P5CC052V0A/V0B

Security Target Lite

Rev. 1.8 — 30 July 2012
BSI-DSZ-CC-0466-2008

Evaluation Documentation
PUBLIC

Document information

Info	Content
Keywords	Security Target Lite, CC, Security Evaluation, Functional Requirements, Security Functionality, Assurance Level, P5CC052V0A/V0B
Abstract	Evaluation of the **NXP P5CC052V0A/V0B Secure Smart Card Controller** developed and provided by **NXP Semiconductors, Business Unit Identification** according to the **Common Criteria for Information Technology Evaluation (CC) at Level EAL5 augmented**

Revision history

Latest Revision: Rev. 1.8 ,30 July 2012

Rev	Date	Description	Remarks
1.8	30-July-2012	remove chapter 1.2 TOE reference	naming inconsistence
1.7	21-Feb-2012	corr. OEF/ eOEF reference in 9.4.2	
1.6	01-Nov-2011	add V0B hardware variant, TOE reference in 1.2, typo corr.	
1.5	09-Jul-2009	Generalize module package type in Table 6	
1.3-1.4	-	-	Skipped for version alignment with main ST
1.2	12-Feb-2009	Added delivery form “XD”, see Table 6, US-en, formatting, typo corr.	„silver modules“
1.1	16 April 2008	Updated table 1	
1.0	28 October 2007	Derived from ST P5CC052V0A Rev. 1.2	

Contact information

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

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

1. ST Introduction

This chapter is divided into the following sections: “ST Identification”, “ST Overview” and “CC Conformance and Evaluation Assurance Level”.

1.1 ST Identification

This Security Target Lite (st_lite_p5cc052v0a_v0b_v1_8.doc, Rev. 1.8, 30 July 2012) refers to the "NXP P5CC052V0A/V0B Secure Smart Card Controller" (TOE) provided by NXP Semiconductors, Business Line Identification for a Common Criteria evaluation.

1.2 ST Overview

1.2.1 Introduction

The TOE is the hardware of the microcontroller chip P5CC052V0A/V0B of the Smart Card Controller IC family produced by NXP. The TOE includes also IC Dedicated Test Software for test purposes and IC Dedicated Support Software, both stored in the Test-ROM of the microcontroller. The Smart Card Controller hardware comprises an 8-bit processing unit, volatile and non-volatile memories accessible via a memory management unit, cryptographic co-processors, security components and one communication interface.

The TOE includes a Data Sheet, a document describing the Instruction Set and the Guidance Document. This documentation contains a description of the architecture, the secure configuration and usage of the chip by the Smartcard Embedded Software.

The security measures of the P5CC052V0A/V0B are designed to act as an integral part of the complete security system in order to strengthen the design as a whole. Several security measures are completely implemented in and controlled by the hardware. Other security measures are controlled by the hardware and allow a configuration by software or software guided exceptions. With the different CPU modes and the memory management unit the TOE is intended to support multi-application projects.

The non-volatile EEPROM can be used as data or program memory. It contains high reliability cells which guarantee data integrity. This is ideal for applications requiring non-volatile data storage and important for the use as memory for native programs. Security functions protect data in the on-chip ROM, EEPROM and RAM. In particular when being used in the banking and finance market or in electronic commerce applications the smart card must provide high security.

Hence the TOE shall

- maintain the integrity and the confidentiality of code and data stored in the memories of it and
- maintain the different CPU modes with the related capabilities for configuration and memory access and
- maintain the integrity, the correct operation and the confidentiality of security functions (security mechanisms and associated functions) provided by the TOE.

These features are ensured by the construction of the TOE and the security functions it provides. The "NXP P5CC052V0A/V0B Secure Smart Card Controller" (TOE) mainly provides a hardware platform for a smart card with

- functions to calculate the Data Encryption Standard (Triple-DES) with up to three keys
- support for large integer arithmetic (multiplication, addition and logical) operations, suited for public key cryptography and elliptic curve cryptography
- a random number generator
- memory management control features
- cyclic redundancy check calculation (CRC)
- ISO 7816 contact interface with UART

In addition several security features independently implemented in hardware or controlled by software will be provided to ensure proper operation as well as integrity and confidentiality of stored data. This includes for example measures for memory protection and sensors to allow operation only under specified conditions.

Note: The arithmetic co-processor for large integer arithmetic operations is intended to be used for the calculation of asymmetric cryptographic algorithms. Any asymmetric cryptographic algorithm needs to be implemented in software by using the calculation functions provided by the co-processor. Therefore the co-processor without software does not provide a security function itself e.g. cryptographic support. This means that Smartcard Embedded Software that implements e.g. the RSA cryptographic algorithm is not included in the evaluation. Nevertheless the co-processor is part of the Smartcard IC and therefore a security relevant component of the TOE that must resist to the attacks mentioned in this Security Target and that must operate correctly as specified in the Data Sheet. The same scope for the evaluation is applied to the CRC module.

1.2.2 Life-Cycle

Regarding the life cycle of the smartcard (refer to the “Smartcard IC Platform Protection Profile”, [7] section 8.1), the development and the production phase of the IC with its dedicated software as described for the Target of Evaluation (TOE) is part of the evaluation.

Referring to the description in the PP [7], the TOE is delivered at the end of phase 3 or of phase 4 as described in section 2.1.

Regarding the Application Note 1 of [7] the TOE supports the authentic delivery using the FabKey feature (refer to the Data Sheet, P5xC012/02x/037/052 family Secure contact PKI smart card controller, NXP Semiconductors and the Guidance, Delivery and Operation Manual for the P5xC012/02x/037/052V0A family of Secure Smart Card Controllers).

Security during Development and Production

During the design and the layout process only people involved in the specific development project for an IC have access to sensitive data. Different people are responsible for the design data and for customer related data. The security measures installed within NXP ensure a secure computer system and provide appropriate equipment for the different development tasks.

The verified layout data is provided by the developers of NXP Semiconductors, Business Unit Identification directly to the wafer fab. The wafer fab generates and forwards the layout data related to the different photo masks to the manufacturer of the photo masks. The photo masks are generated off-site and verified against the design data of the development before the usage. The accountability and the traceability is ensured among the wafer fab and the photo mask provider.

The production of the wafers includes two different steps regarding the production flow. In the first step the wafers are produced with the fixed masks independent of the customer. After that step the wafers are completed with the customer specific mask and the remaining fixed masks. The computer tracking ensures the control of the complete process including the storage of the semi-finished wafers.

The test process of every die is performed by a test centre of NXP. Delivery processes between the involved sites provide accountability and traceability of the produced wafers. NXP embeds the dice into smartcard modules or other packages based on customer demand. Information about non-functional items is stored on magnetic/optical media enclosed with the delivery or the non-functional items are physically marked.

In summary the TOE can be delivered in three different forms:

- Dice on wafers
- Smart Card Modules on a module reel
- Packaged devices in tubes or reels

For each delivery form multiple types are supported. The different (package) types are described in detail in section 2.3.

1.2.3 Specific Issues of Smartcard Hardware and the Common Criteria

Regarding the Application Note 2 of [7] the TOE provides additional functionality which is not covered in the “Smartcard IC Platform Protection Profile”. This additional functionality is added using the policy “P.Add-Components” (see section 3.4 of this Security Target).

1.3 CC Conformance and Evaluation Assurance Level

The evaluation is based upon

- Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, Version 2.3, August 2005, CCMB-2005-08-001, [1]
- Common Criteria for Information Technology Security Evaluation – Part 2: Security functional requirements, Version 2.3, August 2005, CCMB-2005-08-002, [2]
- Common Criteria for Information Technology Security Evaluation – Part 3: Security Assurance Requirements, Version 2.3, August 2005, CCMB-2005-08-003, [3]

For the evaluation the following methodology will be used:

- Common Methodology for Information Technology Security Evaluation CEM-99/045 Part 2: Evaluation Methodology, Version 2.3, August 2005, CCMB-2005-08-004, [4]

The chosen level of assurance is **EAL 5 augmented**. The augmentations used in this Security Target are ALC_DVS.2, AVA_MSU.3 and AVA_VLA.4. The minimum strength level for the TOE security functions is **SOF-high (Strength of functions high)**.

This Security Target claims the following CC conformances:

- Part 2 extended, Part 3 conformant, EAL 5 augmented
- Conformance to the Protection Profile “Smartcard IC Platform Protection Profile”, [7]

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

Note: The “Smartcard IC Platform Protection Profile”, [7] requires the assurance level EAL4 augmented. Regarding the Application Note 3 of [7] the changes which are needed for EAL5 are described in the different relevant sections of this Security Target.

2. TOE Description

This chapter is divided into the following sections: “TOE Definition”, “Evaluated hardware configurations” and “Further Definitions and Explanations”. TOE Definition has the sub-sections “Hardware Description”, “Software Description”, “Documentation”, “Interface of the TOE”, “Life Cycle and Delivery of the TOE”, “TOE Intended Usage”, “TOE User Environment” as well as “General IT features of the TOE”.

2.1 TOE Definition

The Target of Evaluation (TOE) is the smartcard integrated circuit depicted in Fig 1 as block diagram. The TOE named P5CC052V0A/V0B is manufactured in an advanced CMOS process. The TOE includes IC Designer/Manufacturer proprietary IC Dedicated Test Software and IC Dedicated Support Software. All other software is called Smartcard Embedded Software and is not part of the TOE.

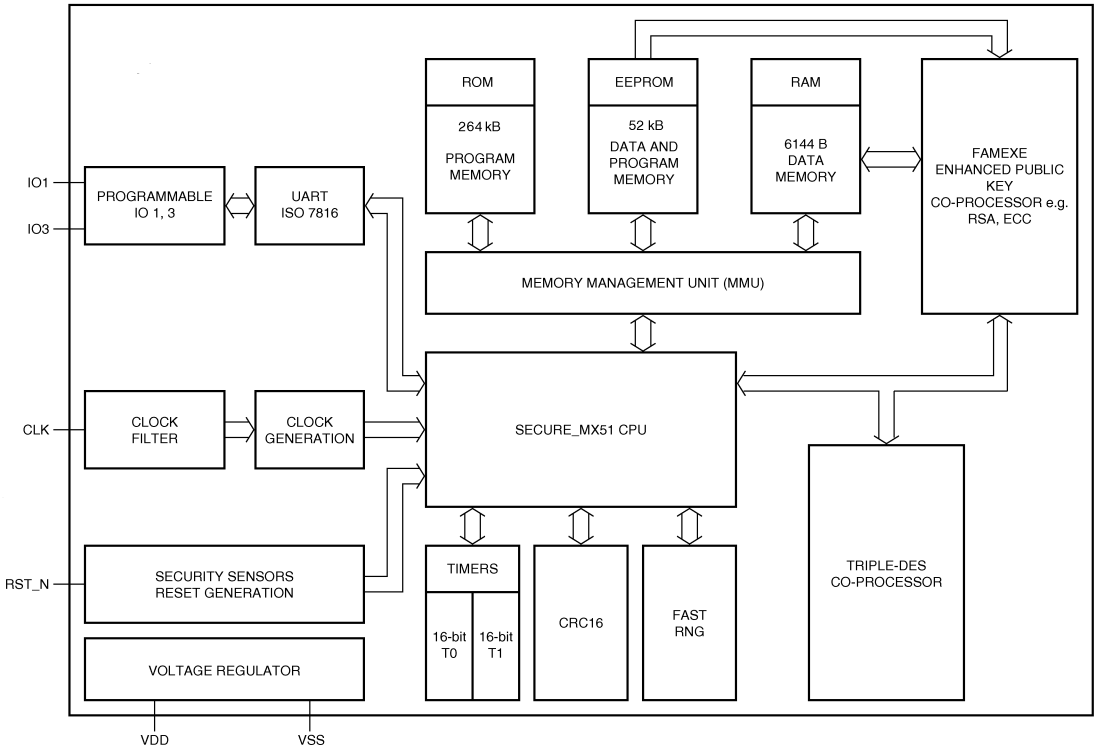

Fig 1. Block Diagram of the P5CC052V0A/V0B

Due to the possible combination of hardware and software the TOE can be configured in one of the following two combinations shown in following Table 1 and Table 2.

Each variation is clearly identified by the Typename.

The following tables list the available variants for the TOE components.

TOE components

Table 1. Components TOE variation P5C052V0A

Type	Name	Release	Date	Form of delivery
Hardware	NXP P5CC052V0A Secure Smart Card Controller	V0A	T039A_20061207 (GDS 2 File)	Wafer, modules and package (dice include reference T039A)
Software	Test ROM Software (the *IC Dedicated Test Software*)	73	Jun 26, 2007	Test ROM on the chip *(tmfos_73.lst)*
Software	Boot ROM Software (the *IC Dedicated Support Software*)	73	Jun 26, 2007	Test ROM on the chip *(tmfos_73.lst)*
Document	Data Sheet, P5xC012/02x/037/052 family Secure contact PKI smart card controller, NXP Semiconductors			Electronic document
Document	Instruction Set	1.1	July 04th, 2006	Electronic document
Document	Guidance, Delivery and Operation Manual for the P5xC012/02x/037/052V0A family of Secure Smart Card Controllers			Electronic document

Table 2. Components TOE variation P5C052V0B

Type	Name	Release	Date	Form of delivery
Hardware	NXP P5CC052V0B Secure Smart Card Controller	B	T039B_20110715 (GDS 2 File)	Wafer, modules and package (dice include reference T039B)
Software	Test ROM Software (the *IC Dedicated Test Software*)	73	Jun 26, 2007	Test ROM on the chip *(tmfos_73.lst)*
Software	Boot ROM Software (the *IC Dedicated Support Software*)	73	Jun 26, 2007	Test ROM on the chip *(tmfos_73.lst)*
Document	Data Sheet, P5xC012/02x/037/052 family Secure contact PKI smart card controller, NXP Semiconductors			Electronic document
Document	Instruction Set	1.1	July 04th, 2006	Electronic document
Document	Guidance, Delivery and			Electronic

Type	Name	Release	Date	Form of delivery
	Operation Manual for the P5xC012/02x/037/052V0A family of Secure Smart Card Controllers[1]			document

2.1.1 Hardware Description

The CPU of the P5CC052V0A/V0B has an 8-bit architecture with an instruction set that is extended from the 80C51 family instruction set. The first and in some cases the second byte of an instruction are used for operation encoding. The P5CC052V0A/V0B distinguishes between five different CPU modes, displayed in the following table.

Table 3. Different CPU modes of the TOE

Super System Mode				
Boot Mode	Test Mode	Mifare Mode	System Mode	User Mode

As shown in the table the three modes Boot Mode, Test Mode and Mifare Mode are sub-modes of the so-called Super System Mode. These three modes are not available for the Smartcard Embedded Software developer, they are reserved for the three software components that belong to the TOE (refer to the beginning of section 2.1). The mapping of modes and software components is one-to-one: In Boot Mode the TOE executes the Boot ROM Software and in Test Mode the TOE executes the Test ROM Software. The Mifare Mode is provided for compatibility purposes within the SmartMX family, the TOE has no associated software for this mode. Note that the Super System Mode is not a mode on its own: When the TOE is in Super System Mode, it is always either in Boot Mode, Test Mode or Mifare Mode, depending on the settings of an internal register not available for the Smartcard Embedded Software.

