundary between the "tolerance reaction" of the TOE and the "active reaction" of sensors (and perhaps other circuitry).

The security functional component **Limited fault tolerance (FRU_FLT.2)** has been selected in order to address the robustness within some limit (as shown by the inner dashed rectangle in Figure 4) before active reaction takes place. Note that the TOE does not actually detect faults or failures and then correct them in order to guarantee further operation of all the TOE's capabilities. This is the way software would implement Limited fault tolerance (FRU_FLT.2). Instead the TOE will achieve exactly the same by eliminating the cause for possible faults (by means of filtering for instance) and by being resistant against influences (robustness). In the case of the TOE the "reaction to a failure" is replaced by the "reaction to operating conditions" which could cause a malfunction without the reaction of the TOE's countermeasure.

If the TOE is exposed to other operating conditions this may not be tolerated. Then the TOE must detect that and "preserve a secure state" (use of detectors and cause a reset for instance). The security functional component **Failure with preservation of secure state (FPT_FLS.1)** has been selected to ensure that. The way the secure state is reached depends on the implementation. Note that the TOE can monitor both external operating conditions and other internal conditions and then react appropriately. Exposure to specific "out of range" external operating conditions (environmental stress) may actually cause failure conditions internally which can be detected by FPT_FLS.1. Referring to external operating conditions the TOE is expected to respond if conditions are detected that may cause a failure. Examples for implementations of the security functional requirement Failure with preservation of secure state (FPT_FLS.1) are a voltage detector (external condition) and a circuitry that detects accesses to address areas that are not used (internal condition).

Those parts of the TOE that support the security functional requirements "Limited fault tolerance (FRU_FLT.2)" and "Failure with preservation of secure state (FPT_FLS.1)" shall be protected from interference of the Smartcard Embedded Software. The security functional component **TSF Domain Separation (FPT_SEP.1)** has been selected to ensure that.

The TOE shall meet the requirement "Limited fault tolerance (FRU_FLT.2)" as specified below.

FRU_FLT.2	Limited fault tolerance
Hierarchical to:	FRU_FLT.1
FRU_FLT.2.1	The TSF shall ensure the operation of all the TOE's capabilities when the following failures occur: *exposure to operating conditions that are not detected according to the requirement Failure with preservation of secure state (FPT_FLS.1)*[6]
Dependencies:	FPT_FLS.1 Failure with preservation of secure state
Refinement:	The term "failure" above means "circumstances". The TOE prevents failures for the "circumstances" defined above.

[6] [assignment: list of type of failures]

Application note:
CXD9861 is assured to maintain secure operation within the scope of limited fault tolerance.

The TOE shall meet the requirement “Failure with preservation of secure state (FPT_FLS.1)” as specified below.

FPT_FLS.1	Failure with preservation of secure state
Hierarchical to:	No other components.
FPT_FLS.1.1	The TSF shall preserve a secure state when the following types of failures occur: *exposure to operating conditions that may not be tolerated according to the requirement Limited fault tolerance (FRU_FLT.2) and where therefore a malfunction could occur*[7].
Dependencies:	ADV_SPM.1 Informal TOE security policy model
Refinement 1:	The term “failure” above also covers “circumstances”. The TOE prevents failures for the “circumstances” defined above.
Refinement 2:	The term “Secure state” means that the TOE is reset, in order to preserve the TOE malfunction when outside the scope of “circumstances” are detected.

The TOE shall meet the requirement “TSF domain separation” state (FPT_SEP.1)” as specified below.

FPT_SEP.1	TSF domain separation
Hierarchical to:	No other components.
FPT_SEP.1.1	The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.
FPT_SEP.1.2	The TSF shall enforce separation between the security domains of subjects in the TSC.
Dependencies:	No dependencies.
Refinement:	Those parts of the TOE that support the security functional requirements “Limited fault tolerance (FRU_FLT.2)” and “Failure with preservation of secure state (FPT_FLS.1)” shall be protected from interference of the Smartcard Embedded Software.

[7] [assignment: list of types of failures in the TSF]

Abuse of Functionality

During testing at the end of Phase 3 before TOE Delivery, the TOE shall be able to store some data (for instance about the production history or identification data of the individual die or other data to be used after delivery). Therefore, the security functional component **Audit storage (FAU_SAS.1)** has been added. The security functional component FAU_SAS.1 has been newly created (refer to [BSI-PP-0002, Section 8.6]) and is used instead of FAU_GEN.1 that is too comprehensive to be applicable in this context.

The requirement FAU_SAS.1 shall be regarded as covering the injection of Initialisation Data, Pre-personalisation Data and of supplements of the Smartcard Embedded Software as described in [BSI-PP-0002, Section 8.1.1]. After TOE Delivery the identification data (injected as part of the Initialisation Data) and the Pre-personalisation Data are available to the Smartcard Embedded Software. These data are protected by the TOE as all other User Data. It's up to the Smartcard Embedded Software to use these data stored and provided by the TOE.

The TOE shall prevent functions (provided by the IC Dedicated Test Software and Test circuits) from being abused after TOE Delivery in order to compromise the TOE's security. (All such functions are called "Test Features" below.) This includes but is not limited to: disclose or manipulate User Data and bypass, deactivate, change or explore security features or functions of the TOE. Details depend on the capabilities of the Test Features provided by the IC Dedicated Test Software and Test circuits.

This can be achieved (i) by limiting the capabilities of these Test Features after Phase3, (ii) by limiting the availability of these Test Features after Phase 3 or (iii) by a combination of both. The security functional components **Limited capabilities (FMT_LIM.1)** and **Limited availability (FMT_LIM.2)** have been newly created (refer to [BSI-PP-0002, Section 8.5]) to address this.

The TOE shall meet the requirement "Limited capabilities (FMT_LIM.1)" as specified below (Common Criteria Part 2 extended).

FMT_LIM.1	Limited capabilities
Hierarchical to:	No other components.
FMT_LIM.1.1	The TSF shall be designed in a manner that limits their capabilities so that in conjunction with "Limited availability (FMT_LIM.2)" the following policy is enforced: *Deploying Test Features after TOE Delivery does not allow User Data to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks*[8].
Dependencies:	FMT_LIM.2 Limited availability.

[8] [assignment: Limited capability and availability policy]

Application note:
The IC dedicated test software and Test circuits implemented in TOE have the “capability” required in FMT_LIM.1.

The TOE shall meet the requirement “Limited availability (FMT_LIM.2)” as specified below (Common Criteria Part 2 extended).

FMT_LIM.2 Limited availability

Hierarchical to: No other components.

FMT_LIM.2.1 The TSF shall be designed in a manner that limits their availability so that in conjunction with “Limited capabilities (FMT_LIM.1)” the following policy is enforced: *Deploying Test Features after TOE Delivery does not allow User Data to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks*[9].

Dependencies: FMT_LIM.1 Limited capabilities.

Application note:
The IC dedicated test software, Test circuits implemented in TOE have the “availability” required in FMT_LIM.2.

The TOE shall meet the requirement “Audit storage (FAU_SAS.1)” as specified below (Common Criteria Part 2 extended).

FAU_SAS.1 Audit storage

Hierarchical to: No other components.

FAU_SAS.1.1 The TSF shall provide *test personnel before TOE Delivery*[10] with the capability to store *the Initialisation Data and/or Pre-personalisation Data and/or supplements of the Smartcard Embedded Software*[11] in the audit records.

Dependencies: No dependencies.

Refinement:
“Initialisation Data” means TOE Identification Data (Chip Manufacturing information), which is written in FRAM and is write-protected by initialization of Hardware. Also, “ Pre-personalisation Data” and “supplements of the Smartcard Embedded Software” include customer confidential data.

[9] [assignment: Limited capability and availability policy]
[10] [assignment: authorised users]
[11] [assignment: list of audit information]

Physical Manipulation and Probing

The TOE can be subject to "tampering" which pertains to (i) manipulation of the chip hardware and its security features with (ii) prior reverse-engineering to understanding the design and its properties and functions, (iii) determination of critical data through measuring using galvanic contacts, (iv) determination of critical data not using galvanic contacts and (v) manipulation or probing of memory contents.

The TOE is not always powered and therefore not able to detect, react or notify that it has been subject to tampering. Nevertheless, its design characteristics make reverse-engineering and manipulations etc. more difficult. This is regarded as being an "automatic response" to tampering. Therefore, the security functional component **Resistance to physical attack (FPT_PHP.3)** has been selected. The TOE may also provide features to actively respond to a possible tampering attack which is also covered by FPT_PHP.3.

The TOE may also leave it up to the Smartcard Embedded Software to react when a possible tampering has been detected. Comprehensive guidance (refer to Common Criteria assurance class AGD) will be given for the developer of the Smartcard Embedded Software in this case. Taking the assumption "Usage of Hardware Platform (A.Plat-Appl)" into consideration this case shall therefore also be covered by FPT_PHP.3.[12]

The TOE shall meet the requirement "Resistance to physical attack (FPT_PHP.3)" as specified below.

FPT_PHP.3	Resistance to physical attack
Hierarchical to:	No other components.
FPT_PHP.3.1	The TSF shall resist *physical manipulation and physical Probing*[13] to the *TSF*[14] by responding automatically such that the TSP is not violated.
Dependencies:	No dependencies.
Refinement:	The TOE will implement appropriate measures to continuously counter physical manipulation and physical probing. Due to the nature of these attacks (especially manipulation) the TOE can by no means detect attacks on all of its elements. Therefore, permanent protection against these attacks is required ensuring that the TSP could not be violated at any time. Hence, "automatic response" means here (i) assuming that there might be an attack at any time regardless power-on and (ii) countermeasures are provided at any time regardless power-on.

[12] This must be evaluated for the final smartcard product.
[13] [assignment: physical tampering scenarios]
[14] [assignment: list of TSF devices/elements]

Leakage

When the Smartcard processes User Data and/or TSF Data, information about these data may be leaked by signals which can be measured. An attacker may also cause malfunctions or perform manipulations of the TOE in order to cause the TOE to leak information. The analysis of those measurement data can lead to the disclosure of User Data and other critical data. Examples are given in [BSI-PP-0002, Section 8.3].

