NXP Secure Smart Card Controllers P5CD016/021/041/051V1A and P5Cx081V1A

Security Target Lite

Rev. 1.4 — 25 October 2010 **Evaluation documentation**

PUBLIC

BSI-DSZ-CC-0555

Document information

Info	Content
Keywords	Evaluation Documentation, CC, Security Target Lite, P5CD016/021/041/051V1A and P5Cx081V1A, P5CD081V1A, P5CD051V1A. P5CD041V1A. P5CD021V1A. P5CD016V1A, P5CC081V1A, P5CN081V1A, Secure Smart Card Controller, BSI-DSZ-CC-0555
Abstract	The document at hand is the Security Target of NXP Secure Smart Card Controllers P5CD016/021/041/051V1A and P5Cx081V1A, which are developed and provided by NXP Semiconductors, Business Line Identification according to the Common Criteria for Information Technology Security Evaluation Version 3.1 at Evaluation Assurance Level 5 augmented.

Revision history

Latest revision: Rev. 1.4, 25 October 2010

Rev	Date	Description
1.4	25-Oct-2010	Add configuration P5CD051V1A, generalized Wafer form Ui in Table 4, generalized module form Xn in Table 4, US-eng., formatting
1.3	21-Sep-2009	corrections in sections 1.4.1.2 and 1.4.1.3
1.2	7-Sep-2009	Derived from full ST V1.2

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 reference”, “TOE reference”, “TOE overview” and “TOE Description”.

1.1 ST reference

“NXP Secure Smart Card Controllers P5CD016/021/041/051V1A and P5Cx081V1A Security Target Lite, Rev. 1.4, NXP Semiconductors, 25 October 2010”.

1.2 TOE reference

NXP Secure Smart Card Controllers P5CD016/021/041/051V1A and P5Cx081V1A

1.3 TOE overview

1.3.1 Usage and major security functionality of the TOE

The TOE is the IC hardware of the microcontroller chip family of NXP Secure Smart Card Controllers P5CD016/021/041/051V1A and P5Cx081V1A. The TOE also comprises 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 incorporates an 8-bit processing unit, volatile and non-volatile memories accessible via a Memory Management Unit, cryptographic coprocessors, other security components and two communication interfaces.

The TOE also includes a Data Sheet, a document describing the Instruction Set and the Guidance Document. This documentation provides a description of the architecture and the secure configuration and usage of the IC hardware by the Security IC Embedded Software.

The security functionality of the TOE is designed to act as an integral part of a complete security system in order to strengthen the design as a whole. Several security mechanisms are completely implemented in and controlled by the TOE. Other security mechanisms allow for configuration or even require handling of exceptions by the Security IC Embedded Software. With different CPU modes and a Memory Management Unit the TOE is intended to support multi-application projects.

On-chip memories are ROM, EEPROM and RAM. The non-volatile EEPROM can be used as data or program memory. It contains high reliability cells, which guarantee data integrity. This is perfect for applications requiring non-volatile data storage and important for the use as memory for native programs. Security functionality protects the contents of all memories.

A Security IC must provide high security in particular when being used in the banking and finance market or in electronic commerce applications. Hence the TOE shall

- maintain the integrity and the confidentiality of code and data stored in its memories 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 functionality provided by the TOE.

This is ensured by the construction of the TOE and its security functionality.

P5CD016/021/041/051V1A and P5Cx081V1A basically provide a hardware platform for a smartcard with

- functionality to calculate the Data Encryption Standard (Triple-DES) with up to three keys,
- functionality to calculate the Advanced Encryption Standard (AES) with different key lengths,
- support for large integer arithmetic operations like multiplication, addition and logical operations, which is suitable for public key cryptography and elliptic curve cryptography,
- a Random Number Generator,
- memory management control,
- cyclic redundancy check (CRC) calculation,
- ISO/IEC 7816 contact interface with UART,
- contactless interface supporting MIFARE and ISO/IEC 14443 A.