Available for the developer of the Smartcard Embedded Software are the System Mode and the User Mode. The System Mode provides unlimited access to the hardware components. In the User Mode the access is restricted to the CPU and specific Special Function Registers. Access rights to hardware components for User Mode software can be granted by software running in System Mode. The on-chip hardware components are controlled by the Smartcard Embedded Software via Special Function Registers. These registers are correlated to the activities of the CPU, the memory management unit, interrupt control, I/O configuration, EEPROM, timers, UART and the co-processors. The communication with the P5CC052V0A/V0B can be performed through an UART or the direct usage of the I/O ports. The P5CC052V0A/V0B provides two different types of interrupts: (i) exception interrupts, called “exception” in the following and (ii) event interrupts, called “interrupts” in the following. These interrupts force the jump to specific fixed vector addresses in the ROM. Every different interrupt can therefore be controlled and guided by a specific part of Smartcard Embedded Software. In conjunction with the jump to a specific fixed vector address the hardware always enables a pre-defined CPU mode, either the System Mode or the User Mode. In addition the TOE provides 32 system call vectors (SVEC) which force the System Mode. These vectors have to be explicitly called by the Smartcard Embedded Software. Special on-chip hardware protects and separates every mode, especially the Boot Mode and Test Mode, from each other.

[1] The manual is valid for the V0B hardware variant, too!

The device includes ROM (288 kByte), RAM (6144 Byte) and EEPROM (52 kByte) memory. The access control is enforced for all three memory types by a memory management unit (MMU). The memory management unit partitions every memory in two parts: The ROM is split in 264 kByte Application-ROM and 24 kByte Test-ROM. 128 Bytes of the EEPROM are always reserved for the manufacturer and cannot be read and written by the Smartcard Embedded Software. The RAM is not split and used commonly by all CPU Modes. Note that the ROM size is displayed as 264 kByte in the block diagram (Fig 1) because only 264 kBytes are available for the Smartcard Embedded Software.

In Test Mode the CPU has unrestricted access to the whole memory. In Boot Mode and Mifare Mode access is limited to the Test-ROM and the smaller part of the EEPROM (128 Byte), additionally a small amount of RAM is temporarily reserved in Boot Mode. In System Mode and User Mode the respective other parts are accessible, namely the Application-ROM and the larger part of the EEPROM. The User Mode is further restricted by the memory management unit, which can be configured in System Mode.

Note that the RAM is also furthermore split in two parts 3584 Bytes general purpose RAM and 2560 Bytes FameXE RAM. The whole RAM is accessible for the CPU, but the FameXE co-processor can only access the FameXE RAM. The FameXE can access its RAM part without control (with regard to access rights) by the memory management unit. Since the MMU does not control accesses of the FameXE, software which has access to the FameXE implicitly has access to this part of the RAM. This holds also for the EEPROM: FameXE accesses to the EEPROM are not controlled by the MMU, software which has access to the FameXE implicitly has access to this part of the EEPROM. However, the separation into parts is enforced also for the FameXE.

The Triple-DES co-processor supports single DES and Triple-DES operations. Only Triple-DES will be used in this evaluation, either in 2-key or 3-key operation. The FameXE co-processor supplies basic arithmetic functions to perform asymmetric crypto algorithms implemented by the Smartcard Embedded Software. The random generator provides true random numbers without pseudo random calculation.

The P5CC052V0A/V0B operates with a single 1.8V, 3V or 5V nominal power supply. The nominal maximum external clock frequency is 10 MHz. The micro controller can be operated with the internal clock especially to decrease the calculation time for security algorithms. The controller provides power saving modes with reduced activity: the IDLE Mode and the SLEEP Mode, which includes the CLOCK STOP Mode.

The TOE protects the secret data stored in and operated by the TOE against physical tampering. Within the composition of this TOE (with Smartcard Embedded Software comprising the operating system and the smart card application) the security functionality is only partly provided by the TOE and causes dependencies between the TOE security functions and the functions on top provided by the Smartcard Embedded Software.

2.1.2 Software Description

The smart card operating system and the application are developed by the customers and they are called Smartcard Embedded Software in the following. The Smartcard Embedded Software is stored in the Application-ROM and/or in the EEPROM and is not part of the TOE. The Smartcard Embedded Software depends on the usage of the smartcard.

The IC Dedicated Test Software (Test ROM Software) in the Test-ROM of the TOE is used by the TOE Manufacturer of the smartcard to test the functionality of the chip. The test functionality is disabled before the operational use of the smart card by disabling the Test Mode of the CPU by hardware. The IC Dedicated Test Software is developed by NXP and embedded in the Test-ROM. The IC Dedicated Test Software includes the test operating system, test routines for the various blocks of the circuitry, control flags for the status of the EEPROM's security row and shutdown functions to ensure that security relevant test operations cannot be executed illegally after phase 3.

The TOE also contains IC Dedicated Support Software which is also stored in the Test-ROM. The IC Dedicated Support Software consists of the Boot ROM Software: This software is executed after each reset of the TOE, i.e. every time when the TOE starts. It sets up the TOE and does some basic configuration.

2.1.3 Documentation

The Data Sheet [9] of the P5CC052V0A/V0B is also part of the TOE. It contains a functional description needed to develop software and guidelines for the use of security features. The instruction set of the TOE is described in a separate document [10]. Additional Guidance describes aspects of the program interface and the use of programming techniques to improve the security [11]. The provided documentation can be used by the software developer to develop the Smartcard Embedded Software.

2.1.4 Interface of the TOE

The electrical interface of the TOE are the pads to connect the lines power supply, reset input, clock input, ground and serial communication pads I/O1 and I/O3.

The software interface of the TOE depends on the CPU mode:

- In the Boot Mode the Boot ROM Software is executed which provides no interface. There is no possibility to interact with this software.
- In the Test Mode (used after production before delivery of the TOE) the logical interface that is visible on the electrical interface is defined by the IC Dedicated Test Software. This IC Dedicated Test Software (Test ROM Software) comprises the test operating system and the package of test function calls stored in the Test-ROM.
- In the System Mode and User Mode (used after TOE Delivery) the software interface is the set of instructions, the bits in the special function registers that are related to these modes and the physical address map of the CPU including memories. The access to the special function registers as well as to the memories depends on the CPU mode configured by the Smartcard Embedded Software.

  Note: The logical interface of the TOE that is visible on the electrical interface after TOE Delivery is based on the Smartcard Embedded Software developed by the software developer. The identification and authentication of the user for the different CPU modes must be controlled by the Smartcard Embedded Software.

Note that the Mifare Mode is not used by the TOE to provide functionality. Therefore it does not provide an interface.

The chip surface can be seen as an interface of the TOE, too. This interface must be taken into account regarding environmental stress e.g. like temperature and in the case of an attack where the attacker manipulates the chip surface.

Note: An external voltage and timing supply as well as a data interface are necessary for the operation of the TOE. Beyond the physical behavior the data interface is defined by the Smartcard Embedded Software.

2.1.5 Life Cycle and Delivery of the TOE

For the usage phase the P5CC052V0A/V0B chip will be implemented in a credit card sized plastic card (micro-module embedded into the plastic card) or another sealed package. The chip provides a hardware computing platform to applications and multiple applications executed by a smart card operating system. Smart card applications will be used to store secret data and calculate cryptographic functions.

The module and card embedding of the TOE provide external security mechanisms because they make it harder for an attacker to access parts of the TOE for physical manipulation.

Regarding the Application Note 4 of [7] NXP will deliver the TOE at the end of phase 3 in form of wafers or at the end of phase 4 in packaged form.

Regarding the Application Note 5 of [7] NXP will deliver the TOE with IC Dedicated Support Software. The IC Dedicated Support Software is described in section 2.1.2.

The TOE is able to control two different logical phases. After production of the chip every start-up will lead to the Test Mode and the execution of the IC Dedicated Test Software. At the end of the production test the chip the Test Mode is disabled. With disabled Test Mode every start-up of the chip will lead to the System Mode with the CPU executing the Smartcard Embedded Software.

2.1.6 TOE Intended Usage

Regarding to phase 7, the combination of the smartcard hardware and the Smartcard Embedded Software is used by the end-user. The method of use of the product in this phase depends on the application. The TOE is intended to be used in an unsecured environment that does not avoid a threat.

The device is developed for most high-end safeguarded applications, and is designed for embedding into chip cards according to ISO 7816 [16]. Usually the smart card is assigned to a single individual only although the smartcard may be expected to be used for multiple applications in a multi-provider environment. Therefore the TOE may store and process secrets of several systems that must be protected from each other. So the TOE must meet security requirements to be applied to security modules. The secret data shall be used as input for the calculation of authentication data, the calculation of signatures and the encryption of data and keys.

The software developer and the system integrators such as the terminal software developer may use samples of the TOE during the development phases for their testing purposes. It is not intended that they are able to change the behavior of the smartcard in another way than an end-user.

2.1.7 TOE User Environment

The TOE user environment is the environment from TOE Delivery to phase 7. At the phases up to 6, the TOE user environment must be a controlled environment.

In the end-user environment (phase 7) Smartcard ICs are used in a wide range of applications to assure authorized conditional access. Examples of such are Pay-TV, Banking Cards, Portable communication SIM cards, Health cards, Transportation cards. The end-user environment therefore covers a wide spectrum of very different functions, thus making it difficult to avoid and monitor any abuse of the TOE.

Note: The phases from TOE Delivery to phase 7 of the smart card life cycle are not part of the TOE construction process in the sense of this Security Target. Information about those phases is just included to describe how the TOE is used after its construction. Nevertheless the security features of the Smartcard IC hardware that are independent of the software are active at TOE Delivery and cannot be disabled by the Smartcard Embedded Software in the phases afterwards.

2.1.8 General IT features of the TOE

The TOE IT functionality consists of:

- tamper resistant data storage
- control of operation conditions to provide correct operation in the specified range
- basic cryptographic functions (Triple-DES co-processor)
- basic arithmetic functions for large integer numbers (FameXE co-processor for the calculation of public key and elliptic curve cryptography algorithms)
- physical random number generator
- memory management to separate different applications
- data communication via contact interface

2.2 Evaluated hardware configurations

The TOE does support some minor configuration options, described in the following subsection.

2.2.1 Minor configuration options

The following options can be selected by the customer:

Table 4. Evaluated minor configuration options

Name	Values	Description
EDATASCALE	10h to D0h	This value determines the size of the memory area available for the extended stack pointer. Refer to section 10.5 of [9].
Card Disable Function	Yes or No	When the Card Disable Function is enabled, the TOE can be locked completely. Once set by the Smartcard Embedded Software, the execution of the Smartcard Embedded Software is inhibited after the next reset. Refer to section 29.5 of [9].
Block ROM read instructions executed from EEPROM	Yes or No	Instructions executed from EEPROM are allowed or not to read ROM contents. Refer to section 10.1.1.9 of [9].
Inverse EEPROM Error Correction	Yes or No	If inverse error correction is activated the detection probability of fault injections to the EEPROM can be increased. Refer to section 10.9.9 of [9].
Extended Voltage Class B activated	Yes or No	If Extended Voltage Class B is activated, the usable "3V supply voltage range" is extended to lower values than the minimum Class B supply voltage 2.7 V, and Class C operation is not supported. “Class BE” supply voltage range: 2.2 V ≤ V_{DD} ≤ 3.3 V. Refer to sections 34.1.2, 35.2, 29.2.2, 5 and 33 of [9].

The values of all options listed in Table 4 can be freely chosen.

The Order Entry Form [12, 13] lists further (Controller) options which must be selected with a fixed value:

- The option “Instruction execution from RAM allowed” must be selected with “NO”.

2.3 Evaluated package types

A number of package types are supported for the TOE. Each package type has a different commercial type name. A commercial type name for the TOE has the following format:

- P5CC052*pp*/T0A*rrff*z

The commercial type name is different depending on the package type - indicated by the variable “pp” – and the Smartcard Embedded Software. - indicated by the variables “rr”, “ff” and “z”.

The variables have the following definition:

Table 5. Variable definitions for commercial type names

Variable	Definition
pp	This is a two character identifier for the package type, e.g. “UA” for a sawn wafer of 150µm thickness which electronically marked defects. The different package types are defined in the next table.
rr	ROM code number, different for every Smartcard Embedded Software
ff	FabKey number, for each Smartcard Embedded Software multiple FabKeys are supported
z	General product option, reserved for future use.

The following package types are supported in this Security Target.

Table 6. Supported package types

Code	Package Type
UA	150µm sawn wafer, inkless
Xn[1]	PCM / PDM module
HN	HVQFN32 package

[1] *n denotes an identifier (0..9, A to Z) for the sub package type*

For example, the commercial type name “P5CC052HN/T0Brrffz” denotes a P5CC052V0B in a HVQFN32 package and “P5CC052UA/T0Arrffz” denotes a P5CC052V0A on a 150µm sawn wafer inkless, which means that the defect IC are electronically marked. The available package types are listed in the P5CC052V0A/V0B datasheet. Not all package types are available for all configurations of the P5CC052V0A/V0B.

The package type does not influence the security functionality of the TOE. It does only define which pads are connected in the package and for what purpose the chip (with the appropriate package) can be used. Note that the security of the TOE is not dependent on which pad is connected or not – the connections just define how the product can be used. If the TOE is delivered as wafer the customer can choose the connection himself.

For all package types listed above the security during development and production is ensured (refer to section 1.2.2).

As already described above the complete resulting commercial type name is dependent on the customer software (Smartcard Embedded Software). In consequence this means that a full commercial product name that fits in the variable forms described in Table 6 determines that the hardware is an evaluated product, however this gives no conclusion on the software and if the software does use the proper hardware configuration as described by section 2.2.1.

2.4 Further Definitions and Explanations

Since the Security Target claims conformance to the PP “Smartcard IC Platform Protection Profile”, the concepts are used in the same sense. For the definition of terms refer to the Protection Profile [7]. This chapter does not need any supplement in the Security Target.

3. TOE Security Environment

This Security Target claims conformance to the Smartcard IC Platform Protection Profile. The Assets, Assumptions, Threats and Organizational Security Policies are completely taken from the Protection Profile. In the following only the extension of the different sections are listed. The titles of the sections that are not extended are cited here for completeness.

3.1 Description of Assets

Since this Security Target claims conformance to the PP “Smartcard IC Platform Protection Profile” [7], the assets defined in section 3.1 of the Protection Profile are applied and the assets regarding threats are clarified in this Security Target.

The assets regarding the threats are:

- logical design data, physical design data, IC Dedicated Software,
- Initialization Data and Pre-personalization Data, specific development aids, test and characterization related data, material for software development support, and photomasks
- the TOE correct operation
- the Smartcard Embedded Software
- the special functions for the communication with an external interface device, the cryptographic co-processor for Triple-DES, the FameXE co-processor for basic arithmetic functions to perform asymmetric and elliptic curve cryptographic algorithms, the random number generator
- the User Data and
- the TSF Data.