The security functional requirements "Basic internal transfer protection (FDP_ITT.1)" and "Basic internal TSF data transfer protection (FPT_ITT.1)" have been selected to ensure that the TOE must resist leakage attacks (both for User Data and TSF data).
The corresponding security policy is defined in the security functional requirement "Subset information flow control (FDP_IFC.1)". These security functional requirements address inherent leakage. With respect to forced leakage they have to be considered in combination with the security functional requirements "Limited fault tolerance (FRU_FLT.2)" and "Failure with preservation of secure state (FPT_FLS.1)" on the one hand and "Resistance to physical attack (FPT_PHP.3)" on the other.

The TOE shall meet the requirement "Basic internal transfer protection (FDP_ITT.1)" as specified below.

FDP_ITT.1	Basic internal transfer protection
Hierarchical to:	No other components.
FDP_ITT.1.1	The TSF shall enforce the *Data Processing Policy*[15] to prevent the *disclosure*[16] of user data when it is transmitted between physically separated parts of the TOE.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
Refinement:	The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor, the DMA controller) are seen as physically separated parts of the TOE.

The *DATA Processing Policy* is defined under FDP_IFC.1 below.

The TOE shall meet the requirement "Basic internal TSF data transfer protection (FPT_ITT.1)" as specified below.

FPT_ITT.1 (1)	Basic internal TSF data transfer protection
Hierarchical to:	No other components.
FPT_ITT.1.1	The TSF shall protect TSF data from *disclosure*[17] when it is transmitted between separate parts of the TOE.

[15] [assignment: access control SFP(s) and/or information flow control SFP(s)]
[16] [selection: disclosure, modification, loss of use]
[17] [selection: disclosure, modification]

Dependencies: No dependencies.

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

This requirement is equivalent to FDP_ITT.1 above but refers to TSF data instead of User Data. Therefore, it should be understood as to refer to the same *Data Processing Policy* defined under FDP_IFC.1 below.

The TOE shall meet the requirement " Subset information flow control (FDP_IFC.1)" as specified below:

FDP_IFC.1 Subset information flow control

Hierarchical to: No other components.

FDP_IFC.1.1 The TSF shall enforce the *Data Processing Policy*[18] on *all confidential data when they are processed or transferred by the TOE or by the Smartcard Embedded Software*[19].

Dependencies: FDP_IFF.1 Simple security attributes

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

Data processing Policy :
User Data and TSF data shall not be accessible from the TOE except when the Smartcard Embedded Software decides to communicate the User Data via an external interface. The protection shall be applied to confidential data only but without the distinction of attributes controlled by the Smartcard Embedded Software.

Random Numbers

The TOE generates random numbers. To define the IT security functional requirements of the TOE an additional family (FCS_RND) of the Class FCS (cryptographic support) is defined in [BSI-PP-0002, Section 8.4]. This class FCS_RND Generation of random numbers describes the functional requirements for random number generation used for cryptographic purposes.

The TOE shall meet the requirement "Quality metric for random numbers (FCS_RND.1)" as specified below (Common Criteria Part 2 extended).

FCS_RND.1 Quality metric for random numbers

FCS_RND.1.1 The TSF shall provide a mechanism to generate random numbers that meet

[18] [assignment: information flow control SFP]
[19] [assignment: list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP]

the functionality class K3 and *the strength of mechanism:high of [AIS20]* [20].

Dependencies: No dependencies.

Application note;
The random number generator conforms to the functionality class K3 and the strength of mechanism:high of [AIS20] using the 2-key Triple-DES co-processor (CBC mode). And algorithm is conformed to ANSIX9.42-2001 Annex C.2.

Cryptographic Support

The TOE provides a DES coprocessor, which can be used to implement single or triple DES. This SFR is additional requirement derived from "Smartcard Integrated Circuit Platform Augmentations, v1.0, march 2002".

FCS_COP.1 (1) Cryptographic operation

Hierarchical to: No other components.

FCS_COP.1.1 The TSF shall perform *encryption and decryption*[21] in accordance with a specified cryptographic algorithm *Data Encryption Standard (DES)*[22] and cryptographic key sizes *of 56 bit*[23] that meet the following *standards*[24] :

U.S. Department of Commerce / National bureau of Standards Data Encryption Standard (DES), FIPS PUB 46-3, 1999 October 25.

Dependencies: [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

FCS_COP.1 (2) Cryptographic operation

Hierarchical to: No other components.

FCS_COP.1.1 The TSF shall perform *encryption and decryption*[25] in accordance with a specified cryptographic algorithm *Triple Data Encryption Standard (3DES)* [26]and cryptographic key sizes *of 112 bit* [27]that meet the following *standards*[28].

[20] [assignment of Quality metric for Random number is under consideration.]
[21] [assignment: list of crypto-graphic operations]
[22] [assignment: cryptographic algorithm]
[23] [assignment: cryptographic key size]
[24] [assignment: list of standard]
[25] [assignment: list of crypto-graphic operations]
[26] [assignment: cryptographic algorithm]
[27] [assignment: cryptographic key size]
[28] [assignment: list of standard]

	U.S. Department of Commerce / National bureau of Standards Data Encryption Standard (DES), FIPS PUB 46-3, 1999 October 25, keying option 2
Dependencies:	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction FMT_MSA.2 Secure security attributes

The following are the additional SFRs to [BSI-PP-0002]

Controlled Access Memory

The TOE provides functions to protect FRAM against illegal exploitation of FRAM data.

FDP_ACC.1	Subset access control
Hierarchical to:	No other components.
FDP_ACC.1.1	The TSF shall enforce *the Controlled Access Memory Policy*[29] on *read or write operations performed on memory by any software code*[30].
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACF.1	Security Attribute based access control
Hierarchical to:	No other components.
FDP_ACF.1.1	The TSF shall enforce the *Controlled Access Memory Policy*[31] to objects based on the following: *- Subject: software code,* *- Object: FRAM, with the security attribute: accessible_area.*[32]
FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *- If a software code performs a read or write access attempt on FRAM out of the address range specified in accessible_area attribute, then a detection flag is raised and an interrupt request is output to CPU.*
FDP_ACF.1.3	The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *[none]*[33].
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the

[29] [assignment: access control SFP]
[30] [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]
[31] [assignment: access control SFP]
[32] [assignment: list of subjects and objects controlled under the indicated SFP, and, for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]
[33] [assignment: rules, based on security attributes, that explicitly authorize access of subjects to objects]

	[none][34].
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FMT_MSA.3	Static attributes initialization
Hierarchical to:	No other components.
FMT_MSA.3.1	The TSF shall enforce the *Controlled Access Memory Policy*[35] to *provide restrictive*[36] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow the *IC dedicated software*[37] to specify alternative initial values to override the default values when an object or information is created.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.1	Management of security attributes
Hierarchical to:	No other components.
FMT_MSA.1.1	The TSF shall enforce the *Controlled Access Memory Policy*[38] to restrict the ability to *modify*[39] the security attributes *accessible_area*[40] to *IC dedicated software*[41].
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_SMR.1	Security role
Hierarchical to:	No other components.
FMT_SMR.1.1	The TSF shall maintain the roles *IC dedicated software*[42].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.
Dependencies:	FIA_UID.1 Timing of identification

[34] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]
[35] [assignment: access control SFP, information flow control SFP]
[36] [selection: choose one of: restrictive, permissive, [assignment: other property]]
[37] [assignment: the authorized identified roles]
[38] [assignment: access control SFP, information flow control SFP]
[39] [selection: change_default, query, modify, delete, [assignment: other operations]]
[40] [assignment: list of security attributes]
[41] [assignment: the authorized identified roles]
[42] [assignment: the authorized identified roles]

Application note: This requirement is intended to identify the roles that will be authorized to modify the security attribute accessible_area.

FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions: *management of protected FRAM area*[43].

Dependencies: No Dependencies

These security functional requirements address the following Security Function Policy:

Controlled Access Memory Policy
The TOE shall monitor the FRAM memory in order to prevent a software code from gaining read or write access to protected FRAM areas. When the TOE detects an illegal access attempt, it notifies the software code with a detection flag and a CPU interrupt.
The TOE shall allow the IC dedicated software to modify the protected FRAM area boundaries.

Memory Integrity

The TOE provides functions to protect FRAM against unauthorized modification of FRAM data.

FDP_SDI.2 Stored data integrity and monitoring

FDP_SDI.2.1 The TSF shall monitor user data stored within the TSC for *integrity errors of FRAM read or write data*[44] on all objects, based on the following attributes: *verification of FRAM read or write data by Cyclic Redundancy Check*[45].

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall *notify the smartcard embedded software via the IC dedicated software*[46].

Dependencies: No dependencies

FPT_ITT.3 TSF data integrity monitoring

FPT_ITT.3.1 The TSF shall be able to detect *a data integrity error on FRAM*[47] for TSF data transmitted between separate parts of the TOE.

FPT_ITT.3.2 Upon detection of a data integrity error, the TSF shall take the following

[43] [assignment: list of security management functions to be provided by the TSF]
[44] [assignment: integrity errors]
[45] [assignment: user data attributes]
[46] [assignment: action to be taken]
[47] [selection: modification of data, substitution of data, re-ordering of data, deletion of data, [assignment: other integrity errors]]

actions:
- *notify error message to the smartcard embedded software via the IC dedicated software.*[48].

Dependencies: FPT_ITT.1 Basic internal TSF data transfer protection

FPT_ITT.1 (2) Basic internal TSF data transfer protection

Hierarchical to: No other components.

FPT_ITT.1.1 The TSF shall protect TSF data from *modification*[49] when it is transmitted between separate parts of the TOE.

Dependencies: No Dependencies

These security functional requirements address the following Security Function Policy:

FRAM Memory Integrity Policy
The TOE shall monitor the FRAM memory to protect user data and TSF data (e.g. RNG seed, TOE identification data) stored in the specified FRAM areas against modification. Upon detection of an integrity error, the IC dedicated software shall notify the smartcard embedded software.

[48] [assignment: specify the action to be taken]
[49] [selection: disclosure, modification]

5.1.2 TOE Assurance Requirements

In this section, the Assurance Requirements of EAL4 augmented are described according to [BSI-PP-0002]. Then they have to conform to Common Criteria v2.3.

The Assurance Requirements for the evaluation of the TOE and its development and operating environment are those taken from the Evaluation Assurance Level 4 (EAL4) and augmented by taking the following components:

ADV_IMP.2
ALC_DVS.2
AVA_MSU.3
AVA_VLA.4

The assurance requirements are:

Evaluation of Security Target (Class ASE)

Development activities (Class ADV)
- Functional Specification (Component ADV_FSP.2)
- Security Policy Modeling (Component ADV_SPM.1)
- High-Level Design (Component ADV_HLD.2)
- Low-Level Design (Component ADV_LLD.1)
- Implementation Representation (Component ADV_IMP.2)
- Representation Correspondence (Component ADV_RCR.1)

Tests activities (Class ATE)
- Coverage (Component ATE_COV.2)
- Depth (Component ATE_DPT.1)
- Functional Tests (Component ATE_FUN.1)
- Independent Testing (Component ATE_IND.2)

Delivery and operation activities (Class ADO)
- Delivery (Component ADO_DEL.2)
- Installation, generation, and start-up (Component ADO_IGS.1)

Guidance documents activities (Class AGD)
- Administrator Guidance (Component AGD_ADM.1)
- User guidance (Component AGD_USR.1)

Configuration management activities (Class ACM)
- CM automation (Component ACM_AUT.1)
- CM Capabilities (Component ACM_CAP.4)
- CM Scope (Component ACM_SCP.2)

Life cycle support activities (Class ALC)
- Development Security (Component ALC_DVS.2)
- Life Cycle Definition (Component ALC_LCD.1)
- Tools and Techniques (Component ALC_TAT.1)

Vulnerability assessment activities (Class AVA)

Misuse (Component AVA_MSU.3)
Strength of TOE Security Functions (Component AVA_SOF.1)
Vulnerability Analysis (Component AVA_VLA.4)

The minimum strength of security functions for the TOE is SOF-high (Strength of Functions High).

5.1.3 Refinements of the TOE Assurance Requirements

Refinements of the following assurance requirements are described in [*BSI-PP-0002, Section 5.1.3*]. They are required by [BSI-PP-0002] and shall support the comparability of evaluations according to [BSI-PP-0002].

Refinements regarding Delivery (ADO_DEL)
Refinements regarding Development Security (ALC_DVS)
Refinement regarding CM scope (ACM_SCP)
Refinement regarding CM capabilities (ACM_CAP)
Refinements regarding Functional Specification (ADV_FSP)
Refinement regarding Test Coverage (ATE_COV)
Refinement regarding Installation, Generation and Start-up (ADO_IGS)
Refinement regarding User Guidance (AGD_USR)
Refinement regarding Administrator Guidance (AGD_ADM)
Additional Guidance regarding Vulnerability Analysis (AVA_VLA)" and Strength of Functions (AVA_SOF)

Note that [BSI-PP-0002, Section 5.1.3] is not mandatory according to the Common Criteria. The Refinements of the TOE Assurance Requirements take into account the peculiarities of the smartcard development and production process (card's life-cycle).

The Refinements of assurance requirements are described according to [BSI-PP-0002, Section 5.1.3] and are modified in order to conform the CC v2.3.

In this security target (public version), the details of refinements are removed. For the details, see [BSI-PP-0002, Section 5.1.3] and CC v2.3.

5.2 Security Requirements for the Environment

5.2.1 Security Requirements for the IT-Environment

In [*BSI-PP-0002, Section 5.2.1*], the security objectives for the environment will be ensured by Non-IT security requirements only (refer to the next subsection, Section 5.2.2). The following requirements for the environment are derived from "Smartcard Integrated Circuit Platform Augmentations, v1.0, march 2002" and are applicable to Smartcard Embedded Software though depending on the users.

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

- [FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation],
- FCS_CKM.4 Cryptographic key destruction,
- FMT_MSA.2 Secure security attributes.

These requirements all address the appropriate management of cryptographic keys used by the specified cryptographic function. All 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. These operations cannot be performed in this Security Target because they all depend on the way the smartcard embedded software will be implemented.

The environment shall meet the requirement "Import of user data without security attributes (FDP_ITC.1)" or "Cryptographic key generation (FCS_CKM.1)" as specified below.

FDP_ITC.1	Import of user data without security attributes
Hierarchical to:	No other components.
FDP_ITC.1.1	The TSF shall enforce the [assignment: access control SFP and/or information flow control SFP] when importing user data, controlled under the SFP, from outside of the TSC.
FDP_ITC.1.2	The TSF shall ignore any security attributes associated with the user data when imported from outside of the TSC.
FDP_ITC.1.3	The TSF enforce the following rules when importing user data controlled under the SFP from outside the TSC: [assignment: additional importation control rules].
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_MSA.3 Static attribute initialization
FCS_CKM.1	Cryptographic key generation
Hierarchical to:	No other components.

FCS_CKM.1.1	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: cryptographic key generation algorithm] and specified cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].
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

The environment shall meet the requirement “Cryptographic key destruction (FCS_CKM.4)” as specified below.

FCS_CKM.4	Cryptographic key destruction
Hierarchical to:	No other components.
FCS_CKM.4.1	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [assignment: cryptographic key destruction method] that meet the following: [assignment: list of standards].
Dependencies:	[FDP_ITC.1 Import of user data without security attributes or FCS_CKM.1 Cryptographic key generation] FMT_MSA.2 Secure security attributes

The environment shall meet the requirement “Secure security attributes (FMT_MSA.2)” as specified below.

FMT_MSA.2	Secure security attributes
Hierarchical to:	No other components.
FMT_MSA.2.1	The TSF shall ensure that only secure values are accepted for security attributes.
Dependencies:	ADV_SPM.1 Informal TOE security policy model [FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles

5.2.2 Security Requirements for the Non-IT-Environment

In this section, security requirements for the non-IT-environment are described according to [BSI-PP-0002, Section 5.2.2].

In the following security requirements for the Non-IT-Environment are defined for the development of the Smartcard Embedded Software (in Phase 1) and the Smartcard Packaging, Finishing and Personalisation (Phases after TOE Delivery up to Phase 7).

The Smartcard Embedded Software is developed in Phase 1 and must support the security functionality of the TOE. This Security Target does not directly define obligatory security functional requirements for the Smartcard Embedded Software itself, because this might restrict the implementation possibilities for the developer. Instead the following general requirement for the design and implementation of the software is stated.

RE.Phase-1 Design and Implementation of the Smartcard Embedded Software

The developers shall design and implement the Smartcard Embedded Software in such way that it meets the requirements from the following documents:

(i) LSI specification;[SPC], HAL-API function specification;[HAL] and DRNG library specification;[DRN] and the TOE application notes
(ii) findings of the TOE evaluation reports relevant for the Smartcard Embedded Software.

The developers shall implement the Smartcard Embedded Software in a way that it protects security relevant User Data (especially cryptographic keys) as required by the security needs of the specific application context[50].

The requirement RE.Phase-1 also addresses the fact that the Smartcard Embedded Software may need to support the security functions of the TOE. Examples for such security functional requirements for the Smartcard Embedded Software are given in [*BSI-PP-0002, Section 8.2.2*]

The responsible parties for the Phases 4-6 are required to support the security of the TOE by appropriate measures:

RE.Process-Card Protection during Packaging, Finishing and Personalisation

The Card Manufacturer (after TOE Delivery up to the end of Phase 6) shall use adequate security measures 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).

[50] In particular, the Smartcard Embedded Software shall not disclose secret User Data to unauthorised users or processes as defined for the application context. Similarly the Smartcard Embedded Software shall not allow unauthorised users or processes to use or modify security relevant User Data

Additional requirement for the environment

The following Security Requirements for the non-IT environment is derived from [SSVG Augmentations]

The Smartcard Embedded Software shall meet the requirements “Cipher Schemas (RE.Cipher)” as specified below

RE.Cipher	Cipher Schemas
	The developers of Smartcard Embedded Software must not implement routines in a way that may compromise keys when the routines are executed as part of the Smartcard Embedded Software. Performing functions that access cryptographic keys could allow an attacker to misuse these functions to gather information about the key that is used in the computation of the function.
	Keys must be kept 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 not possible to derive the private key from 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 an appropriate key management has to be realised in the environment.

6 TOE Summary Specification

6.1 TOE Security Functions

IT Security Functions in this section corresponds to Security Functional Requirements that are defined in Section5.1.1.

Following security functions are active on phase 4 to 7.

SF.RNG:

The TOE implements a Deterministic Random Number Generator (DRNG) which generates 64bit random numbers which satisfy functionality class K3 specified in [AIS20] using the 2-key Triple-DES co-processor (CBC mode). This DRNG is dedicated to the Smartcard Embedded Software in order to generate cryptographic parameter on Smartcard.

This DRNG uses the 2-key Triple-DES co-processor (CBC mode) implemented in TOE and the algorithm of this RNG is conformed to “ANSIX9.42-2001 Annex C.2”. DRNG library as part of IC dedicated software supports operation of DRNG in secure means. Operation of DRNG library is described in [SPC], [HAL] and [DRN].

This Security Function satisfies Security Functional Component FCS_RND.1, and Random Numbers from this Random Number Generator (SF.RNG) are generated by cryptographic mechanism. Then the strength of mechanism is high according to the specific metric defined in AIS20.