The keys for the cryptographic co-processors are seen as User Data.

3.2 Assumptions

Since this Security Target claims conformance to the PP “Smartcard IC Platform Protection Profile” [7], the assumptions defined in section 3.2 of the Protection Profile are valid for this Security Target. The following table lists the assumptions of the Protection Profile.

Table 7. Assumptions defined in the Protection Profile

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

The following additional assumptions are considered in this Security Target.

A.Check-Init — Check of initialization data by the Smartcard Embedded Software

The Smartcard Embedded Software must provide a function to check initialization data. The data is defined by the customer and injected by the TOE Manufacturer into the non-volatile memory to provide the possibility for TOE identification and for traceability.

The following assumption considers the Application Notes 8 and 9 of [7] related to the specialized encryption hardware of the TOE (refer to the augmentation paper [8]).

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

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

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

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

3.3 Threats

Since this Security Target claims conformance to the PP "Smartcard IC Platform Protection Profile" [7], the threats defined in section 3.3 of the Protection Profile are valid for this Security Target. The following table lists the threats defined by the PP:

Table 8. Threats defined by the Protection Profile

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

Considering the Application Notes 10 and 11 of [7] there are no additional high-level security concerns or additional new threats defined in this Security Target.

3.4 Organizational Security Policies

Since this Security Target claims conformance to the PP “Smartcard IC Platform Protection Profile” [7], the policy P.Process-TOE “Protection during TOE Development and Production” of the Protection Profile is applied here also.

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

The IC Developer / Manufacturer must apply the policy “Additional Specific Security Components (P.Add-Components)” as specified below.

P.Add-Components Additional Specific Security Components

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

- Triple DES encryption and decryption
- Area based Memory Access Control
- Memory separation for different software parts (including IC Dedicated Software and Smartcard Embedded Software)
- Special Function Register Access Control.
- Protection of configuration data. The TOE prevents modification of configuration data – including configuration data for TSF – after TOE delivery. This can be used to enable or disable specific blocks on the TOE.

Regarding the Application Note 12 of [7] there are no other additional policies defined in this Security Target.

4. Security Objectives

This chapter contains the following sections: “Security Objectives for the TOE” and “Security Objectives for the Environment”.

4.1 Security Objectives for the TOE

The TOE shall provide the following security objectives, taken from the Protection Profile Smartcard IC Platform Protection Profile [7]:

Table 9. Security objectives defined in the PP

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

Regarding the Application Notes 13 and 14 of [7] the following additional security objectives are defined based on additional functionality provided by the TOE as specified below.

O.HW_DES3 — Triple DES Functionality

The TOE shall provide the cryptographic functionality to calculate a Triple DES encryption and decryption to the Smartcard Embedded Software. The TOE supports directly the calculation of Triple DES with up to three keys.

Note: The TOE will ensure the confidentiality of the User Data (and especially cryptographic keys) during Triple DES operation. This is supported by O.Leak-Inherent.

O.MF_FW — MIFARE Firewall

The TOE shall provide separation between the IC Dedicated Support Software and the Smartcard Embedded Software. The separation shall comprise software execution and data access.

O.MEM_ACCESS	Area based Memory Access Control Access by processor instructions to memory areas is controlled by the TOE. The TOE decides based on the CPU mode (Boot Mode, Test Mode, Mifare Mode, System Mode or User Mode) and the configuration of the Memory Management Unit (MMU) if the requested type of access to the memory area addressed by the operands in the instruction is allowed.
O.SFR_ACCESS	Special Function Register Access Control The TOE shall provide access control to the Special Function Registers depending on the purpose of the Special Function Register or based on permissions associated to the memory area from which the CPU is currently executing code. The access control is used to restrict access to hardware components of the TOE. The possibility to define access permissions to specialized hardware components of the TOE shall be restricted to code running in System Mode.
O.CONFIG	Protection of configuration data The TOE prevents modification of configuration data – including configuration data for TSF – after TOE delivery. More specifically it shall be ensured that the configuration values determined during the test phase are fixed after TOE delivery.

4.2 Security Objectives for the Environment

According to the Protection Profile [7], the following security objectives for the environment are specified:

Table 10. Security objectives for the environment, taken from the PP

Security objective	Description	Applies to phase...
OE.Plat-Appl	Usage of Hardware Platform	Phase 1
OE.Resp-Appl	Treatment of User Data	Phase 1
OE.Process-TOE	Protection during TOE Development and Production	Phase 2 up to the TOE Delivery at the end of phase 3
OE.Process-Card	Protection during Packaging, Finishing and Personalization	Begin of phase 4 up to the end of phase 6

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

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

If the random number generator is used for leakage countermeasures, cryptographic operations (e.g. key generation) or cryptographic protocols (e.g. challenge response) these random numbers must be tested appropriately.

For multi-applications the Smartcard Embedded Software (Operating System) can implement a memory management scheme based upon security features of the TOE to ensure the separation of applications.

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

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, if asymmetric algorithms are used, it must be ensured that it is not possible to derive the private key from a related public key using the attacks defined in this Security Target. If keys are imported into the TOE and/or derived from other keys, quality and confidentiality must be maintained. This implies that appropriate key management has to be realized in the environment.

The treatment of User Data is also required when a multi-application operating system is implemented as part of the Smartcard Embedded Software on the TOE. In this case the multi-application operating system will not disclose security relevant user data of one application to another application when it is processed or stored on the TOE.

Check of initialization data

The TOE provides specific support for OE.Process-TOE that requires the TOE Manufacturer to implement measures for the unique identification of the TOE. Therefore, OE.Check-Init is defined to allow a TOE specific implementation (refer also to A.Check-Init).

OE.Check-Init	Check of initialization data by the Smartcard Embedded Software To ensure the receipt of the correct TOE, the Smartcard Embedded Software shall check a sufficient part of the pre-personalization data. This shall include at least the FabKey Data that is agreed between the customer and the TOE Manufacturer.

5. IT Security Requirements

5.1 TOE Security Requirements

This section consists of the subsections “TOE Security Functional Requirements”, “TOE Security Assurance Requirements” and “Refinements of the TOE Security Assurance Requirements”.

5.1.1 TOE Security Functional Requirements

To support a better understanding of the combination Protection Profile vs. Security Target, the TOE SFRs are presented in the following two different sections.

5.1.1.1 SFRs of the Protection Profile

Table 11 below shows all SFRs which are specified in the Protection Profile Smartcard IC Platform Protection Profile [7] (in the order of definition in the PP). Some of the SFRs are CC Part 2 extended and defined in the Protection Profile. This is shown in the third column of the table.

Table 11. SFRs taken from the PP

SFR	Title	Defined in ...
FAU_SAS.1	Audit storage	PP, Section 8.6
FCS_RND.1	Quality metric for random numbers	PP, Section 8.4
FDP_IFC.1	Subset information flow control	CC, Part 2
FDP_ITT.1	Basic internal transfer protection	CC, Part 2
FMT_LIM.1	Limited capabilities	PP, Section 8.5
FMT_LIM.2	Limited availability	PP, Section 8.5
FPT_FLS.1	Failure with preservation of secure state	CC, Part 2
FPT_ITT.1	Basic internal TSF data transfer protection	CC, Part 2
FPT_PHP.3	Resistance to physical attack	CC, Part 2
FPT_SEP.1[PP]	TSF domain separation	CC, Part 2
FRU_FLT.2	Limited fault tolerance	CC, Part 2

Note that the SFR FPT_SEP.1 from the PP is iterated to FPT_SEP.1[PP] to distinguish it from the SFR that will be defined in section 5.1.1.2. The operation just renames the SFR.

With one exception, all assignment and selection operations are performed. The exception is the left open definition of a quality metric for the random numbers required by FCS_RND.1. This assignment operation is filled in by the following statement:

FCS_RND.1	Quality metric for random numbers
FCS_RND.1.1	The TSF shall provide a mechanism to generate random numbers that meet *the requirement to provide an entropy of at least 7.976 bit in each byte* [2].
Dependencies:	No dependencies.
Note:	The entropy of the random number is measured by the Shannon-Entropy as follows: $E = -\sum_{i=0}^{255} p_i \times \log_2 p_i$, where p_i is the probability that the byte $(b_7, b_6, \mathrm{K}, b_0)$ is equal to i as binary number. Here term “bit” means measure of the Shannon-Entropy. The value “7.976” is assigned due to the requirements of AIS31, [5].

By this, all assignment/selection operations are performed. This Security Target does not perform any other/further operations than stated in the Protection Profile.

Considering the Application Note 15 of [7] in the following paragraphs the additional functions for cryptographic support and access control are defined. These SFRs are not required in the Protection Profile.

Regarding the Application Note 16 of [7] an additional generation of audit is not defined for “Limited fault tolerance” (FRU_FLT.2) and “Failure with preservation of secure state” (FPT_FLS.1).

Considering the Application Note 17 of [7] no additional requirement is defined for the TOE itself but refer to “A.Check-Init” in chapter 3.2.

5.1.1.2 Additional SFRs regarding cryptographic functionality

The (DES co-processor of the) TOE shall meet the requirement “Cryptographic operation (FCS_COP.1[DES])” as specified below.

FCS_COP.1[DES]	**Cryptographic operation**
Hierarchical to:	No other components.
FCS_COP.1.1	The TSF shall perform *encryption and decryption* [3] in accordance with a specified cryptographic algorithm *Triple Data Encryption Algorithm (TDEA)* [4] and cryptographic key sizes *of 112 or 168 bit* [5] that meet the following *list of standards* [6]: *FIPS PUB 46-3 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION DATA ENCRYPTION*

[2] [assignment: a defined quality metric]

[3] [assignment: list of cryptographic operations]

[4] [assignment: cryptographic algorithm]

[5] [assignment: cryptographic key sizes]

[6] [assignment: list of standards]

	STANDARD (DES) Reaffirmed 1999 October 25, keying options 1 and 2.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation], FCS_CKM.4 Cryptographic key destruction, FMT_MSA.2 Secure security attributes.

5.1.1.3 Additional SFRs regarding protection of configuration data

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

FPT_SEP.1[CONF]	**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:	The TOE allows to configure the TSF derived by the organizational policy P.Add-Components before TOE delivery. In this phase it is possible to enable or disable certain functional blocks of the TOE including interfaces and to limit the available memory space for the memory blocks. Before TOE delivery the configuration is fixed and cannot be changed or influenced by the Smartcard Embedded Software. Therefore the TSF maintain a security domain for its own execution that protects it from interference and tampering by the Smartcard Embedded Software.

5.1.1.4 Additional SFRs regarding access control

Access Control Policy

The hardware shall provide different CPU modes to the IC Dedicated Software and Smartcard Embedded Software. The TOE shall separate IC Dedicated Software and Smartcard Embedded Software from each other by both partitioning of memory and different CPU modes. The management of access to code and data as well as the configuration of the hardware shall be performed in respective dedicated modes. The hardware shall enforce a separation between different applications (i.e. parts of the Smartcard Embedded Software) running on the TOE. Unless explicitly granted permission, an application shall not be able to access hardware components directly to support the separation of applications.

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

The subjects are

- The **Smartcard Embedded Software** i.e. data in the memories of the TOE executed as instructions by the CPU
- The "**Test ROM Software**" as IC Dedicated Test Software
- The "**Boot ROM Software**" as IC Dedicated Support Software

The objects are

- the **memories** consisting of
  - ROM which is partitioned into Test-ROM and Application-ROM,
  - EEPROM which is partitioned into two parts. For the ease of referencing the part reserved for the IC Dedicated Support Software is called Configuration-EEPROM, the other part Application-EEPROM.
  - RAM which is not partitioned.
  - the code and data in the Memory Segments defined by the Memory Management Unit (MMU) in Application-ROM, Application-EEPROM and RAM. Note that this memory is a subset of the first three.
- the physical memory locations within the three memories that are used by the MMU for the MMU Segment Table.
- the **Special Function Registers** consisting of
  - Special Function Registers to configure the MMU segmentation. This group contains the registers that define the pointer to the MMU Segment Table.
  - Special Function Registers related to system management, a number of Special Function Registers that are intended to be used for overall system management by the operating system.
  - Special Function Registers related to testing. These Special Function Registers are reserved for testing purposes.
  - Special Function Registers related to hardware components. These Special Function Registers are used to utilize hardware components like the co-processors or the interrupt system.
  - Special Function Registers related to general CPU functionality. This group contains e.g. the accumulator, stack pointer and data pointers.

The memory operations are

- **read** data from the memory,
- **write** data into the memory and
- **execute** data in the memory.

The Special Function Register operations are

- **read** data from a Special Function Register and
- **write** data into a Special Function Register.

The security attributes are

- **CPU mode**: There are five different CPU modes based on the configuration of the Special Function Register "Program Status Word High (PSWH)" and two internal bits

defining whether the instruction is executed in the Boot Mode, Test Mode, Mifare Mode, System Mode or User Mode.

- The values of the **Special Function Registers to configure the MMU segmentation** and **Special Function Registers related to system management**. These groups contain the pointer to the MMU Segment Table and those relevant for the overall system management of the TOE, especially PSWH.
- **MMU Segment Table**: Configuration of the Memory Segments comprising access rights (read, write and execute), the virtual code memory base address of the first and last valid address, and the relocation offset to the physical memory location for each of the 64 possible Memory Segments. For every segment also the access rights to the Special Function Registers related to hardware components are defined.

In the following the term “code running” combined with a CPU mode (e.g. “code running in System Mode”) will be used to name subjects.

Note: A Memory Segment will be disabled for use if no access permissions are granted. It is not necessary to define all 64 possible Memory Segments, the MMU is capable of managing an arbitrary number of segments up to the limit of 64.

The amount of the partitioned memory for the memory types is fixed. Refer to section 2.1.1.

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

FDP_ACC.1[MEM]	**Subset access control**
Hierarchical to:	No other components.
FDP_ACC.1.1	The TSF shall enforce the *Access Control Policy* [7] on *all code running on the TOE, all memories and all memory operations* [8].
Dependencies:	FDP_ACF.1 Security attribute based access control
Application Note:	The Access Control Policy shall be enforced by implementing a MMU, which maps virtual addresses to physical addresses. The CPU always uses virtual addresses, which are mapped to physical addresses by the MMU. Prior to accessing the respective memory address, the MMU checks if the access is allowed.
FDP_ACC.1[SFR]	**Subset access control**
Hierarchical to:	No other components.
FDP_ACC.1.1	The TSF shall enforce the *Access Control Policy* [9] on *all code running on the TOE, all Special Function Registers, and all Special Function Register operations* [10].