SF.DES:

CXD9861 is equipped with a DES Co-processor. This DES Co-processor, which is in conformance with FIPS46-3, supplies 1-key DES processing and 2-key triple-DES processing and supports the ECB or CBC mode, encryption and decryption.

It also contains countermeasures against side-channel attacks (SPA, DPA, SEMA, DEMA) and DFA (Differential Fault Attack). This security function reduces the risk of leakage of confidential User data and TSF data. While the Security Function is active, an attacker cannot measure TOE behaviour from outside to analyse the cryptographic key and plain text.

This Security Function satisfies the Security Functional Components, FCS_COP.1, FDP_ITT.1, FPT_ITT.1 (1), and FDP_IFC.1.
Also, it satisfies the Data Processing Policy which is defined by FDP_ITT.1, FPT_ITT.1 (1), and FDP_IFC.1, and the policy “Additional Specific Security Functionality (P.Add-Functions)” which is defined by FCS_COP.1, on the other hand.

SF.Mal-Detect:

This Security Function has the sensor functions that detect when the TOE is used outside the scope of defined environment such as abnormal temperature, frequency and voltage.
This function includes four different functions below.

- Function to detect abnormal voltage outside of limit of fault tolerance:
  After detection, it asserts the internal reset.
- Function to detect abnormal FRAM voltage outside of limit of fault tolerance:
  After detection, FRAM access will be inhibited for protecting FRAM data including user data and TSF data.
- Function to detect abnormal clock frequency outside of limit of fault tolerance:
  After detection, it asserts the internal reset.
- Function to detect abnormal temperature outside of limit of fault tolerance:
  After detection, it asserts the internal reset.

This Security Function satisfies Security Functional Component, FRU_FLT.2, FPT_FLS.1, and FPT_SEP.1.

SF.Phy-Detect:

This Security Function provides an active shield which detects the physical modification of the TOE in order to protect the TOE against physical-probing and physical-manipulation. If the active shield detects physical manipulation, this security function generates interrupt request signals.

This Security Function satisfies FPT_PHP.3, and supports FDP_ITT.1, FPT_ITT.1 (1), and FDP_IFC.1.
Also, it satisfies the Data Processing Policy which is defined by FDP_ITT.1, FPT_ITT.1 (1), and FDP_IFC.1.

SF.Phy-Protect:

This Security Function provides the physical layout that protects the TOE from physical manipulation and physical probing and make difficult to attack. Furthermore, this security function provides the physical functions to protect confidential information (user data and TSF data) against the electro magnetic attack and electric power attack, like side channel attack.

In the TOE, various physical mechanisms are implemented to protect the TOE from the attacks.

This Security Function satisfies FPT_PHP.3, and supports FDP_ITT.1, FPT_ITT.1 (1), and FDP_IFC.1.
Also, it satisfies the Data Processing Policy that is defined by FDP_ITT.1, FPT_ITT.1 (1), and FDP_IFC.1.

SF.TEST：

At the end of phase3 (before TOE delivery), IC testing is performed by the Test features including IC Dedicated Test software and Test circuits in order to assure the correct operation of TOE function and the quality of TOE operation.

Once IC testing is completed, Test features are invalidated. Therefore, User data and TSF data are not disclosed by manipulating the test functions at the user phases.

This Security Function satisfies Security Functional Component, FMT_LIM.1, and FMT_LIM.2. Also, it satisfies the underlying policy defined by FMT_LIM.1 and FMT_LIM.2.

SF.Identification：

This Security Function provides the ability to write each TOE identification data on FRAM and pre-personalization data.

The identification data includes information about TOE Chip manufacturing. This identification data is stored in FRAM at the IC testing and is prohibited to write access after testing.

Pre-personalization data and Supplement of Smartcard Embedded Software are provided as confidential data that depend on Smartcard Embedded Software by smartcard embedded software developer in phase2 and are stored on the TOE at the IC testing.

This Security Function satisfies Security Functional Component, FAU_SAS.1.

SF.Memory-Access:

This Security Function provides the functions that detect when the malicious software code performs unauthorized access to the TOE and the DMA access data is modified deliberately, in order to prevent disclose of the confidential data (User data and TSF data).

- Controlled access of DMA
  DMA controller has a function which detects that the DMA access data is modified deliberately. If this function detects the modification of the DMA access data, the interrupt signal is requested to the CPU.

- Controlled write/read on the accessible FRAM area
  This function detects that a software code performs a read or write access on FRAM out of the address range specified in accessible area. By default, the whole FRAM area is inaccessible. The range of the protected FRAM area can be modified only by the IC Dedicated Software (HAL-API, DRNG Library). If it detects the unauthorized access, a detection flag is raised and an interrupt signal is requested to the CPU.

This Security Function satisfies Security Functional Component, FDP_ACC.1, FDP_ACF.1, FMT_MSA.3, FMT_MSA.1, FMT_SMF.1 and FMT_SMR.1, and supports FDP_ITT.1, FPT_ITT.1 (1), and FDP_IFC.1.

Also it satisfies the Controlled Access Memory Policy that are defined by FDP_ACC.1, FDP_ACF.1, FMT_MSA.3, FMT_MSA.1, FMT_SMF.1, FMT_SMR.1.

SF.Memory-Scramble:

This Security Function provides the function that protects the confidential data stored

in TOE memory areas against the manipulation and probing attacks. This function scrambles memory address data logically and makes difficult to read the memory data physically from outside.

This Security Function satisfies Security Functional Component, FPT_PHP.3 and supports to FDP_ITT.1, FPT_ITT.1 (1), and FDP_IFC.1.
Also, it satisfies the Data Processing Policy that is defined by FDP_ITT.1, FPT_ITT.1 (1), and FDP_IFC.1.

SF.Memory-Verification:

This Security Function provides a CRC function to assure the integrity of FRAM data. If CRC error is generated, error message is notified the smartcard embedded software via HAL-API and DRNG Library interface.

This Security Function satisfies Security Functional Component, FPT_ITT.3, FPT_ITT.1 (2) and FDP_SDI.2.

Also it satisfies the Memory Integrity Policy that is related to these requirements.

6.2 Assurance measures

The table below is the summary of Documents that satisfy Security Assurance Requirements of Section5.1.2.

Security Assurance components	Security Assurance documents
ADV_FSP.2	Functional Specification
ADV_HLD.2	High Level Design for each SFs
ADV_LLD.1	Low Level Design for each SFs
ADV_IMP.2	Implementation Representation
ADV_RCR.1	Correspondence analysis between TOE summary specification, functional specification, high level design, low level design and implementation
ADV_SPM.1	This Security Target includes this requirement.
AGD_ADM.1	No document required
AGD_USR.1	LSI Specification, HAL-API function specification, DRNG Library specification
ACM_AUT.1	Configuration Management system document
ACM_CAP.4	
ACM_SCP.2	
ADO_DEL.2	Parts of Delivery Procedure documentation
ADO_IGS.1	No document required
ALC_DVS.2	Development Security documentation for each development sites
ALC_LCD.1	Life cycle flow for TOE development
ALC_TAT.1	Tool list is included in configuration management system documentation
ATE_COV.2	Functional tests specification including tests coverage and test depth
ATE_DPT.1	
ATE_FUN.1	
ATE_IND.2	No document required
AVA_MSU.3	Misuse analysis
AVA_SOF.1	Strength of function analysis
AVA_VLA.4	Vulnerability analysis

Table 2 List of document describing the measures regarding the assurance requirements

7 PP Claims

7.1 PP reference

.

This Security Target is conformant to [BSI-PP-0002], which has been registered at the German Certification Body.

Additionally, this Security Target makes reference to [SSVG Augmentations]

7.2 PP tailoring

This section shows SFRs that is tailored from [BSI-PP-0002].

Security Functional Requirement	Operation	Note
FCS_RND.1	assignation	Tailoring
FPT_FLS.1	refinement	Addition of “secure state”
FAU_SAS	refinement	Modification of the PP’s refinement
ACM_SCP	Conformance of CCv2.3	Modification of the PP’s assurance requirements
ACM_CAP	Conformance of CCv2.3	Modification of the PP’s assurance requirements
ADO_IGS	Conformance of CCv2.3	Modification of the PP’s assurance requirements
ADO_DEL	refinement	Modification of the PP’s refinement
ATE_COV	refinement	Modification of the PP’s refinement

Table 3 PP tailoring

Note also that FPT_ITT.1 is renamed into FPT_ITT.1 (1).

7.3 PP additions

This section shows the security objectives and IT security requirements statements that are in addition to those contained in the PP [BSI-PP-0002] towhich this Security Target conforms.

Additions	Sections
O.Add-Functions	Section 4.1
O.Memory_Integrity	Section 4.1
O.Memory_Access	Section 4.1
FDP_ACC.1	Section 5.1.1
FDP_ACF.1	Section 5.1.1
FMT_MSA.3	Section 5.1.1
FMT_MSA.1	Section 5.1.1
FMT_SMF.1	Section 5.1.1
FMT_SMR.1	Section 5.1.1
FDP_SDI.2	Section 5.1.1
FCS_COP.1	Section 5.1.1
FPT_ITT.3	Section 5.1.1
FPT_ITT.1 (2)	Section 5.1.1

Table 4 PP additions

8 Rationale

8.1 Security Objectives Rationale

Table 5 below gives an overview, how the assumptions, threats, and organisational security policies that are given in [BSI-PP-0002] are addressed by the objectives. Furthermore the Table 5 below adds the rational for the additional assumptions and policy in this Security Target. The text following after the table justifies this in detail.

Assumption, Threat or Organisational Security Policy	Security Objective	Note
A.Plat-Appl	OE.Plat-Appl	(Phase1)
A.Resp-Appl	OE.Resp-Appl	(Phase1)
A.Key-Function	OE.Plat-Appl OE.Resp-Appl	(Phase1)
P.Process-TOE	OE.Process-TOE, O.Identification	(Phase2 - 3)
P.Add-Functions	O.Add-Functions	
A.Process-Card	OE.Process-Card	(Phase4 - 6)
T.Leak-Inherent	O.Leak-Inherent	
T.Phys-Probing	O.Phys-Probing	
T.Malfunction	O.Malfunction	
T.Phys-Manipulation	O.Phys-Manipulation	
T.Leak-Forced	O.Leak-Forced	
T.Abuse-Func	O.Abuse-Func	
T.RND	O.RND	
T.Memory-Integrity	O.Memory-Integrity	
T.Memory-Access	O.Memory-Access	

Table 5 Security Objectives versus Assumption, Threat or Policies

The justification related to the assumption “Usage of Hardware Platform (A.Plat-Appl)” is as follows:

Since OE.Plat-Appl requires the Smartcard Embedded Software developer to implement those measures assumed in A.Plat-Appl, the assumption is covered by the objective.

The justification related to the assumption “Treatment of User Data (A.Resp-Appl)” is as follows:

Since OE.Resp-Appl requires the developer of the Smartcard Embedded Software to implement measures as assumed in A.Resp-Appl, the assumption is covered by the objective.

The justification related to the assumption “Usage of Key-dependent Function (A.Key-Function)” is as follows:

Since OE.Plat-Appl and OE.Resp-Appl require the Smartcard Embedded Software to use cryptographic services and functions as assumed in A.Key-Function, the assumption is covered by the objective.

The justification related to the organisational security policy “Protection during TOE Development and Production (P.Process-TOE)” is as follows:

OE.Process-TOE requires the TOE Manufacturer to implement those measures assumed in P.Process-TOE. Therefore, the organisational security policy is covered by this objective, as far as organisational measures are concerned. The only issue not completely covered by these measures is the fact that the TOE has to support the possibility of unique identification. This is the content of O.Identification. Therefore, the organisational security policy is covered by OE.Process-Card and O.Identification.

The justification related to the assumption “Protection during Packaging, Finishing and Personalisation (A.Process-Card)” is as follows:

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

The justification related to the threats “Inherent Information Leakage (T.Leak-Inherent)”, “Physical Probing (T.Phys-Probing)”, “Malfunction due to Environmental Stress (T.Malfunction)”, “Physical Manipulation (T.Phys-Manipulation)”, “Forced Information Leakage (T.Leak-Forced)“, “Abuse of Functionality (T.Abuse-Func)”, “Deficiency of Random Numbers (T.RND)”, “Memory Integrity of FRAM (T.Memory-Integrity)”and “Memory Access to FRAM (T.Memory-Access)” is as follows:

For all threats the corresponding objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced, O.Abuse-Func, O.RND, O.Memory-Integrity and O.Memory-Access are stated in a way, which directly corresponds to the description of the threat (refer to Section3.3). It is clear from the description of each objective (refer to Section4.1), that the corresponding threat is removed if the objective is valid. More specifically, in every case the ability to use the attack method successfully is countered, if the objective holds.

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

Since O.Add-Functions requires the TOE to implement exactly the same specific functionality as required by P.Add-Functions, the organizational security policy is covered by the objective.

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

Compared to [BSI-PP-0002] a 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. 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 [BSI-PP-0002] a clarification has been made for the security objective “Treatment of User Data (OE.Resp-Appl)”: By definition encryption data, plain text data and cryptographic keys are User Data. So, the Smartcard Embedded Software will protect such data if required and use keys and functions appropriately in order to ensure the strength of cryptographic operation. Quality and confidentiality must be maintained for keys that are imported and/or derived from other keys. This implies that appropriate key management has to be realized in the environment. 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.

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

8.2 Security Requirements Rationale

8.2.1 Rationale for the security functional requirements

The way in which [BSI-PP-0002] objectives are implemented by SFRs and requirements on the environment is given in [BSI-PP-0002, Section 7.2]. The table below includes the mapping from [BSI-PP-0002, Section7.2] and adds the rationale for the additional SFRs in this Security Target.

Objective	TOE SFR	Security Requirements for the environment
O.Leak-Inherent	FDP_ITT.1 FPT_ITT.1 (1) FDP_IFC.1	RE.Phase-1
O.Phys-Probing	FPT_PHP.3	RE.Phase-1
O.Malfunction	FRU_FLT.2 FPT_FLS.1 FPT_SEP.1	-
O.Phys-Manipulation	FPT_PHP.3	RE.Phase-1 (e.g. by implementing FDP_SDI.1 Stored data integrity monitoring)
O.Leak-Forced	FDP_ITT.1 FPT_ITT.1 (1) FDP_IFC.1 FRU_FLT.2 FPT_FLS.1 FPT_SEP.1 FPT_PHP.3	RE.Phase-1
O.Abuse-Func	FMT_LIM.1 FMT_LIM.2 FDP_ITT.1 FPT_ITT.1 (1) FDP_IFC.1 FPT_PHP.3 FRU_FLT.2 FPT_FLS.1 FPT_SEP.1	-
O.Identification	FAU_SAS.1	-

O.RND	FCS_RND.1 FDP_ITT.1 FPT_ITT.1 (1) FDP_IFC.1 FPT_PHP.3 FRU_FLT.2 FPT_FLS.1 FPT_SEP.1	RE.Phase-1
O.Add-Functions	FCS_COP.1	RE.Phase-1 RE.Cipher
O.Memory-Integrity	FPT_ITT.3 FDP_SDI.2 FPT_ITT.1 (2)	-
O.Memory-Access	FDP_ACC.1 FDP_ACF.1 FMT_MSA.3 FMT_MSA.1 FMT_SMF.1 FMT_SMR.1	-
OE.Plat-Appl	-	RE.Phase-1
OE.Resp-Appl	-	RE.Phase-1 RE.Cipher FDP_ITC.1 or FCS_CKM.1, FCS_CKM.4, FMT_MSA.2
OE.Process-TOE	FAU_SAS.1	Assurance Components: Refer to below***
OE.Process-Card	-	RE.Process-Card possibly supported by RE.Phase-1

Table 6 Security Requirements versus Security Objectives

*** Assurance Components: Delivery (ADO_DEL); Installation, generation, and start- up (ADO_IGS) (using Administrator Guidance (AGD_ADM), User guidance (AGD_USR); CM automation (ACM_AUT); CM Capabilities (ACM_CAP); CM Scope (ACM_SCP); Development Security (ALC_DVS); Life Cycle Definition (ALC_LCD); Tools and Techniques (ALC_TAT)

The justification related to the security objective “Protection against Inherent Information Leakage (O.Leak-Inherent)” is as follows:

The refinements of the security functional requirements FPT_ITT.1 (1) and FDP_ITT.1 together with the policy statement in FDP_IFC.1 explicitly require the prevention of disclosure of secret data (TSF data as well as User Data) when transmitted between separate parts of the TOE or while being processed. This includes that attackers cannot reveal such data by measurements of electro/magnetic emanations, power consumption or other behaviour of the TOE while data are transmitted between or processed by TOE parts.

Of course this has also to be supported by the Smartcard Embedded Software. For example timing attacks were possible if the processing time of algorithms implemented in the software

would depend on the content of secret variables. The requirement RE.Phase-1 makes sure that this is avoided.

The justification related to the security objective “Protection against Physical Probing (O.Phys-Probing)” is as follows:

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

It is possible that the TOE needs additional support by the Smartcard Embedded Software (e.g. to send data over certain buses only with appropriate precautions). If necessary this support is provided according to RE.Phase-1. Together with this FPT_PHP.3 is suitable to meet the objective.

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

The definition of this objective shows that it covers a situation, where malfunction of the TOE might be caused by the operating conditions of the TOE (while direct manipulation of the TOE is covered O.Phys-Manipulation). There are two possibilities in this situation: Either the operating conditions are inside of the tolerated range or at least one of them is outside of this range. The second case is covered by FPT_FLS.1, because it states that a secure state is preserved in this case. The first case is covered by FRU_FLT.2 because it states that the TOE operates correctly under normal (tolerated) conditions. To support this, FPT_SEP.1 the functions implementing FRU_FLT.2 and FPT_FLS.1 must work independently so that their operation cannot affected by the Smartcard Embedded Software. Therefore, there is no possible instance of conditions under O.Malfunction, which is not covered.

The justification related to the security objective “Protection against Physical Manipulation (O.Phys-Manipulation)” is as follows:

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

It is possible that the TOE needs additional support by the Embedded Software (for instance by implementing FDP_SDI.1 to check data integrity with the help of appropriate checksums). This support is provided according to RE.Phase-1. Together with this FPT_PHP.3 is suitable to meet the objective.

The justification related to the security objective “Protection against Forced Information Leakage (O.Leak-Forced)“ is as follows:

This objective is directed against attacks, where an attacker wants to force an information leakage, which would not occur under normal conditions. In order to achieve this the attacker has to combine a first attack step, which modifies the behaviour of the TOE (either by exposing it to extreme operating conditions or by directly manipulating it) with a second attack step measuring and analysing some output produced by the TOE. The first step is prevented by the same measures that support O.Malfunction and O.Phys-Manipulation, respectively. The requirements covering O.Leak-Inherent also support O.Leak-Forced because they prevent the

attacker from being successful if he tries the second step directly.

The justification related to the security objective “Protection against Abuse of Functionality (O.Abuse-Func)” is as follows:

This objective states that abuse of functions (especially provided by the IC Dedicated Test Software, for instance in order to read secret data) must not be possible in Phase7 of the life cycle. There are two possibilities to achieve this: (i) They cannot be used by an attacker (i. e. its availability is limited) or (ii) using them would not be of relevant use for an attacker (i. e. its capabilities are limited) since the functions are designed in a specific way. The first possibility is specified by FMT_LIM.2 and the second one by FMT_LIM.1. Since these requirements are combined to support the policy, which is suitable to fulfil O.Abuse-Func, both security functional requirements together are suitable to meet the objective.