[7] [assignment: access control SFP]

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

[9] [assignment: access control SFP]

Dependencies:	FDP_ACF.1 Security attribute based access control
Application Note:	The Access Control Policy shall be enforced by implementing hardware access control to each Special Function Register. For every access the CPU mode is used to determine if the access shall be granted or denied. In addition in User Mode the access rights to the Special Function Registers related to hardware components are provided by the MMU Segment Table. A denied read access returns “0” instead of the actual value, a denied write access is in fact ignored. The read and/or write access to a Special Function Register may be not allowed depending on the function of the register or on the CPU mode to enforce the access control policy or ensure a secure operation.

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

FDP_ACF.1[MEM]	**Security attribute based access control**
Hierarchical to:	No other components.
FDP_ACF.1.1	The TSF shall enforce the *Access Control Policy* [11] to objects based on the following: *all subjects and objects and the attributes CPU mode, the MMU Segment Table, the Special Function Registers to configure the MMU segmentation and the Special Function Registers related to system management* [12].
FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *Code executed in the Boot Mode* · *has read and execute access to all code/data in the Test-ROM,* · *has read, write and execute access to all code/data in the Configuration-EEPROM* · *has read and write access to all data in the RAM* *Code executed in the Test Mode* · *has read and execute access to all code/data in the whole ROM,* · *has read, write and execute access to all code/data in the whole EEPROM* · *has read and write access to all data in the RAM*

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

[11] [assignment: access control SFP]

[12] [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]

Code executed in the Mifare Mode

- *has read and execute access to all code/data in the Test-ROM,*
- *has read, write and execute access to all code/data in the Configuration-EEPROM*
- *has no read and write access to data in the RAM*

Code executed in the System Mode

- *has read and execute access to all code/data in the Application-ROM,*
- *has read, write and execute access to all code/data in the Application-EEPROM,*
- *has read and write access to all data in the RAM,*

Code executed in the User Mode

- *has read and/or execute access to code/data in the Application-ROM controlled by the MMU Segment Table used by the MMU,*
- *has read and/or write and/or execute access to code/data in the Application-EEPROM controlled by the MMU Segment Table used by the MMU,*
- *has read and/or write access to data in the RAM controlled by the MMU Segment Table used by the MMU.* [13]

FDP_ACF.1.3	The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *Code running in Boot Mode or Mifare Mode has read access to the Security Row stored in the Application-EEPROM. The FameXE co-processor has read access to the EEPROM and read/write access to the FameXE RAM.* [14]
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the *rules: none* [15].
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
FDP_ACF.1[SFR]	**Security attribute based access control**
Hierarchical to:	No other components.
FDP_ACF.1.1	The TSF shall enforce the *Access Control Policy* [16] to objects based on the following: *all subjects and objects and the attributes CPU mode and the MMU Segment Table* [17].

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

[14] [assignment: rules, based on security attributes, that explicitly authorize access of subjects to objects]

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

[16] [assignment: access control SFP]

FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: · *The code executed in Boot Mode is allowed to access all Special Function Register groups.* · *The code executed in Test Mode is allowed to access all Special Function Register groups.* · *The code executed in Mifare Mode is allowed to access Special Function Registers related to hardware components.* · *The code executed in System Mode is allowed to access Special Function Registers to configure the MMU segmentation, Special Function Registers related to system management and Special Function Registers related to hardware components.* · *The code executed in the User Mode is allowed to access Special Function Registers related to hardware components based on the access rights defined in the respective Memory Segment in the MMU Segment Table from which the code is actually executed* [18].
FDP_ACF.1.3	The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *In any CPU mode access to the Special Function Registers related to general CPU functionality is allowed. The Special Function Register PSWH belonging to group Special Function Registers related to system management is additionally readable in Mifare Mode and User Mode.* [19]
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the rules: *Access to Special Function Registers to configure the MMU segmentation is denied in all CPU modes except System Mode. The Special Function Registers RPT0, RPT1 and RPT2 of the group Special Function Registers related to system management are not readable. The Special Function Register RNR of the group Special Function Registers related to hardware components is read-only. The Special Function Register DKEY of the group Special Function Registers related to hardware components is not readable.* [20]
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization

[17] [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]

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

[19] [assignment: rules, based on security attributes, that explicitly authorize access of subjects to objects]

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

Implications of the Access Control Policy

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

- Code executed in the Boot Mode or the Test Mode is quite powerful and used to configure and test the TOE.
- Code executed in the Mifare Mode is separated from code executed in System Mode or User Mode. The separation is enforced by the partition of the memories provided by the MMU.
- Code executed in the System Mode can administrate the configuration of MMU, because it has access to the respective Special Function Registers. Configuration means that the code can change the address of the MMU Segment Table and also modify the contents of it (as long as the table is located in write-able memory).
- Code executed in the User Mode cannot administrate the configuration of the MMU, because it has no access to the Special Function Registers to configure the MMU segmentation. Therefore changing the pointer to the MMU Segment Table is not possible.
- It may be possible for User Mode code to modify the MMU Segment Table contents if the table itself is residing in a memory location that is part of a Memory Segment that the code has write access to.

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

FMT_MSA.3[MEM]	**Static attribute initialization**
Hierarchical to:	No other components.
FMT_MSA.3.1	The TSF shall enforce the *Access Control Policy* [21] to provide *restrictive* [22] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow *no subject* [23] to specify alternative initial values to override the default values when an object or information is created.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
Application Note:	Restrictive means here that the reset values of the Special Function Register regarding the address of the MMU Segment Table are set to zero, which effectively disables any memory segment so that no User Mode code can be executed by the CPU. Furthermore the memory partition cannot be configured at all. The TOE does not provide objects or information that can be created, since it provides access to memory areas. The

[21] [assignment: access control SFP, information flow control SFP]

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

[23] [assignment: the authorised identified roles]

	definition of objects that are stored in the TOE's memory is subject to the Smartcard Embedded Software.

FMT_MSA.3[SFR]	**Static attribute initialization**
Hierarchical to:	No other components.
FMT_MSA.3.1	The TSF shall enforce the *Access Control Policy* [24] to provide *restrictive* [25] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow *no subject* [26] to specify alternative initial values to override the default values when an object or information is created.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
Application Note:	The TOE does not provide objects or information that can be created, since no further security attributes can be derived (i.e. the set of Special Function Registers that contain security attributes is fixed). The definition of objects that are stored in the TOE's memory is subject to the Smartcard Embedded Software.

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

FMT_MSA.1[MEM]	**Management of security attributes**
Hierarchical to:	No other components.
FMT_MSA.1.1	The TSF shall enforce the *Access Control Policy* [27] to restrict the ability to *modify* [28] the security attributes *Special Function Registers to configure the MMU segmentation* [29] to *code executed in the System Mode* [30].
Dependencies:	[FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
Application Note:	The MMU Segment Table is not included in this requirement because it is located in the memory of the TOE and access to it is possible for every role that has access to the respective memory locations.

[24] [assignment: access control SFP, information flow control SFP]

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

[26] [assignment: the authorised identified roles]

[27] [assignment: access control SFP, information flow control SFP]

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

[29] [assignment: list of security attributes]

[30] [assignment: the authorised identified roles]

This component does not include any management functionality for the configuration of the memory partition. This is because the memory partition is fixed and cannot be changed after TOE delivery.

FMT_MSA.1[SFR]	**Management of security attributes**
Hierarchical to:	No other components.
FMT_MSA.1.1	The TSF shall enforce the *Access Control Policy* [31] to restrict the ability to *modify* [32] the security attributes *defined in Special Function Registers* [33] to *code executed in a CPU mode which has write access to the respective Special Function Registers* [34].
Dependencies:	[FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions

The TOE shall meet the requirement “Specification of Management Functions (FMT_SMF.1)” as specified below.

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: *Change of the CPU mode by calling a system vector (SVEC) address,* *change of the CPU mode by invoking an exception or interrupt,* *change of the CPU mode by finishing an exception/interrupt (with a RETI instruction),* *change of the CPU mode with a special LCALL/ACALL/ECALL address,* *change of the CPU mode by writing to the respective bits in the PSWH Special Function Register and* *modification of the Special Function Registers containing security attributes and* *modification of the MMU Segment Table.* [35]

[31] [assignment: access control SFP, information flow control SFP]

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

[33] [assignment: list of security attributes]

[34] [assignment: the authorised identified roles]

[35] [assignment: list of security management functions to be provided by the TSF]

Dependencies:	No dependencies
Application Note:	The iteration of FMT_MSA.1 with the dependency to FMT_SMF.1 may imply a separation of the Specification of Management Functions. Iteration of FMT_SMF.1 is not needed because all management functions rely on the same features implemented in the hardware.

5.1.1.5 SOF claim for TOE security functional requirements

Since the assurance level is augmented with AVA_VLA.4 the required level for the Strength of Function (SOF) of the above listed security functional requirements level is “SOF-high”.

5.1.2 TOE Security Assurance Requirements

Table 12 below lists all security assurance components that are valid for this Security Target. These security assurance components are required by EAL5 (see section 1.3) or by the Protection Profile.

Considering the Application Note 18 of [7] the column “Required by” shows the differences in the requirements of security assurance components between the PP and the Security Target. The entry “EAL5 / PP” denotes that a SAR is required by both EAL5 and the requirement of the PP, “EAL5” means that this requirement is due to EAL5 and beyond the requirement of the PP, and “PP” identifies this component as a requirement of the PP which is beyond EAL5. The Security Target does not include additional augmentations. The refinements of the PP “Smartcard IC Platform Protection Profile” that must be adapted for EAL5 are described in section 0.

Table 12. Security Assurance Requirements EAL5 and PP augmentations

SAR	Title	Required by
ACM_AUT.1	Partial CM automation	EAL5 / PP
ACM_CAP.4	Generation support and acceptance procedures	EAL5 / PP
ACM_SCP.3	Development tools CM coverage	EAL5
ADO_DEL.2	Detection of modification	EAL5 / PP
ADO_IGS.1	Installation, generation, and start-up procedures	EAL5 / PP
ADV_FSP.3	Semiformal functional specification	EAL5
ADV_HLD.3	Semiformal high-level design	EAL5
ADV_IMP.2	Implementation of the TSF	EAL5 / PP
ADV_INT.1	Modularity	EAL5
ADV_LLD.1	Descriptive low-level design	EAL5 / PP
ADV_RCR.2	Semiformal correspondence demonstration	EAL5
ADV_SPM.3	Formal TOE security policy model	EAL5
AGD_ADM.1	Administrator guidance	EAL5 / PP

SAR	Title	Required by
AGD_USR.1	User guidance	EAL5 / PP
ALC_DVS.2	Sufficiency of security measures	PP
ALC_LCD.2	Standardized life-cycle model	EAL5
ALC_TAT.2	Compliance with implementation standards	EAL5
ATE_COV.2	Analysis of coverage	EAL5 / PP
ATE_DPT.2	Testing: low-level design	EAL5
ATE_FUN.1	Functional testing	EAL5 / PP
ATE_IND.2	Independent testing – sample	EAL5 / PP
AVA_CCA.1	Covert channel analysis	EAL5
AVA_MSU.3	Analysis and testing for insecure states	PP
AVA_SOF.1	Strength of TOE security function evaluation	EAL5/ PP
AVA_VLA.4	Highly resistant	PP

5.1.3 Refinements of the TOE Security Assurance Requirements

The ST claims conformance to the Protection Profile “Smartcard IC Platform Protection Profile”, and therefore it has to be conform to the refinements of the TOE security assurance requirements (see Application Note 19 of the PP). Because the refinements in the PP are defined for the security assurance components of EAL4, some refinements have to be applied to assurance components of the higher level EAL5 stated in the Security Target.

Table 13 lists the influences of the refinements of the PP on the ST. Most of the refined security assurance components have the same level in both documents (Protection Profile and Security Target). The following two subsections apply the refinements to ACM_SCP.3 and ADV_FSP.3 which are different between the PP and the ST.

Table 13. Security Assurance Requirements, overview of differences of refinements

Refined in PP	Influence on ST
ACM_CAP.4	Same as in PP, refinement valid without change
ACM_SCP.2	ACM_SCP.3, refinements have to be adapted
ADO_DEL.2	Same as in PP, refinement valid without change
ADO_IGS.1	Same as in PP, refinement valid without change
ADV_FSP.2	ADV_FSP.3, refinements have to be adapted
AGD_ADM.1	Same as in PP, refinement valid without change
AGD_USR.1	Same as in PP, refinement valid without change

Refined in PP	Influence on ST
ALC_DVS.2	Same as in PP, refinement valid without change
ATE_COV.2	Same as in PP, refinement valid without change

5.1.3.1 Refinements regarding CM scope (ACM_SCP)

This Security Target requires a higher evaluation level for the CC family ACM_SCP, namely ACM_SCP.3 instead of ACM_SCP.2. The refinement of the PP regarding ACM_SCP.2 is a clarification of the configuration item “TOE implementation representation”. Since in ACM_SCP.3, the content and presentation of evidence element ACM_SCP.3.1C only adds a further configuration item to the list of items to be tracked by the CM system, the refinement can be applied without changes.

The refinement of the configuration item “TOE implementation representation” of ACM_SCP.2 can be found in section 5.1.3.3 of the Protection Profile [7] and is not cited here.

5.1.3.2 Refinements regarding functional specification (ADV_FSP)

This Security Target requires a higher evaluation level for the CC family ADV_FSP, namely ADV_FSP.3 instead of ADV_FSP.2. The refinement of the PP regarding ADV_FSP.2 is concerned with the description of the TSF and its external interfaces, the purpose and method of use of all external TSF interfaces, the complete representation of the TSF and the accuracy and completeness of the TOE SFR instantiations. The refinement is not a change in the wording of the action elements, but a more detailed definition of the above items.

Since the higher level ADV_FSP.3 requires a Functional Specification in a “semiformal style, supported by informal, explanatory text where appropriate” (ADV_FSP.3.1C) the changes only affect the style of description, the refinements can be applied without changes and are valid for ADV_FSP.3.

The refinement of the original component ADV_FSP.2 can be found in section 5.1.3.5 of the Protection Profile [7] and is not cited here.

5.2 Security Requirements for the Environment

This chapter consists of the sections Security Requirements for the IT-Environment and Security Requirements for the Non-IT-Environment

5.2.1 Security Requirements for the IT-Environment

There are no Security Requirements for the IT-Environment defined in the PP “Smartcard IC Platform Protection Profile”. The dependencies derive from the added security functional requirements for cryptographic operation (FCS_COP.1[DES]) and for Management of security attributes (FMT_MSA.1[MEM] and FMT_MSA.1[SFR]) as well as for Static attribute initialization (FMT_MSA.3[MEM] and FMT_MSA.3[SFR]) are defined as Security Requirements for the IT-Environment in this Security Target. Since the requirements must be fulfilled by the implemented Smartcard Embedded Software it is consequently seen as IT-Environment.

The dependencies of FCS_COP.1[DES] deal with cryptographic key management (CC family FCS_CKM) and import of data (CC family FDP_ITC) that is subject to the applications and cannot be provided by the hardware.