Other security functional requirements that prevent attackers from circumventing the functions implementing these two security functional requirements (for instance by manipulating the hardware) also support the objective. The relevant objectives are also listed in *Table 6*.

It was chosen to define FMT_LIM.1 and FMT_LIM.2 explicitly (not using Part 2 of the Common Criteria) for the following reason: Though taking components from the Common Criteria catalogue makes it easier to recognise functions, any selection from Part 2 of the Common Criteria would have made it harder for the reader to understand the special situation meant here. As a consequence, the statement of explicit security functional requirements was chosen to provide more clarity.

The justification related to the security objective “TOE Identification (O.Identification)“ is as follows:

Obviously the operations for FAU_SAS.1 are chosen in a way that they require the TOE to provide the functionality needed for O.Identification. The Initialisation Data are used for TOE identification.

It was chosen to define FAU_SAS.1 explicitly (not using a given security functional requirement from Part 2 of the Common Criteria) for the following reason: The security functional requirement FAU_GEN.1 in Part 2 of the CC requires the TOE to generate the audit data and gives details on the content of the audit records (for instance data and time). The possibility to use the functions in order to store security relevant data which are generated outside of the TOE, is not covered by the family FAU_GEN or by other families in Part 2. Moreover, the TOE cannot add time information to the records, because it has no real time clock. Therefore, the new family FAU_SAS was defined for this situation.

The justification related to the security objective “Random Numbers (O.RND)” is as follows:

FCS_RND.1 requires the TOE to provide random numbers, which is in conformity with [AIS20].

Other security functional requirements, which prevent physical manipulation and malfunction of the TOE (see the corresponding objectives listed in the table) support this objective because they prevent attackers from manipulating or otherwise affecting the random number generator.

Random numbers are often used by the Smartcard Embedded Software to generate

cryptographic keys for internal use. Therefore, the TOE must prevent the unauthorised disclosure of random numbers. Other security functional requirements that prevent inherent leakage attacks, probing and forced leakage attacks ensure the confidentiality of the random numbers provided by the TOE.

Depending on the functionality of specific TOEs the Smartcard Embedded Software will have to support the objective by providing runtime-tests of the random number generator. Together, these requirements allow the TOE to provide cryptographically good random numbers and to ensure that no information about the produced random numbers is available to an attacker.

It was chosen to define FCS_RND.1 explicitly, because Part 2 of the Common Criteria do not contain generic security functional requirements for Random Number Generation. (Note, that there are security functional requirements in Part 2 of the Common Criteria, which refer to random numbers. However, they define requirements only for the authentication context, which is only one of the possible applications of random numbers.)

The justification related to the security objective “Memory Integrity of FRAM (O.Memory-Integrity)” is as follows:

The objective states protection of TSF data and User data on FRAM against modification. SFRs, which satisfy the requirement of this objective, are twofold. Regarding TSF data integrity, FPT_ITT.1 (2) and FPT_ITT.3 require the TSF to respectively protect TSF data from modification and monitor TSF data integrity by detecting a data integrity error on FRAM with a cyclic redundancy check and notifying error message to the smartcard embedded software via the IC dedicated software. Regarding user data integrity, FDP_SDI.2 requires verifying FRAM read or write data by performing cyclic redundancy check, in order to protect user data stored in FRAM against its modification. If integrity errors of FRAM read or write data are detected, FDP_SDI.2 requires notifying the smartcard embedded software via the IC dedicated software. FDP_SDI.2 together with FPT_ITT.3 and FPT_ITT.1 (2) monitor user data and TSF data integrity of FRAM read or write data and detect a data integrity error of them in order to prevent disclosure or modification of TSF data and User data on FRAM. Therefore these three SFRs satisfy O.Memory-Integrity.

The justification related to the security objective “Memory Access to FRAM (O.Memory-Access)” is as follows:

The objective states protection against attempts to gain illegal access on protected FRAM area. Therefore, in order to achieve this objective, FDP_ACC.1 and FDP_ACF.1 require enforcing the Controlled Access Memory Policy by detecting against unauthorized access to FRAM. The smartcard embedded software shall not be able to modify the protected FRAM area boundaries. This is met by the requirements FMT_MSA.3, FMT_MSA.1 FMT_SMR.1 and FMT_SMF.1. Indeed, FMT_SMR.1 requires the TSF to maintain the role of IC dedicated software, which is part of the TOE, and FMT_SMF.1 requires the TSF to perform a management function on the protected FRAM area whose rules are defined by FMT_MSA.3 and FMT_MSA.1, that respectively require that only the IC dedicated software can initialize and modify the protected FRAM area boundaries.

The justification related to the security objective “Usage of Hardware Platform (OE.Plat-Appl)” is as follows:

RE.Phase-1 requires the Smartcard Embedded Software developer to design and implement the

software in a way, which is suitable to meet OE.Plat-Appl.

The justification related to the security objective “Treatment of User Data (OE.Resp-Appl)” is as follows:

FDP_ITC.1 or FCS_CKM.1, FCS_CKM.4 and FMT_MSA.2 require the appropriate management of cryptographic keys used by the specified cryptographic function. These requirements satisfy the OE.Resp-Appl since cryptographic keys and plain text data are defined user data and the Smartcard Embedded Software must treat them appropriately in OE.Resp-Appl.

RE.Phase-1 requires the developer of the Smartcard Embedded Software to design and implement the software in a way, which is suitable to meet OE.Resp-Appl.

RE.Cipher requires the developers of Smartcard Embedded Software not to implement routines in a way that may compromise keys when the routines are executed as part of the Smartcard Embedded Software. This requirement satisfies the OE.Resp-Appl that requires using cryptographic keys and plain text data appropriately.

The justification related to the security objective “Protection during TOE Development and Production (OE.Process-TOE)” is as follows:

The objective OE.Process-TOE has mainly to be fulfilled by organisational and other measures, which the TOE Manufacturer has to implement in Phase2 and3. These measures are a subset of those measures, which are examined during the evaluation of the assurance requirements of the classes ACM, AGD, ALC and ADO. The technical capability of the TOE to store Initialisation Data and/or Pre-personalisation Data is provided according to FAU_SAS.1. Together these security requirements are suitable to meet the objective.

The justification related to the security objective “Protection during Packaging, Finishing and Personalisation (OE.Process-Card)” is as follows:

RE.Process-Card requires the Card Manufacturer for Phase4-6 to use adequate measures to fulfil OE.Process-Card. Depending on the security needs of the application, the Smartcard Embedded Software may have to support this for instance by using appropriate authentication mechanisms for personalisation functions. Therefore, RE.Phase-1 may support RE.Process-Card in fulfilling the objective in addition.

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 require those functions to be implemented that are demanded by O.Add-Functions. Therefore, FCS_COP.1 is suitable to meet the security objective.

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 requirement RE.Phase-1 and more specific by the security functional requirements.

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

key generation],
- FCS_CKM.4 Cryptographic key destruction,
- FMT_MSA.2 Secure security attributes.

to be met by the environment.

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. In this case, RE.Cipher requires that these functions ensure that confidential data (User Data) cannot be disclosed while they are just being processed by the Smartcard Embedded Software. Therefore, with respect to the Smartcard Embedded Software the issues addressed by the objectives just mentioned are addressed by the requirement RE.Cipher.

The usage of cryptographic algorithms requires using appropriate keys. Otherwise they do not provide security. The requirement RE.Cipher addresses these specific issues since cryptographic keys and other data are provided by the Smartcard Embedded Software. RE.Cipher requires that keys must be kept confidential. They must be unique with a very high probability, cryptographically strong etc. If keys are imported into the TOE (usually after TOE Delivery), it must be ensured that quality and confidentiality is maintained. Therefore, with respect to the environment the issues addressed by the objectives just mentioned and implicitly by O.Add-Functions are addressed by the requirement RE.Cipher.

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

8.2.2 Dependencies of security functional requirements

Table 7 below lists the security functional requirements applied to [*BSI-PP-0002, Section 7.2.2*].

Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
FRU_FLT.2	FPT_FLS.1	Yes
FPT_FLS.1	ADV_SPM.1	Yes (Part of EAL4)
FPT_SEP.1	None	No dependency
FMT_LIM.1	FMT_LIM.2	Yes
FMT_LIM.2	FMT_LIM.1	Yes
FAU_SAS.1	None	No dependency
FPT_PHP.3	None	No dependency
FDP_ITT.1	FDP_ACC.1 or FDP_IFC.1	Yes
FDP_IFC.1	FDP_IFF.1	See discussion below
FPT_ITT.1 (1)	None	No dependency
FCS_RND.1	None	No dependency

Table 7 Dependencies of Security Functional Requirements

Part 2 of the Common Criteria defines the dependency of FDP_IFC.1 (information flow control policy statement) on FDP_IFF.1 (Simple security attributes). The specification of FDP_IFF.1 would not capture the nature of the security functional requirement nor add any detail. As stated in the *Data Processing Policy* referred to in FDP_IFC.1 there are no attributes necessary. The security functional requirement for the TOE is sufficiently described using FDP_ITT.1 and its *Data Processing Policy* (FDP_IFC.1). Therefore the dependency is considered satisfied.

As Table 7 shows, all other dependencies are fulfilled by security requirements defined in this Security Target.

The discussion in Section8.2.1 has shown, how the security functional requirements support each other in meeting the security objectives of this Security Target. In particular the security functional requirements providing resistance of the hardware against manipulations (e.g. FPT_PHP.3) support all other more specific security functional requirements (e.g. FCS_RND.1) because they prevent an attacker from disabling or circumventing the latter. Together with the discussion of the dependencies above this shows that the security functional requirements build a mutually supportive whole.

The additional dependencies relating to the new SFRs introduced in this ST are analysed below.

Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
FCS_COP.1	[FDP_ITC.1 or FCS_CKM.1] FCS_CKM.4 FMT_MSA.2	Yes (By the IT environment)
FDP_ACC.1	FDP_ACF.1	Yes
FDP_ACF.1	FDP_ACC.1, FMT_MSA.3	Yes
FMT_MSA.3	FMT_MSA.1, FMT_SMR.1	Yes
FMT_MSA.1	[FDP_ACC.1 or FDP_IFC.1], FMT_SMF.1, FMT_SMR.1	Yes
FMT_SMF.1	None	No dependency
FMT_SMR.1	FIA_UID.1	See discussion
FDP_SDI.2	None	No dependency
FPT_ITT.3	FPT_ITT.1 (2)	Yes
FPT_ITT.1 (2)	None	No dependency

Table 8 Additional SFR dependencies

The dependencies of FCS_COP.1 (Cryptographic operation) defined Part2 of the Common Criteria are covered by the IT environment because the Smartcard Embedded Software uses the cryptographic functions provided by the TOE. Dependencies of security requirements in the IT environment are fulfilled FCS_COP.1 by the Smartcard Embedded Software.

The dependency of FMT_SMR.1 on FIA_UID.1 is not relevant in this particular case because the defined role is the IC dedicated software, which is, in fact, part of the TOE. So there is no need of identification by the TSF.

As Table 8 shows, all other dependencies are fulfilled by security requirements defined in this Security Target.

8.2.3 Assurance Requirements and the Strength of Function Level

This Section is described according to [*BSI-PP-0002, Section 7.2.3*].

The assurance level EAL4 and the augmentation with the requirements ADV_IMP.2, ALC_DVS.2, AVA_MSU.3, and AVA_VLA.4 were chosen in order to meet assurance expectations explained in the following paragraphs.

An assurance level of EAL4 is required for this type of TOE since it is intended to defend against sophisticated attacks. This evaluation assurance level was selected since it is designed 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 the low level design and source code.

ADV_IMP.2 Implementation of the TSF

This assurance component is a higher hierarchical component to EAL4 (which only requires ADV_IMP.1). It is important for a smartcard IC that the evaluation includes the implementation representation of the entire TSF and determines whether the functional requirements in the Security Target are addressed by the representation of the TSF. IC dedicated software source code and IC hardware drawings are examples of TSF implementation representation.

The implementation representation is used to express the notion of the least abstract representation of the TSF, specifically the one that is used to create the TSF itself without further design refinement.

ADV_IMP.2 has dependencies with ADV_LLD.1 “Descriptive Low-Level design”, ADV_RCR.1 “Informal correspondence demonstration”, ALC_TAT.1 “Well defined development tools”. These assurance components are included in EAL4, then these dependencies are satisfied.

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 Smartcard Integrated Circuit the TOE is developed and produced within a complex and distributed industrial process that must especially be protected. Details about the implementation (e.g. from design, test and development tools as well as Initialisation Data) may make such attacks easier.
Therefore, in the case of a Smartcard Integrated Circuit, 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.

AVA_MSU.3 Analysis and testing for insecure states

The user guidance must be correct and sufficient to ensure that the TOE can be used in a secure way and that vulnerabilities are not introduced.

This component is included to ensure that misleading, unreasonable and conflicting guidance is absent from the guidance documentation, and that secure procedures for all modes of operation have been addressed. Insecure states should be easy to detect. In this component, an analysis of the guidance documentation provided by the developer is validated and confirmed through testing by the evaluator to provide additional assurance.

This assurance component is a higher hierarchical component to EAL4 (which only requires AVA_MSU.2).

AVA_MSU.3 has dependencies with ADO_IGS.1 “Installation, generation, and start-up procedures“, ADV_FSP.1 “Informal functional specification”, AGD_ADM.1 “Administrator guidance” and AGD_USR.1 “User guidance”. The dependencies are satisfied in EAL4.

AVA_VLA.4 Highly resistant

Due to the intended use of the TOE, it must be shown to be highly resistant to penetration attacks. This assurance requirement is achieved by the AVA_VLA.4 component.

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

AVA_VLA.4 has dependencies with ADV_FSP.1 “Informal functional specification”, ADV_HLD.2 “Security enforcing high-level design”, ADV_LLD.1 “Descriptive low-level design”, ADV_IMP.1 “Subset of the implementation of the TSF”, AGD_ADM.1 “Administrator Guidance”, AGD_USR.1 “User Guidance”.

All these dependencies are satisfied by EAL4.

It has to be assumed that attackers with high attack potential try to attack smart cards used for digital signature applications or payment systems. Therefore, specifically AVA_VLA.4 was chosen in order to assure that even these attackers can not successfully attack the TOE. For the same reason the Strength of Function level “SOF-high” and the specific metric strength of mechanism: high of [AIS20] are required.

The assurance components in an evaluation assurance level (EAL) are chosen in a way that they build a mutually supportive and complete set of components. The requirements chosen for augmentation do not add any dependencies, which are not already fulfilled for the corresponding requirements contained in EAL4. Therefore, these components add additional assurance to EAL4, but the mutual support of the requirements is still guaranteed.
Note that detailed refinements for assurance requirements are given in Section5.1.3

8.2.4 Mutually Supportive and Internally Consistent

This Section is described according to [BSI-PP-0002, Section 7.3].

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

The security functional requirement FPT_PHP.3 makes it harder to manipulate User Data and TSF Data. This protects the primary assets identified in Section3.1 and other security features or functions that use these data.

Though a manipulation of the TOE (refer to FPT_PHP.3) is not of great value for an attacker in itself, it can be an important step in order to threaten the primary assets identified in Section3.1. Therefore, the security functional requirement FPT_PHP.3 is not only required to meet the security objective O.Phys-Manipulation. Instead it protects other security features or functions of both the TOE and the Smartcard Embedded Software from being bypassed, deactivated or changed. In particular this may pertain to the security features or functions being specified using FDP_ITT.1, FPT_ITT.1 (1), FPT_FLS.1, FMT_LIM.2, FCS_RND.1, FDP_SDI.2, FPT_ITT.3, FPT_ITT.1 (2), and those implemented in the Smartcard Embedded Software.

A malfunction of TSF (refer to FRU_FLT.2 and FPT_FLS.1) can be an important step in order to threaten the primary assets identified in Section3.1. Therefore, the security functional requirements FRU_FLT.2 and FPT_FLS.1 are not only required to meet the security objective O.Malfunction. Instead they protect other security features or functions of both the TOE and the Smartcard Embedded Software from being bypassed, deactivated or changed. In particular this pertains to the security features or functions being specified using FDP_ITT.1, FPT_ITT.1 (1), FMT_LIM.1, FMT_LIM.2, FCS_RND.1, FDP_SDI.2, FPT_ITT.3, FPT_ITT.1 (2), and those implemented in the Smartcard Embedded Software.

In a forced leakage attack the methods described in “Malfunction due to Environmental Stress” (refer to T.Malfunction) and/or “Physical Manipulation” (refer to T.Phys-Manipulation) are used to cause leakage from signals that normally do not contain significant information about secrets. Therefore, in order to avert the disclosure of primary assets identified in Section3.1 it is important that the security functional requirements averting leakage (FDP_ITT.1, FPT_ITT.1 (1)) and those against malfunction (FRU_FLT.2 and FPT_FLS.1) and physical manipulation (FPT_PHP.3) are effective and bind well. The security features and functions against malfunction ensure correct operation of other security functions (refer to above) and help to avert forced leakage themselves in other attack scenarios. The security features and functions against physical manipulation make it harder to manipulate the other security functions (refer to above). Physical probing (refer to FPT_PHP.3) shall directly avert the disclosure of primary assets identified in Section3.1. In addition, physical probing can be an important step in other attack scenarios if the corresponding security features or functions use secret data. For instance the security functional requirement FMT_LIM.2 may use passwords. Therefore, the security functional requirement FPT_PHP.3 (against probing) help to protect other security features or functions including those being implemented in the Smartcard Embedded Software. Details depend on the implementation.

Leakage (refer to FDP_ITT.1, FPT_ITT.1 (1)) shall directly avert the disclosure of primary assets identified in Section3.1. In addition, inherent leakage and forced leakage (refer to above) can be an important step in other attack scenarios if the corresponding security features or functions use secret data. For instance the security functional requirement FMT_LIM.2 may use passwords. Therefore, the security functional requirements FDP_ITT.1 and FPT_ITT.1 (1) help to protect other security features or functions implemented in the Smartcard Embedded Software (FDP_ITT.1) or provided by the TOE (FPT_ITT.1 (1)). Details depend on the implementation.

According to the assumption Usage of Hardware Platform (A.Plat-Appl) the Smartcard Embedded Software will correctly use the functions provided by the TOE. Hereby the User Data are treated as required to meet the requirements defined for the specific application context (refer to Treatment of User Data (A.Resp-Appl)). However, the TOE may implement additional functions. This can be a risk if their interface cannot completely be controlled by the Smartcard Embedded Software. Therefore, the security functional requirements FMT_LIM.1 and FMT_LIM.2 are very important. They ensure that appropriate control is applied to the interface of these functions (limited availability) and that these functions, if being usable, provide limited capabilities only.

The combination of the security functional requirements FMT_LIM.1 and FMT_LIM.2 ensures that (especially after TOE Delivery) these additional functions can not be abused by an attacker to (i) disclose or manipulate User Data, (ii) to manipulate (explore, bypass, deactivate or change) security features or functions of the TOE or of the Smartcard Embedded Software or (iii) to enable an attack. Hereby the binding between these two security functional requirements is very important:

The security functional requirement Limited Capabilities (FMT_LIM.1) must close gaps that could be left by the control being applied to the function's interface (Limited Availability (FMT_LIM.2)). Note that the security feature or function that limits the availability can be bypassed, deactivated or changed by physical manipulation or a malfunction caused by an attacker. Therefore, if Limited Availability (FMT_LIM.2) is vulnerable[51], it is important to limit the capabilities of the functions in order to limit the possible benefit for an attacker.