The dependency of FMT_MSA.1[MEM] and FMT_MSA.1[SFR] as well as FMT_MSA.3[MEM] and FMT_MSA.3[SFR] are related to security roles. The security roles may be realized mode-based but the associated identification of the user must be implemented by the Smartcard Embedded Software that also must define the number and behavior of the security roles.

Table 14. Security Requirements for the IT Environment

SFR	Name	Note
FDP_ITC.1 *or* FDP_ITC.2 *or* FCS_CKM.1	Import of user data without security attributes / Import of user data with security attributes / Cryptographic key generation	Any import of user data must be realized by the Smartcard Embedded Software. Although the Random Number Generator can be used to derive random numbers, the generation of keys at least require Smartcard Embedded Software to access the Random Number Generator several times to create a key.
FCS_CKM.4	Cryptographic key destruction	Keys can only be deleted by the Smartcard Embedded Software
FMT_MSA.2	Secure security attributes	The security attributes must be defined and assigned by the Smartcard Embedded Software.
FMT_SMR.1	Security roles	The hardware provides different CPU modes that shall be used by the Smartcard Embedded Software to realize the required security roles.

5.2.2 Security Requirements for the Non-IT-Environment

Since this ST claims conformance to the PP “Smartcard IC Platform Protection Profile”, the following security requirements for the Non-IT-Environment are taken from the PP:

- RE.Phase-1
- RE.Process-Card

The Security Target specifies the following additional security requirements for the Non-IT-Environment.

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

RE.Cipher	Cipher Schemata The developers of Smartcard Embedded Software must not implement routines in a way which may compromise keys when the routines are executed as part of the Smartcard Embedded Software. Performing functions which access cryptographic keys could allow an attacker to misuse these functions to gather information about the key which 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 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 an appropriate key management has to be realized in the environment.
RE.RNG	Test of Random Numbers The developers of Smartcard Embedded Software must implement test routines dependent on the usage of the random number generator. The requirements for testing the random numbers provided by the random number generator are given by the AIS31 and described in the Guidance, Delivery and Operation Manual for the P5xC012/02x/037/052V0A family of Secure Smart Card Controllers.
RE.Check-Init	Check of initialization data The Card Manufacturer shall use appropriate measures to protect and check a sufficient part of the pre-personalization data. This shall include at least the FabKey data that is part of the pre-personalization data (to prevent the use of Smartcard ICs that are not correctly tested and pre-personalized by the TOE Manufacturer).

6. TOE Summary Specification

This chapter is divided in the sections “TOE Security Functions” and “Assurance Measures”.

6.1 TOE Security Functions

The TOE Security Functions (TSF) directly correspond to the TOE security functional requirements defined in chapter 5.1.1.

The following security functions are applicable to the phases 4 to 7.

Note: Some of the security functions are configured at the end of phase 3 and all security functions are already active during the delivery from phase 3 to phase 4.

The TOE comprises additional features that are not listed as security function in the following. They do not provide a complete security function by themselves but they can be used to support security functions implemented by the Smartcard Embedded Software, e.g. the FameXE co-processor for asymmetric cryptographic algorithms or the CRC calculation for the control of data integrity.

F.RNG: Random Number Generator

The random number generator continuously produces random numbers with a length of one byte. The TOE implements the F.RNG by means of a physical hardware random number generator working stable within the limits guaranteed by F.OPC (operational conditions).

The TSF provides a hardware test functionality that can be used by the Smartcard Embedded Software to detect faults in the hardware implementing the random number generator.

According to AIS31 the random number generator claims the fulfillment of the requirements of functionality class P2. This means that the random number generator is suitable for generation of signature key pairs, generation of session keys for symmetric encryption mechanisms, random padding bits, zero-knowledge proofs and generation of seeds for DRNGs.

F.HW_DES: Triple-DES Co-processor

The TOE provides the Triple Data Encryption Algorithm (TDEA) according to the Data Encryption Standard (DES). F.HW_DES is a modular basic cryptographic function which provides the TDEA algorithm as defined by FIPS PUB 46 by means of a hardware co-processor and supports (a) the 3-key Triple-DEA algorithm according to keying option 1 and (b) the 2-key Triple DEA algorithm according to keying option 2 in FIPS PUB 46-3 [14]. The two/three 56 bit keys (112/168 bit) for the 2-key/3-key Triple DES algorithm shall be provided by the Smartcard Embedded Software. For encryption the Smartcard Embedded Software provides 8 bytes of the plain text and F.HW_DES calculates 8 bytes cipher text. The calculation output is read by the Smartcard Embedded Software. For decryption the Smartcard Embedded Software also provides 8 bytes of cipher text and F.HW_DES calculates 8 bytes plain text. The calculation output is read by the Smartcard Embedded Software.

F.OPC: Control of Operating Conditions

The function F.OPC ensures the correct operation of the TOE (functions offered by the micro-controller including the standard CPU as well as the Triple-DES co-processor, the arithmetic co-processor, the memories, registers, I/O interface and the other system peripherals) during the execution of the IC Dedicated Support Software and Smartcard Embedded Software. This includes all specific security features of the TOE which are able to provide an active response.

The TOE ensures its correct operation and prevents any malfunction using the following sub-functions: filtering of power supply and clock input as well as monitoring of power supply, the frequency of the clock and the temperature of the chip by means of sensors. There are multiple sensors for the different ISO 7816 supply voltage classes. Light sensors are distributed over the chip surface and used to detect light attacks. The thresholds allowed for these parameters are defined within the range where the TOE ensures its correct operation. Additionally to the light sensors the EEPROM provides two functions to detect light attacks. The Smartcard Embedded Software can select one function and also disable both functions of the EEPROM detection function.

Specific functional units of the TOE are equipped with special circuitry to detect a number of single fault injection attacks: The Program Counter, the stack pointer, the logic that implements the PSWH register, the DES co-processor and the FameXE co-processor.

If one of the monitored parameters is out of the specified range, either (i) a reset is forced and the actual running program is aborted or (ii) an exception is raised which interrupts the program flow and allows a reaction of the Smartcard Embedded Software. A reset is forced by the sensors for voltage, frequency, temperature and light. An exception is forced by the EEPROM light detector and the single fault injection detection circuitry. If the inverse error correction of the EEPROM is enabled (refer to section 2.2.1) the probability to detect fault injection errors increases and the error correction logic will also raise an exception if an error is detected. If the TOE is reset all components of the TOE are initialized with their reset values. In addition the TOE provides a reset cause indicator to the Smartcard Embedded Software. In the case an exception is raised an indicator for the reason of the exception is provided.

Before TOE delivery the Test Mode is disabled. In all other modes except the Test Mode the TOE enables the sensors automatically when operated. Furthermore the TOE prevents that the Smartcard Embedded Software disables the sensors. The assignment which sensor raises an exception or forces a reset is hard-wired and cannot be changed by software.

In addition, the TOE controls the specified range of the stack pointer. The stack pointer and the control logic is implemented threefold for the User Mode, System Mode and Super System Mode (comprising Boot Mode, Test Mode and Mifare Mode). In case the specified limits are reached an exception is generated.

Beside the sensors the security function comprises an additional sensor to check the high voltage for the write process to the EEPROM during every write sequence. The result of this sensor must be read from a Special Function Register and does not force an automatic event (e.g. exception).

F.PHY: Protection against Physical Manipulation

The function F.PHY protects the TOE against manipulation of (i) the hardware, (ii) the IC Dedicated Software in the ROM, (iii) the Smartcard Embedded Software in the ROM and the EEPROM, (iv) the application data in the EEPROM and RAM including the configuration data in the security row. It also protects User Data or TSF data against disclosure by physical probing when stored or while being processed by the TOE.

The protection of the TOE comprises different features within the design and construction which make reverse-engineering and tamper attacks more difficult. These features comprise dedicated shielding techniques for different components and specific encryption features for the memory blocks. The security function F.PHY supports the efficiency of other security functions.

F.PHY also supports the integrity of the EEPROM and the ROM. The EEPROM is able to correct a 1-bit error within each byte. The ROM provides a parity check. The EEPROM corrects errors automatically without user interaction, a ROM parity error forces a reset.

F.LOG: Logical Protection

The function F.LOG implements measures to limit or eliminate the information that might be contained in the shape and amplitude of signals or in the time between events found by measuring such signals. This comprises the power consumption and signals on the other pads that are not intended by the terminal or the Smartcard Embedded Software. Thereby this security function prevents the disclosure of User Data or TSF data stored and/or processed in the smartcard IC through the measurement of the power consumption and subsequent complex signal processing. The protection of the TOE comprises different features within the design that support the other security functions.

The Triple-DES co-processor includes special features to prevent SPA/DPA analysis of shape and amplitude of the power consumption and ensures that the calculation time is independent from any key and plain/cipher text.

The FameXE co-processor provides measures to prevent timing attacks on basic modular function. The calculation time of one operation depends on the lengths of the operands but not on the value of the operands, with the following exceptions: multiplication with reduction, modular inversion and modular division. These three operations have no constant timing due to correction cycles that are needed based on the calculation method. In addition special features are included to provide limitations of the capability for the analysis of shape and amplitude of the power consumption. Of course the FameXE does not realize an algorithm on its own and algorithm-specific leakage countermeasures have to be added for the FameXE.

Additional features that can be configured by the Smartcard Embedded Software comprise (i) the FameXE HIGHSEC mode which adds dummy calculations and (ii) CPU clock configurations that can be used to prevent the possibility to synchronize the internal operation with the external clock or to synchronize with the characteristics of the power consumption that can be used as trigger signal to support leakage attacks (DPA or timing attacks)

Specific features as described for the function F.PHY (e.g. the encryption features) and for the function F.OPC (e.g. the filter feature) support the logical protection.

F.COMP: Protection of Mode Control

The function F.COMP provides a control of the CPU mode for (i) Boot Mode, (ii) Test Mode and (iii) Mifare Mode. This includes the protection of electronic fuses stored in a protected memory area, the so-called “Security Row”, and the possibility to store initialization or pre-personalization data in the so-called “FabKey Area”.

The control of the CPU mode according to Boot Mode, Test Mode and Mifare Mode prevents the abuse of test functions after TOE delivery. Additionally it also ensures that features used at boot time to configure the TOE cannot be abused. The initial – but not user visible – CPU mode is the Boot Mode. Hardware circuitry determines whether the Test Mode is available or not. If it is available, the TOE starts the IC Dedicated Test Software in the Test Mode. Otherwise, the TOE switches to the System Mode – the initial user visible CPU mode – and starts execution of the Smartcard Embedded Software.

The protection of electronic fuses ensures the secure storage of configuration- and calibration data stored in the Test Mode. F.COMP protects CPU mode changes regarding Boot Mode, Test Mode and Mifare Mode in the following way: Switches from Boot Mode to Test Mode or Mifare Mode are allowed, switches from these two modes back to Boot Mode are prevented. The switch to the Test Mode is prevented after TOE delivery, therefore it is permanently disabled. F.COMP also ensures that the Boot Mode is only active during the boot phase of the TOE after every reset and cannot be invoked afterwards. Therefore, once the TOE has left the test phase and every time the TOE has started up, the Mifare Mode is the only CPU mode available when the PSWH.SSM bit is set. All three CPU modes Boot Mode, Test Mode and Mifare Mode are meant with “Super System Mode” and F.COMP controls which mode is used if the PSWH.SSM bit indicates the Super System Mode.

The protection of electronic fuses especially ensures that configuration options with regard to Security Functions cannot be changed, abused or influenced in any way. F.COMP ensures that activation or deactivation of security features cannot be influenced by the Smartcard Embedded Software so that the TSF maintain a security domain for its own execution that protects it from interference and tampering.

The TSF controls access to the Security Row, the top-most 128 Bytes of the EEPROM memory, accessible at reserved addresses within the memory map. The available EEPROM memory space for the Smartcard Embedded Software is reduced by this area. F.COMP provides three memory areas within the security row that can be used by the Smartcard Embedded Software:

- the User Read Only Area
- the User Write Protected Area and
- the User Write Once Area.

The User Read Only Area contains 32 bytes that are read-only for the Smartcard Embedded Software. The User Write Protected area contains 16 bytes that can be write-protected by the Smartcard Embedded Software on demand. The User Write Once Area contains 32 bytes in which each bit independently can be – once set to ‘1’ – not reset to ‘0’.

If the Card Disable Function is used (refer to section 2.2.1) the security function F.COMP prevents any start-up of the Smartcard Embedded Software once the Smartcard Embedded Software disables the card.

F.COMP also provides the FabKey Area in which initialization or identification data can be stored. The FabKey area does not belong to the Security Row and is not protected by hardware mechanisms. The FabKey Area as well as the Security Row can be used by F.COMP to store a unique identification for each die.

For all areas the initial values are set during chip testing and pre-personalization. They depend on the choice of the Smartcard Embedded Software developer and are included in the Order Entry Form. The User Write Protected Area and the User Write Once Area are designed to store the identification of a (fully personalized) smartcard or a sequence of events over the life cycle that can be coded by an increasing number of bits set to "one" or protecting bytes, respectively.

F.COMP limits the capabilities of the test functions and provides test personnel during phase 3 with the capability to store the identification and/or pre-personalization data and/or supplements of the Smartcard Embedded Software in the EEPROM. The security function F.COMP maintains the security domain for its own execution that protects it from interference and tampering by untrusted subjects both in the Test Mode and in the other modes. It also enforces the separation between the security domains of subjects regarding the IC Dedicated Software and the Smartcard Embedded Software.

F.MEM_ACC: Memory Access Control

F.MEM_ACC controls access of any subject (program code comprising processor instructions) to the memories of the TOE through the Memory Management Unit (MMU). Memory access is based on virtual addresses that are mapped to physical addresses. The CPU always uses virtual addresses. The Memory Management Unit performs the translation from virtual to physical addresses and the physical addresses are provided from the MMU to the memory interfaces to access the memories. The access control is performed in two ways:

- Partition of the memories: Every memory type (RAM, EEPROM, ROM) is split into two parts. In Boot Mode, Mifare Mode, System Mode and User Mode the CPU has access to only one part of each memory. In the Test Mode access to both parts is allowed in order to test the memory blocks.
- Segmentation of the memory in the User Mode: All three accessible parts (RAM, EEPROM, ROM) of the memory can be segmented into smaller areas and access rights (readable, writeable or executable) can be defined for these segments. Additionally access rights to Special Function Registers related to hardware components can be defined for code that is executed from a segment.

The memory partition is fixed and cannot be changed. It is determined during production of the TOE.

The memory segmentation can be defined in the System Mode. The segmentation is active when the CPU switches to the User Mode. The segments and the access rights to Special Function Registers related to hardware components are defined in the MMU Segment Table. The MMU Segment Table stores five values for each segment: The memory access rights, the virtual start address of the segment, the virtual end address of the segment, the address offset for the segment and the access rights for Special Function Registers accessible from within the segment. The address offset is used to relocate the segment anywhere in the memory map. The resulting address computed by the MMU is also subject to the partition of the memories. Up to 64 segments can be defined in the MMU Segment Table. Special values in the memory access rights allow to specify less segments and to distribute the MMU Segment Table in several parts.

Note that the MMU Segment Table itself is stored in the memory and therefore the table itself can be placed within a segment accessible for User Mode code.

As stated above the MMU provides information about access rights to Special Function Registers related to hardware components for code running in User Mode. This information is used by the TSF F.SFR_ACC to determine if the access is allowed or not. The access rights can be defined for up to 16 groups of Special Function Registers related to 16 peripheral components. The MMU provides the information about the access rights also in the other CPU modes: In Boot Mode, Test Mode, Mifare Mode and System Mode the MMU indicates full access to the 16 groups. Note that F.MEM_ACC only provides information about the access rights to F.SFR_ACC, the access control is enforced by F.SFR_ACC itself.

In addition F.MEM_ACC permanently checks whether the selected addresses are within the boundary of the physical implemented memory range. Access violations (i.e. access to forbidden memory addresses in User Mode) and accesses outside the boundary of the physical implemented memory range are notified by raising an exception.

F.SFR_ACC: Special Function Register Access Control

The function F.SFR_ACC controls access to the Special Function Registers and the switch between the CPU modes.

The TSF implements the access control to the Special Function Registers as specified in the Access Control Policy and the Security Functional Requirements FDP_ACC.1[SFR] and FDP_ACF.1[SFR].

Based on the function of the register or on the CPU mode, the read and/or write access for a specific Special Function Register is not allowed. Examples for this are read access to DES key register or write access to the output register of the random number generator. The TSF will ignore any operation on the Special Function Register that is not allowed. Ignored means that the write access has no influence and/or that the read access always provides a fixed return value independent of the content of the Special Function Register. Some Special Function Registers are implemented threefold, for User Mode, System Mode and Super System Mode (comprising Boot Mode, Test Mode and Mifare Mode) which by its nature separates the Special Function Registers.

F.SFR_ACC used information provided by F.MEM_ACC in order to determine access to the Special Function Registers related to hardware components. Access to all other Special Function Registers is pre-defined and cannot be changed.

This implies that the security functions F.RNG and F.HW_DES can only be used in User Mode if the access right to the respective Special Function Registers are explicitly granted by code running in the System Mode. This holds for all specific hardware components controlled by Special Function Registers belonging to the 16 groups mentioned above.

The TSF also implements mode transitions between the different CPU modes based on the PSWH Special Function Register. This Special Function Register contains two bits, the PSWH.SSM and PSWH.SM bit. The PSWH.SSM indicates one of three modes belonging to the Super System Mode, namely Boot Mode, Test Mode or Mifare Mode. The PSWH.SM bit indicates the System Mode. If both bits are zero, the CPU is in User Mode.

The following operations can switch the CPU mode:

- Call of a system vector (SVEC) call address. A call of a SVEC sets the PSWH.SM bit and enables System Mode. Calls of SVEC addresses are only allowed in User Mode, otherwise an exception will be raised.
- Execution of an exception or interrupt. Any event that leads to the execution of an exception sets the PSWH.SM bit. Interrupts can be executed in User Mode or System Mode. The Smartcard Embedded Software running in System Mode can configure this option at run time and based on this configuration PSWH.SM is modified or not.
- Return from an exception/interrupt or vector call with a RETI instruction. This will restore the value of the PSWH to the value before the event occurred. Since the User Mode is the least privileged mode, a RETI is only allowed if interrupts are allowed to execute in User Mode and an interrupt actually active, otherwise an exception will be raised.
- Execution a LCALL/ACALL/ECALL instruction with a specific address. Calls of address 0x800000 in System Mode will enable the User Mode and start execution at this (virtual) address. This is similar to a SVEC call, but no return address is pushed onto the stack.
- Direct modification of the two bits in PSWH. Hardware logic provided by F.SFR_ACC ensures that the bits can only be cleared. Therefore it is not possible for code running in User Mode to enter more privileged modes like the System Mode.

Only two modes are available to the Smartcard Embedded Software, the System Mode and the User Mode. The System Mode is the more privileged mode since it allows access to all Special Function Registers for the peripheral components and for system management (i.e. configuring the MMU, clock settings or additional features provided by F.LOG). The User Mode is the less privileged mode, but at least with regard to the peripheral components it can be made as powerful as the System Mode.

The combination of F.SFR_ACC and F.COMP ensures that the other CPU modes are not available for the Smartcard Embedded Software, but reserved for specific purposes fulfilled by the IC Dedicated Software. In addition F.MEM_ACC provides separation of the memories and access control information.

SOF claim

According to the CEM [4] a Security Target shall identify all mechanisms which can be assessed according to the assurance requirement AVA_SOF.1.

The following mechanisms contributing to these functions were identified, which can be analyzed for their permutational or probabilistic properties:

1. The output of the Random Number Generator F.RNG can be analyzed with probabilistic methods.
2. The quality of the mechanism contributing to the leakage attacks of F.LOG especially for F.HW_DES can be analyzed using probabilistic methods on power consumption of the TOE.

Therefore an explicit SOF claim of “high” is made for these mechanisms.

Note: The cryptographic algorithm of F.HW_DES can also be analyzed with permutational or probabilistic methods but that this is not in the scope of CC evaluations.

6.2 Assurance Measures

Appropriate assurance measures will be employed to satisfy the security assurance requirements defined in section 5.1.1.5. The developer will provide documents containing the measures and further information needed to examine conformance of the measures to the assurance requirements. The following table gives a mapping between the assurance requirements and the documents containing the information needed for the respective requirement either directly or referring to further documents containing this information.

Table 15. List of documents describing the measures regarding the assurance requirements

Document containing or referring the relevant information	Input evidence according to CC Part 3, which is contained or referred to in the document	Input for assurance classes and families (according to developer actions in CC Part 3)
Functional Specification, Data Sheet, Instruction Set	semiformal functional specification	ADV_FSP
	correspondence analysis between the TOE summary specification and the functional specification	ADV_RCR
Formal Model	TSP model (formal)	ADV_SPM
High Level Design, Design Report	high-level design (semiformal)	ADV_HLD
	correspondence analysis between functional specification and high-level design	ADV_RCR
Correspondence Demonstration,	low level design	ADV_LLD
	architectural description	ADV_INT

Document containing or referring the relevant information	Input evidence according to CC Part 3, which is contained or referred to in the document	Input for assurance classes and families (according to developer actions in CC Part 3)
Design Report	correspondence analysis between high-level design and low-level design	ADV_RCR
	correspondence analysis between low-level design and implementation representation	ADV_RCR
Implementation representation, Source Code	implementation representation	ADV_IMP
Quality Management Manual and Security Management Manual	configuration management documentation	ACM
	development tools documentation	ALC
	development security documentation	
	life cycle definition documentation	
	parts of the delivery documentation	ADO
Guidance, Delivery and Operation Manual, Data Sheet, Instruction Set	administrator guidance	AGD_ADM, AVA_MSU
	secure installation, generation, and start-up procedures	ADO_IGS
	user guidance	AGD_USR, AVA_MSU
	parts of the delivery documentation	ADO_DEL
Vulnerability Assessment, Design Report	vulnerability assessment	AVA
	covert channel analysis	
	strength of function claims analysis	
Test Documentation Roadmap, Verification Test, Characterization Report, Electrical Test Specification	test documentation	ATE
	test coverage analysis	
	depth of testing analysis	

7. PP Claims

This Security Target claims conformance to the following Protection Profile:

Smartcard IC Platform Protection Profile, Version 1.0, July 2001; registered and certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-PP-0002-2001, [7]

The short term for this Protection Profile used in this document is “Smartcard IC Platform Protection Profile”.

8. Rationale

This chapter contains the following sections: "Security Objectives Rationale", "Security Requirements Rationale", "

TOE Summary Specification Rationale" and "PP Claims Rationale".

8.1 Security Objectives Rationale

Section 7.1 of the Protection Profile provides a rationale how the assumptions, threats, and organizational security policies are addressed by the objectives that are subject of the PP “Smartcard IC Platform Protection Profile”. The following Table 16 reproduces the table in section 7.1 of [7].

Table 16. Security Objectives versus Assumptions, Threats or Policies

Assumption, Threat or OSP	Security Objective	Note
A.Plat-Appl	OE.Plat-Appl	(Phase 1)
A.Resp-Appl	OE.Resp-Appl	(Phase 1)
P.Process-TOE	OE.Process-TOE O.Identification	(Phase 2 – 3)
A.Process-Card	OE.Process-Card	(Phase 4 – 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	

The following Table 17 provides the justification for the additional security objectives. They are in line with the security objectives of the Protection Profile and supplement these according to the additional assumptions and organizational security policy.

Table 17. Additional Security Objectives versus Assumptions or Policies

Assumption/Policy	Security Objective	Note
P.Add-Components	O.HW_DES3 O.MF_FW O.MEM_ACCESS O.SFR_ACCESS O.CONFIG	
A.Key-Function	OE.Plat-Appl OE.Resp-Appl	(Phase 1)
A.Check-Init	OE.Check-Init	(Phase 1) and (Phase 4 – 6)

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

The justification related to the security objectives O.HW_DES3, O.MF_FW, O.MEM_ACCESS, O.SFR_ACCESS and O.CONFIG is as follows: Since these objectives requires the TOE to implement exactly the same specific security functionality as required by P.Add-Components, the organizational security policy is covered by the objectives.

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-Components. These security objectives are also valid for the additional specific security functionality since they must avert the related threats also for the components added related to the policy.

The requirements for a multi-application platform necessitate the separation of users. Therefore it is volitional that most of the security functions cannot be influenced or used in the User Mode.

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

- Compared to [7] a clarification has been made for the security objective “Usage of Hardware Platform (OE.Plat-Appl)”: If required the Smartcard Embedded Software shall use the cryptographic service of the TOE and their interface as specified. In addition, the Smartcard Embedded Software (i) must implement operations on keys (if any) in such a manner that they do not disclose information about confidential data and (ii) must configure the memory management in a way that different applications are sufficiently separated. If the Smartcard Embedded Software uses random numbers provided by the security function F.RNG these random numbers must be tested as appropriate for the intended purpose. This addition ensures that the assumption A.Key-Function is still covered by the objective OE.Plat-Appl although additional functions are being supported according to P.Add-Components.
- Compared to [7] a clarification has been made for the security objective “Treatment of User Data (OE.Resp-Appl)”: By definition cipher or plain text data and cryptographic keys are User Data. So, the Smartcard Embedded Software will

protect such data if required and use keys and functions appropriately in order to ensure the strength of cryptographic operation. Quality and confidentiality must be maintained for keys that are imported and/or derived from other keys. This implies that appropriate key management has to be realized in the environment. In addition the treatment of User Data comprises the implementation of a multi-application operating system that does not disclose security relevant User Data of one application to another one. These measures make sure that the assumption A.Key-Function is still covered by the security objective OE.Resp-Appl although additional functions are being supported according to P.Add-Components.

The justification related to the assumption "Check of initialization data by the Smartcard Embedded Software (A.Check-Init)" is as follows:

Since OE.Check-Init requires the Smartcard Embedded Software developer to implement a function assumed in A.Check-Init, the assumption is covered by the objective.

The justification of the additional policy and the additional assumptions 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 Security Requirements Rationale

8.2.1 Rationale for the security functional requirements

Section 7.2 of the PP “Smartcard IC Platform Protection Profile” provides a rationale for the mapping between security functional requirements and security objectives defined in the Protection Profile. The mapping is reproduced in the following table.

Table 18. Security Requirements versus Security Objectives

Objective	TOE Security Functional Requirements	Security Requirements for the environment
O.Leak-Inherent	FDP_ITT.1 “Basic internal transfer protection” FPT_ITT.1 “Basic internal TSF data transfer protection” FDP_IFC.1 “Subset information flow control”	RE.Phase-1 “Design and Implementation of the Smartcard Embedded Software”
O.Phys-Probing	FPT_PHP.3 “Resistance to physical attack”	RE.Phase-1 “Design and Implementation of the Smartcard Embedded Software”
O.Malfunction	FRU_FLT.2 “Limited fault tolerance FPT_FLS.1 “Failure with preservation of secure state” FPT_SEP.1[PP] “TSF domain separation”	
O.Phys-Manipulation	FPT_PHP.3 “Resistance to physical attack”	RE.Phase-1 “Design and Implementation of the Smartcard Embedded Software” (e.g. by implementing FDP_SDI.1

Objective	TOE Security Functional Requirements	Security Requirements for the environment
		Stored data integrity monitoring)
O.Leak-Forced	All requirements listed for O.Leak-Inherent FDP_ITT.1, FPT_ITT.1, FDP_IFC.1 plus those listed for O.Malfunction and O.Phys-Manipulation FRU_FLT.2, FPT_FLS.1, FPT_SEP.1[PP], FPT_PHP.3	RE.Phase-1 “Design and Implementation of the Smartcard Embedded Software”
O.Abuse-Func	FMT_LIM.1 “Limited capabilities” FMT_LIM.2 “Limited availability” plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1, FPT_SEP.1[PP]	
O.Identification	FAU_SAS.1 “Audit storage”	
O.RND	FCS_RND.1 “Quality metric for random numbers” plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1, FPT_SEP.1[PP]	RE.Phase-1 “Design and Implementation of the Smartcard Embedded Software” (e.g. by implementing FPT_AMT.1 “Abstract machine testing”)
OE.Plat-Appl		RE.Phase-1 “Design and Implementation of the Smartcard Embedded Software”
OE.Resp-Appl		RE.Phase-1 “Design and Implementation of the Smartcard Embedded Software”
OE.Process-TOE	FAU_SAS.1 “Audit storage”	Assurance Components: refer to below -
OE.Process-Card		RE.Process-Card possibly supported by RE.Phase-1

- 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 Security Target additionally defines the SFRs for the TOE that are listed in Table 19. In addition Security Requirements for the Environment are defined. The following table gives an overview, how the requirements are combined to meet the security objectives.

Table 19. Mapping of security objectives and requirements

Objective	TOE Security Functional Requirement	Security Requirements for the environment
O.HW_DES3	FCS_COP.1[DES]	RE.Phase-1 with RE.Cipher
O.MF_FW	FDP_ACC.1[MEM] FDP_ACF.1[MEM] FMT_MSA.3[MEM]	
O.MEM_ACCESS	FDP_ACC.1[MEM] FDP_ACF.1[MEM] FMT_MSA.3[MEM] FMT_MSA.1[MEM] FMT_MSA.1[SFR] FMT_SMF.1	RE.Phase-1 “Design and Implementation of the Smartcard Embedded Software” (e.g. definition of separated memory segments and sufficiently graded exception handling)
O.SFR_ACCESS	FDP_ACC.1[SFR] FDP_ACF.1[SFR] FMT_MSA.3[SFR] FMT_MSA.1[SFR] FMT_SMF.1	
O.CONFIG	FPT_SEP.1[CONF]	
OE.Plat-Appl (clarification)		RE.Phase-1 with RE.Cipher and RE.RNG
OE.Resp-Appl (clarification)		RE.Phase-1 with RE.Cipher
OE.Check-Init		RE.Check-Init

The justification related to the security objective “Triple DES Functionality” (O.HW_DES3) is as follows:

O.HW_DES3 requires the TOE to support Triple DES encryption and decryption. Exactly this is the requirement of FCS_COP.1[DES]. Therefore FCS_COP.1[DES] is suitable to meet O.HW_DES3.