The security functional requirement Limited Availability (FMT_LIM.2) must close gaps that could result from the fact that the function's kernel in principle would allow to perform attacks. The TOE must limit the availability of functions that potentially provide the capability to disclose or manipulate User Data, to manipulate security features or functions of the TOE or of the Smartcard Embedded Software or to enable an attack. Therefore, if an attacker could benefit from using such functions[52], it is important to limit their availability so that an attacker is not able to use them.

No perfect solution to limit the capabilities (FMT_LIM.1) is required if the limited availability (FMT_LIM.2) alone can prevent the abuse of functions. No perfect solution to limit the availability (FMT_LIM.2) is required if the limited capabilities (FMT_LIM.1) alone can prevent the abuse of functions. Therefore, it is correct that both requirements are defined in a way that they together provide sufficient security.

It is important to avert malfunctions of TSF and of security functions implemented in the Smartcard Embedded Software (refer to above). There are two security functional requirements

[51] or, in the extreme case, not being provided
[52] the capabilities are not limited in a perfect way (FMT_LIM.1)

which ensure that malfunctions can not be caused by exposing the TOE to environmental stress. First it must be ensured that the TOE operates correctly within some limits (Limited fault tolerance (FRU_FLT.2)). Second the TOE must prevent its operation outside these limits (Failure with preservation of secure state (FPT_FLS.1)). Both security functional requirements together prevent malfunctions. The two functional requirements must define the “limits”. Otherwise there could be some range of operating conditions that is not covered so that malfunctions may occur. Consequently, the security functional requirements Limited fault tolerance (FRU_FLT.2) and Failure with preservation of secure state (FPT_FLS.1) are defined in a way that they together provide sufficient security.

For the additional functionality included in O.Add-Functions, the security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced, O.Memory-Integrity and O.Memory-Access 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.

8.3 TOE Summary Specification Rationale

8.3.1 TOE security functions rationale

The Table 9 below shows the relationship between TOE Security Functional Requirements and Security Functions of TOE Summary Specification.
This security target (public version) removes the rational of the TOE security function, because description of the refinements includes proprietary information of the TOE.

SFR	SF.RNG	SF.DES	SF.Mal-Detect	SF.Phy-Detect	SF.Phy-Protect	SF.TEST	SF. Identification	SF.Memory-Access	SF.Memory-Scramble	SF.Memory-Verification
FRU_FLT.2			X							
FPT_FLS.1			X							
FPT_SEP.1			X							
FMT_LIM.1						X				
FMT_LIM.2						X				
FAU_SAS.1							X			
FPT_PHP.3				X	X				X	
FDP_ITT.1		X		X	X			X	X	
FDP_IFC.1		X		X	X			X	X	
FPT_ITT.1 (1)		X		X	X			X	X	
FCS_RND.1	X									
FCS_COP.1		X								
FDP_ACC.1								X		
FDP_ACF.1								X		
FMT_MSA.3								X		
FMT_MSA.1								X		
FMT_SMF.1								X		
FMT_SMR.1								X		
FDP_SDI.2										X
FPT_ITT.3										X
FPT_ITT.1 (2)										X

Table 9 Relationship between Security Requirements and Security Functions

8.3.2 Assurance measures rationale

Table 2 lists the documents describing the measures regarding the assurance requirements. This table shows that each assurance requirement is satisfied by at least one assurance measure. Actually, as it is obvious that the assurance measures are suitable to meet the TOE security assurance requirements, there is no need of further explanation to justify that the stated assurance measures are compliant with the assurance requirements.

8.4 PP Claims Rationale

This Security Target claims conformance to Protection Profile “Smartcard IC Platform Protection Profile Version 1.0, BSI-PP-0002 Version1.0”.

Additionally, this Security Target makes reference to “Smartcard Integrated Circuit Platform Augmentations Version 1.0, March 8, 2002 ”.

Furthermore, this security target adds to this PP refer to the following kinds of things:

(a) O.Add-Function and FCS_COP.1 are added to cover the organisational security policy P.Add-Functions which requires the TOE provides cryptographic services to the Smartcard Embedded Software;

(b) O.Memory-Integrity, FDP_SDI.2, FPT_ITT.3 and FPT_ITT.1 (2), are added to preserve integrity of User data and TSF data from the TOE against the additional threat T.Memory-Integrity related to modification of TSF data and User data in FRAM.

(c) O.Memory-Access, FDP_ACC.1, FDP_ACF.1, FMT_MSA.3, FMT_MSA.1, FMT_SMR.1 and FMT_SMF.1, are added to preserve confidentiality of data stored into the protected FRAM areas against the additional threat T.Memory-Access related to illegal access to protected FRAM areas.

9 Glossary and References

9.1 Vocabulary

Smartcard Embedded Software
Software embedded in a smartcard IC and not being developed by the IC Designer. The Smartcard Embedded Software is designed in Phase 1 and embedded into the Smartcard IC in Phase3.
Some part of that software may actually implement a smartcard application others may provide standard services. Nevertheless, this distinction doesn't matter here so that the Smartcard Embedded Software can be considered as being application dependent whereas the IC Dedicated Software is definitely not.

IC Dedicated Software
IC proprietary software embedded in a smartcard IC and developed by the IC Developer. This software may provide additional services to facilitate usage of the hardware. In this Security Target, this software is defined HAL-API (Hardware Abstraction Layer Application Program Interface)

IC Dedicated Test Software
That part of the IC Dedicated Software that is used to test the TOE before TOE Delivery but which does not provide any functionality thereafter.

Test Features
All features and functions (implemented by the IC Dedicated Test Software and Test circuits) that are designed to be used before TOE Delivery only and delivered as part of the TOE.

Initialisation Data
Any data defined by the TOE Manufacturer and injected into the non-volatile memory by the Integrated Circuits manufacturer (Phase 3). These data are for instance used for traceability and for TOE identification.

Pre-personalisation Data
Any data supplied by the Card Manufacturer that is injected into the non-volatile memory by the Integrated Circuits manufacturer (Phase 3). These data are for instance used for traceability and to secure shipment between phases.

Smartcard
(as used in this [*BSI-PP-0002*]) Composition of the TOE, the Smartcard Embedded Software, User Data and the package (the smartcard carrier).

TOE Delivery
The period when the TOE is delivered which is (refer to *Section 2.2*) after Phase 3 if the TOE is delivered in form of chip die.

TOE Manufacturer
The TOE Manufacturer must ensure that all requirements for the TOE and its development and production environment are fulfilled. In this ST, the TOE Manufacturer has the following roles: IC Developer and IC Manufacturer.

Card Manufacturer

The customer of the TOE Manufacturer who receives the TOE during TOE Delivery. The Card Manufacturer includes all roles after TOE Delivery up to Phase 7 (refer to *section 2.2*). The Card Manufacturer has the following roles, the IC Packaging Manufacturer, the Smartcard Product Manufacturer, the Personaliser.

TSF data
Data created by and for the TOE, that might affect the operation of the TOE (for example configuration data). Note that the TOE is the Smartcard IC.Initialisation Data defined by the Integrated Circuits manufacturer to identify the TOE and to keep track of the product's production and further life-cycle phases are also considered as belonging to the TSF data.

User
The TOE serves as a platform for the Smartcard Embedded Software. Therefore, the "user" of the TOE (as used in the Common Criteria assurance class AGD: guidance) is the Smartcard Embedded Software. Guidance is given for the Smartcard Embedded Software Developer. On the other hand the Smartcard (with the TOE as a major element) is used in a terminal where communication is performed through the ISO interface provided by the TOE. Therefore, another "user" of the TOE is the terminal (with its software).

User Data
All data managed by the Smartcard Embedded Software in the application context. User data comprise all data in the final Smartcard IC except the TSF data.

9.2 List of Abbreviations

CC	Common Criteria
EAL	Evaluation Assurance Level.
IC	Integrated circuit.
IT	Information Technology.
PP	Protection Profile.
SOF	Strength of function.
ST	Security Target.
TOE	Target of Evaluation.
TSC	TSF Scope of control.
TSF	TOE Security functions.
TSP	TOE Security Policy.
SF	Security Function
SFP	Security Function Policy
SFR	Security Function Requirement
RAM	Random Access Memory
ROM	Read Only Memory
FRAM	Ferroelectric Random Access Memory
SRAM	Static Random Access Memory
DES	Data Encryption Standard
RNG	Random Number Generator
DRNG	Deterministic Random Number Generator
CRC	Cyclic Redundancy Check
DMA	Direct Memory Access
ASK	amplitude shift keying

9.3 References

Related documents

[CC/1]
Common Criteria for Information Technology Security Evaluation
Part1: Introduction and general model, Version2.3, August 2005

[CC/2]
Common Criteria for Information Technology Security Evaluation
Part2: Functional Requirement, Version2.3, August 2005

[CC/3]
Common Criteria for Information Technology Security Evaluation
Part3: Assurance Requirement, Version2.3, August 2005

[BSI-PP-0002]
Smartcard IC Platform Protection Profile, Version1.0, BSI-PP-0002, July 2001

[SSVG Augmentation]
Smartcard Integrated Circuit Platform Augmentations, Version1.0, March 2002
Atmel, Hitachi Europe, Infineon Technologies and Philips Semiconductors

[ISO/IEC 18092]
Information technology -- Telecommunications and information exchange between systems -- Near Field Communication -- Interface and Protocol (NFCIP-1)

[FIPS PUB 46-3]
DATA ENCRIPTION STANDARD, Reaffirmed 1999 October 25

[FIPS PUB 81]
Announcing the standard for DES MODES OPERATION, 1980 December 2

[AIS20]
Functionality Classes and Evaluation Methodology for Deterministic Random Number Generators, Version 2.0, 2 December 1999, BSI

Security Target

[ST] IC Platform of FeliCa Contactless Smartcard “CXD9861/ MB94RS402”, Security Target, version 5, level 7

User Guidance

[SPC] LSI Specification, version 5, level 6
[HAL] HAL-API Function Specification, version 5, level 4
[DRN] DRNG Library, version 5, level 2

---- End of Document ----