The justification related to the security objective “MIFARE Firewall” (O.MF_FW) is as follows:

The security functional requirement “Subset access control (FDP_ACC.1[MEM])” with the related Security Function Policy (SFP) “Access Control Policy” exactly requires to implement a memory partition as demanded by O.MF_FW. Therefore, FDP_ACC.1[MEM] with its SFP is suitable to meet the security objective.

The security functional requirement “Security attribute based access control (FDP_ACF.1[MEM])” with the related Security Function Policy (SFP) “Access Control Policy” defines the rules to implement the partition as demanded by O.MF_FW. Therefore, FDP_ACF.1[MEM] with its SFP is suitable to meet the security objective.

The security functional requirement “Static attribute initialization (FMT_MSA.3[MEM])” requires that the TOE provide default values for the security attributes used by the memory management unit to enforce the memory partition. These default values are generated by the reset procedure and the Boot ROM Software for the related Special Function Register. Restrictive with respect to memory partition means that the partition cannot be changed at all and for the memory segmentation means that the initial setting is very restrictive since it effectively disables any memory segment. They are needed by the TOE to provide a default configuration after reset. Therefore this requirement (as dependency from FDP_ACF.1) is suitable to meet the security objective.

The security functional requirement “Management of security attributes (FMT_MSA.1)” requires that the ability to update the security attributes is restricted to privileged subject(s). No management ability is specified in the two iterations of FMT_MSA.1 that can be used to change the memory partition. Also no related management function is specified by FMT_SMF.1. Therefore the memory partition is fixed and cannot be changed any subject, which is the requirement of O.MF_FW.

The justification related to the security objective “Area based Memory Access Control (O.MEM_ACCESS)” is as follows:

The security functional requirement “Subset access control (FDP_ACC.1[MEM])” with the related Security Function Policy (SFP) “Access Control Policy” exactly requires to implement an area based memory access control as demanded by O.MEM_ACCESS. Therefore, FDP_ACC.1[MEM] with its SFP is suitable to meet the security objective.

The security functional requirement “Security attribute based access control (FDP_ACF.1[MEM])” with the related Security Function Policy (SFP) “Access Control Policy” defines the rules to implement the area based memory access control as demanded by O.MEM_ACCESS. Therefore, FDP_ACF.1[MEM] with its SFP is suitable to meet the security objective.

The security functional requirement “Static attribute initialization (FMT_MSA.3[MEM])” requires that the TOE provide default values for the security attributes used by the memory management units. Since the TOE is a hardware platform these default values are generated by the reset procedure for the related Special Function Register. They are needed by the TOE to provide a default configuration after reset. Therefore this requirement (as dependency from FDP_ACF.1) is suitable to meet the security objective.

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

Finally, the security functional requirement “Specification of Management Functions (FMT_SMF.1)” is used for the specification of the management functions to be provided by the TOE as demanded by O.MEM_ACCESS. Therefore, FMT_SMF.1 is suitable to meet the security objective.

The justification related to the security objective “Special Function Register Access Control (O.SFR_ACCESS)” is as follows:

The security functional requirement “Subset access control (FDP_ACC.1[SFR])” with the related Security Function Policy (SFP) “Access Control Policy” requires to implement access control for Special Function Register as demanded by O.SFR_ACCESS. Therefore, FDP_ACC.1[SFR] with its SFP is suitable to meet the security objective.

The access to Special Function Register is related to the CPU mode. The Special Function Register used to configure the MMU can only be accessed in the System Mode. The Special Function Register required to use hardware components like e.g. the co-processors or the Random Number Generator can be accessed in the System Mode as specified by the Security Function Policy (SFP) “Access Control Policy”. In the User Mode only Special Function Register required to run the CPU are accessible by default. In addition specific Special Function Registers related to hardware components can be made accessible for the User Mode if the MMU is configured to allow this.

The security functional requirement “Security attribute based access control (FDP_ACF.1[SFR])” with the related Security Function Policy (SFP) “Access Control Policy” exactly requires certain security attributes to implement the access control to Special Function Register as demanded by O.SFR_ACCESS. Therefore, FDP_ACF.1[SFR] with its SFP is suitable to meet the security objective.

The security functional requirement “Static attribute initialization (FMT_MSA.3[SFR])” requires that the TOE provides default values for the Special Function Register (values as well as access control). The default values are needed to ensure a defined setup for the operation of the TOE. Therefore this requirement (as dependency from FDP_ACF.1) is suitable to meet the security objective.

The security functional requirement “Management of security attributes (FMT_MSA.1[SFR])” is realized in a way that – besides the definition of access rights to Special Function Registers related to hardware components in User Mode and Mifare Mode - no management of the security attributes is possible because the attributes are implemented in the hardware and cannot be changed.

Finally, the security functional requirement “Specification of Management Functions (FMT_SMF.1)” is used for the specification of the management functions to be provided by the TOE as demanded by O.SFR_ACCESS. Therefore, FMT_SMF.1 is suitable to meet the security objective.

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

The justification related to the security objective “Protection of configuration data” (O.CONFIG) is as follows:

O.CONFIG requires the TOE to protect configuration data after TOE delivery. Exactly this is the requirement of FPT_SEP.1[CONF]. Therefore FPT_SEP.1[CONF] is suitable to meet O.CONFIG.

The justification related to the clarification of the security objectives “Usage of Hardware Platform (OE.Plat-Appl)” and “Treatment of User Data (OE.Resp-Appl)” is as follows:

The usage of cryptographic algorithms requires to use appropriate keys. Otherwise they do not provide security. RE.Cipher requires that keys 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.

RE.Cipher addresses the usage of keys generated inside the Smartcard IC as well as keys downloaded into the Smartcard IC. If keys are generated by the Smartcard Embedded Software using the security function F.RNG these random numbers must be tested since F.RNG does include hardware tests that have to be supplemented with statistical tests. The required test effort depends on the intended usage of the random numbers. The requirements RE.Cipher and RE.RNG for the usage of appropriate cryptographic keys for the cryptographic functions and strong random numbers are suitable to meet OE.Plat-Appl and OE.Resp-Appl.

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. Using a multi-application operating system may add additional requirements for the separation of different applications by a memory management scheme based upon security mechanisms of the TOE. These issues are addressed by the requirement RE.Phase-1. The Smartcard Embedded Software must implement additional measures regarding RE.Phase-1 defined in [7] (refer to the third point of the enumeration under RE.Phase-1 "findings of the TOE evaluation reports relevant for the Smartcard Embedded Software"). These measures are addressed in the Guidance, Delivery and Operation Manual for the P5xC012/02x/037/052V0A family of Secure Smart Card Controllers.

In addition RE.Phase-1 requires beside the specified usage of all security functions the treatment of User Data that means security relevant user data of one application cannot be disclosed to another application when a multi-application operating system is implemented as part of the Smartcard Embedded Software. Therefore the developer of the Smartcard Embedded Software shall design mainly the operating system in a way that user data cannot be disclosed to an unauthorized subject.

The justification related to the security objective for the environment “Check of initialization data by the Smartcard Embedded Software (OE.Check-Init)” is as follows:

RE.Check-Init requires at least to check the FabKey data that is part of the pre-personalization data to prevent the use of Smartcard ICs that are not correctly tested and pre-personalized by the TOE Manufacturer. The FabKey may comprise secret information that is exchanged between the Card Manufacturer and the TOE Manufacturer. F.COMP supports the storage of the FabKey data at the end of the test phase in the Test Mode. The Smartcard Embedded Software is able to check this data in the System Mode or User Mode. Therefore RE.Check-Init is suitable to meet OE.Check-Init.

The justification of the additional security objective and the additional requirements (both Security Functional Requirements and Security Requirements for the 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

The dependencies listed in the Protection Profile [7] are independent form the additional dependencies listed in the table below. The dependency of the Protection Profile is fulfilled within the Protection Profile and at least one dependency is considered to be satisfied.

The following discussion demonstrates how the dependencies defined by Part 2 of the Common Criteria for the requirements specified in sections 5.1.1.2, 5.1.1.3 and 5.1.1.4 are satisfied.

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

Table 20. Dependencies of security functional requirements

Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
FCS_COP.1[DES]	FDP_ITC.1, or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4 FMT_MSA.2	Yes (by the environment)
FPT_SEP.1[CONF]	None	Not applicable
FDP_ACC.1[MEM]	FDP_ACF.1	Yes, by FDP_ACF.1[MEM]
FDP_ACC.1[SFR]	FDP_ACF.1	Yes, by FDP_ACF.1[SFR]
FDP_ACF.1[MEM]	FDP_ACC.1 FMT_MSA.3	Yes, by FDP_ACC.1[MEM] Yes
FDP_ACF.1[SFR]	FDP_ACC.1 FMT_MSA.3	Yes, by FDP_ACC.1[SFR] Yes
FMT_MSA.3[MEM]	FMT_MSA.1 FMT_SMR.1	Yes, by FMT_MSA.1[MEM] See discussion below
FMT_MSA.3[SFR]	FMT_MSA.1 FMT_SMR.1	Yes, by FMT_MSA.1[SFR] See discussion below
FMT_MSA.1[MEM]	FDP_ACC.1 or FDP_IFC.1 FMT_SMR.1 FMT_SMF.1	Yes, by FDP_ACC.1[MEM] See discussion below Yes
FMT_MSA.1[SFR]	FDP_ACC.1 or FDP_IFC.1 FMT_SMR.1 FMT_SMF.1	Yes, by FDP_ACC.1[SFR] See discussion below Yes

The developer of the Smartcard Embedded Software must ensure that the additional security functional requirement FCS_COP.1[DES] is used as specified and that the User

Data processed by the related security function is protected as defined for the application context. These issues are addressed by the requirement RE.Phase-1.

The dependent requirements of FCS_COP.1[DES] completely address the appropriate management of cryptographic keys used by the specified cryptographic function and the management of access control rights as specified for the memory access control function. All requirements concerning these management functions shall be fulfilled by the environment (Smartcard Embedded Software) according to the requirements RE.Phase-1 and RE.Cipher.

The functional requirements [FDP_ITC.1, or FDP_ITC.2 or FCS_CKM.1], FCS_CKM.4 and FMT_MSA.2 are not included in this Security Target since the TOE only provides a pure engine for encryption and decryption without additional features for the handling of cryptographic keys. These security functional requirements are explicitly moved to the "Security Requirements for the IT-Environment" because the Smartcard Embedded Software is seen as "IT-Environment" that must fulfill these requirements related to the needs of the realized application.

The dependency FMT_SMR.1 introduced by the two components FMT_MSA.1 and FMT_MSA.3 is also addressed by the requirement RE.Phase-1 and more specific by the security functional requirements as stated in the chapter "Security Requirements for the IT-Environment". The definition and maintenance of the roles that act on behalf of the functions provided by the hardware must be subject of the Smartcard Embedded Software.

8.2.3 Rationale for the Assurance Requirements and the Strength of Function Level

The selection of assurance components is based on the underlying Protection Profile [7]. The Security Target uses the same augmentations as the PP, but chooses a higher assurance level. The level EAL5 is chosen in order to meet assurance expectations of digital signature applications and electronic payment systems. Additionally, the requirement of the PP to choose at least EAL4 is fulfilled.

The rationale for the augmentations is the same as in the PP. The assurance level EAL5 is an elaborated pre-defined level of the CC, part 3 [3]. The assurance components in an EAL level are chosen in a way that they build a mutually supportive and complete set of components. The requirements chosen for augmentation do not add any dependencies, which are not already fulfilled for the corresponding requirements contained in EAL 5. Therefore, these components add additional assurance to EAL 5, but the mutual support of the requirements is still guaranteed.

As stated in the Protection Profile, section 7.2.3, 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 by the PP in order to assure that even these attackers cannot successfully attack the TOE. For the same reason the Strength of Function level “high” is required.

Note that for the augmentation to EAL5 the document “Smartcard Integrated Circuit Platform Augmentations” [8] as supposed by Application Note 21 was considered regarding assurance requirements, but no additional assurance requirements are proposed in the document.

8.2.4 Security Requirements are Mutually Supportive and Internally Consistent

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 show that the security functional and assurance requirements support each other and that there are no inconsistencies between these groups.

The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced also protect the cryptographic algorithms and the memory access/separation control function as well as the access control to Special Function Register implemented according to the security functional requirements FCS_COP.1[DES] and FDP_ACC.1[MEM], FDP_ACC.1[SFR] with reference to the Access Control Policies defined in FDP_ACF.1[MEM] and FDP_ACF.1[SFR]. Therefore, these security functional requirements support the secure implementation and operation of FCS_COP.1[DES] and of FDP_ACC.1 with FDP_ACF.1 as well as the dependent security functional requirements.

A smartcard platform requires Smartcard Embedded Software to build a secure product. Thereby the Smartcard Embedded Software must support the security functions of the hardware and implement a sufficient management of the security functions implemented in the hardware. The realization of the Security Functional Requirements within the TOE provides a good balance between flexible configuration and restrictions to ensure a secure behavior of the TOE.

8.3 TOE Summary Specification Rationale

8.3.1 Rationale for TOE security functions

The following table provides a mapping of TSF to SFR. The mapping is described in detail in the text following the table (only in the full version of the Security Target).

Table 21. Mapping of Security Functional Requirements and the TOE Security Functions

	F.RNG	F.HW_DES	F.OPC	F.PHY	F.LOG	F.COMP	F.MEM_ACC	F.SFR_ACC
FAU_SAS.1				X		X		
FCS_RND.1	X			X				
FDP_IFC.1				X	X			
FDP_ITT.1				X	X			
FMT_LIM.1				X		X		
FMT_LIM.2				X		X		
FPT_FLS.1			X	X				
FPT_ITT.1				X	X			
FPT_PHP.3				X				
FPT_SEP.1[PP]			X	X		X		
FRU_FLT.2			X	X				
FCS_COP.1[DES]		X		X				
FPT_SEP.1[CONF]						X		
FDP_ACC.1[MEM]				X			X	
FDP_ACC.1[SFR]				X				X
FDP_ACF.1[MEM]				X			X	
FDP_ACF.1[SFR]				X				X
FMT_MSA.1[MEM]				X			X	
FMT_MSA.1[SFR]				X				X
FMT_MSA.3[MEM]				X			X	
FMT_MSA.3[SFR]				X				X
FMT_SMF.1				X			X	X

The "X" means that the TOE Security Function realizes or supports the functionality required by the respective Security Functional Requirement.

As already stated in the definition of the security function there are additional security features that can contribute to the security of the TOE when they are sufficiently controlled by the Smartcard Embedded Software. The CRC-component can be used to verify the integrity of memory areas defined by the Smartcard Embedded Software, the FameXE co-processor can be used to build leakage-resistant asymmetric crypto algorithms.

8.3.2 Rationale for assurance measures

The assurance measures defined in section 6.2 are considered to fulfill the assurance requirements of the CC [3] level EAL5. Since the Protection Profile defines assurance measures that are suitable to fulfill the requirements of EAL4, all input deliverables as listed in section 6.2 shall be sufficient to fulfill the assurance requirements of the PP. The assurance measures are defined especially for the development and production of Smartcard ICs and observe also the refinements made in the PP.

As already explained in the Protection Profile, annex 8.1, the development and production process of a smartcard IC is complex. Regarding the great number of assurance measures, a detailed mapping of the assurance measures to the assurance requirements is beyond the scope of this Security Target. Nevertheless the suitability of the assurance measures is subject of different evaluation tasks. The documents "Quality Management Manual" and "Security Management Manual" describe the general benchmark of NXP.

8.4 PP Claims Rationale

According to chapter 7 this Security Target claims conformance to the Protection Profile “Smartcard IC Platform Protection Profile, Version 1.0, July 2001; registered and certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-PP-0002-2001” [7].

The sections of this document where threats, objectives and security requirements are defined, clearly state which of these items are taken from the Protection Profile and which are added in this ST. Therefore this is not repeated here. Moreover all additional stated items in this ST do not contradict to the items included from the PP (see the respective sections in this document). The operations done for the SFRs taken from the PP are also clearly indicated.

The evaluation assurance level claimed for this target (EAL5+) is shown in section 5.1.1.5 to include respectively exceed the requirements claimed by the PP (EAL4+).

These considerations show that the Security Target correctly claims conformance to the Smartcard IC Platform Protection Profile, [7].

9. Annexes

9.1 Further Information contained in the PP

The Annex of the Protection Profile ([7], chapter 9) provides further information. Section 8.1 of the PP describes the development and production process of smartcards, containing a detailed life-cycle description and a description of the assets of the Integrated Circuits Designer/Manufacturer. Section 8.2 is concerned with security aspects of the Smartcard Embedded Software (further information regarding A.Resp-Appl and examples of specific Functional Requirements for the Smartcard Embedded Software). Section 8.3 gives examples of Attack Scenarios.

9.2 Glossary and Vocabulary

Note: To ease understanding of the used terms the glossary of the Protection Profile [7] is included here.

Administrator	(in the sense of the Common Criteria) The TOE may provide security functions which can or need to be administrated (i) by the Smartcard Embedded Software or (ii) using services of the TOE after delivery to Phases 4-6. Then a privileged user (in the sense of the Common Criteria, refer to definition below) becomes an administrator.
Boot Mode	CPU mode of the TOE dedicated to the start-up of the TOE after every reset. This mode is not accessible for the Smartcard Embedded Software.
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 [7], Figure 4 on page 17 and Section 8.1.1). The Card Manufacturer has the following roles (i) the Smartcard Product Manufacturer (Phase 5) and (ii) the Personalizer (Phase 6). If the TOE is delivered after Phase 3 in form of wafers or sawn wafers (dice) he has the role of the IC Packaging Manufacturer (Phase 4) in addition.
CPU mode	Mode in which the CPU operates. The TOE supports five modes, the Boot Mode, Test Mode, Mifare Mode, System Mode and User Mode.
Exception interrupts	Non-maskable interrupt of program execution starting from fixed (depending on exception source) addressees and enabling the System Mode. The sources of exceptions are: hardware breakpoints, single fault injection detection, illegal instructions, stack overflow, unauthorized system, User Mode execution of RETI instruction and access violations or collisions.
FabKey Area	A memory area in the EEPROM that contains data that is programmed during testing by the IC Manufacturer. The amount of data and the type of information can be selected by the customer.

Integrated Circuit (IC)	Electronic component(s) designed to perform processing and/or memory functions.
IC Dedicated Software	IC proprietary software embedded in a smartcard IC (also known as IC firmware) and developed by the IC Developer. Such software is required for testing purpose (IC Dedicated Test Software) but may provide additional services to facilitate usage of the hardware and/or to provide additional services (IC Dedicated Support Software).
IC Dedicated Support Software	Part of the IC Dedicated Software (refer to above) which provides functions after TOE Delivery. The usage of parts of the IC Dedicated Software might be restricted to certain phases.
IC Dedicated Test Software	Part of the IC Dedicated Software (refer to above) which is used to test the TOE before TOE Delivery but which does not provide any functionality thereafter.
Initialization 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 (identification data).
Memory	The memory comprises of the RAM, ROM and the EEPROM of the TOE.
Memory Management Unit	The MMU maps the virtual addresses used by the CPU into the physical addresses of the RAM, ROM and EEPROM. The mapping is determined by (a) the memory partition and (b) the memory segments in User Mode. Up to 64 memory segments are supported for the User Mode, whereas the memory partition is fixed. Each segment can be individually (i) positioned and sized (ii) enabled or disabled, (iii) controlled by access permissions for read, write and execute and (iv) assigns access rights for Special Function Registers related to hardware components for code executed in User Mode from this segment.
Memory Segment	Address spaces provided by the Memory Management Unit based on its configuration (the MMU Segment Table). The memory segments define which memory areas are accessible for code running in User Mode. They are located in RAM, ROM and EEPROM.
MIFARE	Contact-less smart card interface standard, complying with ISO14443A.
Mifare Mode	CPU mode of the TOE dedicated for the execution of IC Dedicated Support Software, e.g. a MIFARE Operating System. This mode is not accessible for the Smartcard Embedded Software.
MMU Segment Table	This structure defines the segments that the Memory Management Unit will used for code running in User Mode. The structure can be located anywhere in the available memory for System Mode code. It also contains access rights for Special Function Registers related to hardware components for User Mode code.

Pre-personalization 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/or to secure shipment between phases.
Security Row	Top-most 128 bytes of the EEPROM memory reserved for configuration purposes as well as dedicated memory area for the Smartcard Embedded Software to store life-cycle information about the TOE.
Smartcard	(as used in the Protection Profile [7]) Composition of the TOE, the Smartcard Embedded Software, User Data and the package (the smartcard carrier).
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 Phase 3 or in later phases of the smartcard product life-cycle. 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.
Special Function Registers	Registers used to access and configure the functions for the communication with an external interface device, the cryptographic co-processors for Triple-DES, the FameXE co-processor for basic arithmetic functions to perform asymmetric cryptographic algorithms, the random numbers generator and chip configuration.
Super System Mode	This mode represents either the Boot Mode, Test Mode or Mifare Mode.
System Mode	The System Mode has unlimited access to the hardware resources (with respect to the memory partition). The Memory Management Unit can be configured in this mode.
Test Features	All features and functions (implemented by the IC Dedicated Test Software and/or hardware) which are designed to be used before TOE Delivery only and delivered as part of the TOE.
Test Mode	CPU mode for configuration of the TOE executing the IC Dedicated Test Software. The Test Mode is permanently and irreversible disabled after production testing. In the Test Mode specific Special Function Registers are accessible for test purposes.
TOE Delivery	The period when the TOE is delivered which is (refer to [7], Figure 4 on page 17) either (i) after Phase 3 (or before Phase 4) if the TOE is delivered in form of wafers or sawn wafers (dice) or (ii) after Phase 4 (or before Phase 5) if the TOE is delivered in form of modules.

TOE Manufacturer	The TOE Manufacturer must ensure that all requirements for the TOE and its development and production environment are fulfilled (refer to [7], Figure 4 on page 17). The TOE Manufacturer has the following roles: (i) IC Developer (Phase 2) and (ii) IC Manufacturer (Phase 3). If the TOE is delivered after Phase 4 in form of modules, he has the role of the (iii) IC Packaging Manufacturer (Phase 4) in addition.
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. Initialization 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	(in the sense of the Common Criteria) 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.
User Mode	The User Mode has access to the memories under control of the Memory Management Unit. The access to the Special Function Registers is limited.

9.3 List of Abbreviations

CC	Common Criteria Version 2.3.
CPU	Central Processing Unit
DEA	Data Encryption Algorithm.
DES	Data Encryption Standard.
DRNG	Deterministic Random Number Generator
EAL	Evaluation Assurance Level.
ECC	Elliptic Curve Cryptography
IC	Integrated circuit.
IT	Information Technology.
MMU	Memory Management Unit
MX	Memory eXtension
NDA	Non Disclosure Agreement.
PKC	Public Key Cryptography
PP	Protection Profile.
PSW(H)	Program Status Word (High byte)
SAR	Security Assurance Requirement.
SF	Security function.
SFR	as abbreviation of the CC term: Security Functional Requirement, as abbreviation of the technical term of the SmartMX-family: Special Function Register[36]
SIM	Subscriber Identity Module.
SOF	Strength of function.
ST	Security Target.
TOE	Target of Evaluation.
TRNG	True Random Number Generator
TSC	TSF Scope of control.
TSF	TOE Security functions.
TSFI	TSF Interface.
TSP	TOE Security Policy.
UART	Universal Asynchronous Receiver and Transmitter.

[36] This security target does not use SFR as abbreviation of Special Function Register in the explanatory text to avoid confusion. However, the abbreviation is used in objective or security function identifiers and to distinct iterations.

9.4 Bibliography

9.4.1 Evaluation Documents

[1] Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, Version 2.3, August 2005, CCMB-2005-08-001

[2] Common Criteria for Information Technology Security Evaluation – Part 2: Security functional requirements, Version 2.3, August 2005, CCMB-2005-08-002

[3] Common Criteria for Information Technology Security Evaluation – Part 3: Security Assurance Requirements, Version 2.3, August 2005, CCMB-2005-08-003

[4] Common Methodology for Information Technology Security Evaluation CEM-99/045 Part 2: Evaluation Methodology, Version 2.3, August 2005, CCMB-2005-08-004

[5] Anwendungshinweise und Interpretationen zum Schema, AIS31: Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren, Version 1, 25.09.2001, Bundesamt für Sicherheit in der Informationstechnik

[6] Anwendungshinweise und Interpretationen zum Schema, AIS32: Übernahme international abgestimmter CC-Interpretationen ins deutsche Zertifizierungsschema, Version 1, 02.07.2001, Bundesamt für Sicherheit in der Informationstechnik

[7] Smartcard IC Platform Protection Profile, Version 1.0, July 2001; registered and certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-PP-0002-2001

[8] Smartcard Integrated Circuit Platform Augmentations, Version 1.00, March 8th, 2002

9.4.2 Developer Documents

[9] **Data Sheet, P5xC012/02x/037/**052 family Secure contact PKI smart card controller, NXP Semiconductors

[10] Instruction Set, SmartMX-Family**,** Secure and PKI Smart Card Controller, Philips Semiconductors, Revision 1.1, Document Number: 084111, July 04th, 2006

[11] **Guidance, Delivery and Operation Manual** for the P5xC012/02x/037/052V0A family of Secure Smart Card Controllers

[12] **Order Entry Form, P5CC052, NXP** Semiconductors, Business Line Identification

[13] **Electronic Order Entry Form, P5CC052,** NXP Semiconductors Extranet (https://extranet.nxp.com)

9.4.3 Other Documents

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

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

[16] ISO/IEC 7816-2:1996 Information technology – Identification cards – Integrated circuit(s) cards with contacts – Part 2: Dimensions and location of contacts

[17] ISO/IEC 7816-3:1997 Information technology – Identification cards – Integrated circuit(s) cards with contacts – Part 3: Electronic signals and transmission protocols

10. Legal information

10.1 Definitions

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

10.2 Disclaimers

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

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

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

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

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

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

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

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

10.3 Licenses

ICs with DPA Countermeasures function

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

10.4 Trademarks

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

FabKey — is a trademark of NXP B.V.

MIFARE — is a trademark of NXP B.V.

MIFARE Plus — is a trademark of NXP B.V.

MIFARE FleX — is a trademark of NXP B.V.

11. Contents

1. ST Introduction 3
1.1 ST Identification 3
1.2 ST Overview 3
1.2.1 Introduction 3
1.2.2 Life-Cycle 4
1.2.3 Specific Issues of Smartcard Hardware and the Common Criteria 5
1.3 CC Conformance and Evaluation Assurance Level 6
2. TOE Description 7
2.1 TOE Definition 7
2.1.1 Hardware Description 9
2.1.2 Software Description 10
2.1.3 Documentation 11
2.1.4 Interface of the TOE 11
2.1.5 Life Cycle and Delivery of the TOE 12
2.1.6 TOE Intended Usage 13
2.1.7 TOE User Environment 13
2.1.8 General IT features of the TOE 13
2.2 Evaluated hardware configurations 14
2.2.1 Minor configuration options 14
2.3 Evaluated package types 14
2.4 Further Definitions and Explanations 16
3. TOE Security Environment 17
3.1 Description of Assets 17
3.2 Assumptions 17
3.3 Threats 18
3.4 Organizational Security Policies 19
4. Security Objectives 20
4.1 Security Objectives for the TOE 20
4.2 Security Objectives for the Environment 21
5. IT Security Requirements 23
5.1 TOE Security Requirements 23
5.1.1 TOE Security Functional Requirements 23
5.1.1.1 SFRs of the Protection Profile 23
5.1.1.2 Additional SFRs regarding cryptographic functionality 24
5.1.1.3 Additional SFRs regarding protection of configuration data 25
5.1.1.4 Additional SFRs regarding access control 25
5.1.1.5 SOF claim for TOE security functional requirements 34
5.1.2 TOE Security Assurance Requirements 34
5.1.3 Refinements of the TOE Security Assurance Requirements 35
5.1.3.1 Refinements regarding CM scope (ACM_SCP) 36
5.1.3.2 Refinements regarding functional specification (ADV_FSP) 36
5.2 Security Requirements for the Environment 37
5.2.1 Security Requirements for the IT-Environment. 37
5.2.2 Security Requirements for the Non-IT-Environment 38
6. TOE Summary Specification 39
6.1 TOE Security Functions 39
6.2 Assurance Measures 46
7. PP Claims 48
8. Rationale 48
8.1 Security Objectives Rationale 49
8.2 Security Requirements Rationale 51
8.2.1 Rationale for the security functional requirements 51
8.2.2 Dependencies of security functional requirements 57
8.2.3 Rationale for the Assurance Requirements and the Strength of Function Level 58
8.2.4 Security Requirements are Mutually Supportive and Internally Consistent 59
8.3 TOE Summary Specification Rationale 60
8.3.1 Rationale for TOE security functions 60
8.3.2 Rationale for assurance measures 61
8.4 PP Claims Rationale 61
9. Annexes 62
9.1 Further Information contained in the PP 62
9.2 Glossary and Vocabulary 62
9.3 List of Abbreviations 66
9.4 Bibliography 67
9.4.1 Evaluation Documents 67
9.4.2 Developer Documents 67
9.4.3 Other Documents 68
10. Legal information 69
10.1 Definitions 69
10.2 Disclaimers 69
10.3 Licenses 69
10.4 Trademarks 69
11. Contents 70

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

Date of release: 30 July 2012