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

Note: Large integer arithmetic operations are intended to be used for calculation of asymmetric cryptographic algorithms. Any asymmetric cryptographic algorithm utilizing the support for large integer arithmetic operations has to be implemented in Security IC Embedded Software. Thus, the support for large integer arithmetic operations itself does not provide security functionality like cryptographic support. The Security IC Embedded Software implementing an asymmetric cryptographic algorithm is not included in this evaluation. Nevertheless the support for large integer arithmetic operations is part of the Security 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 of evaluation is applied to the CRC calculation.

1.3.2 TOE type

The TOE is the family of NXP Secure Smart Card Controllers P5CD016/021/041/051V1A and P5Cx081V1A. The TOE includes the IC hardware, IC Designer/Manufacturer proprietary IC Dedicated Test Software and IC Dedicated Support Software.

The TOE is delivered in a module, inlay or package, or as a sawn wafer.

1.3.3 Required non-TOE hardware/software/firmware

None

1.4 TOE Description

1.4.1 Physical Scope of TOE

P5CD016/021/041/051V1A and P5Cx081V1A is manufactured in an advanced CMOS process. A block diagram is depicted in Fig 1.

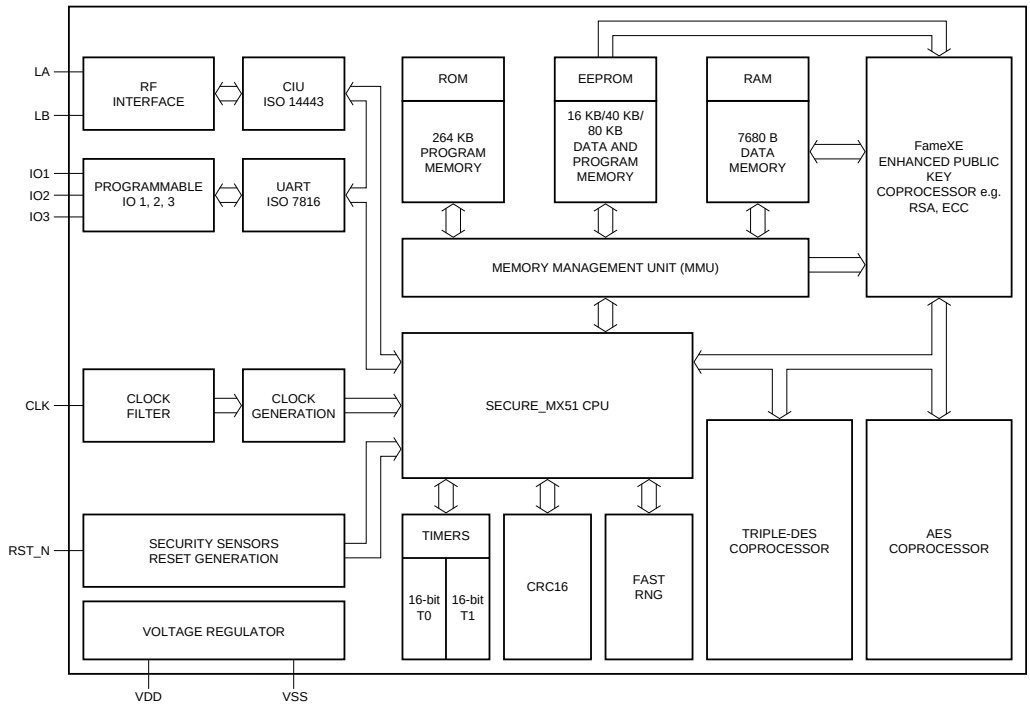

Fig 1. Block Diagram of P5CD016/021/041/051V1A and P5Cx081V1A

The TOE consists of the IC hardware and the IC Dedicated Software as composed of IC Dedicated Test Software and IC Dedicated Support Software. All other software is called Security IC Embedded Software and is not part of the TOE. The following table lists the TOE components.

TOE components

Table 1. Components of the TOE

Type	Name	Release	Date	Form of delivery
IC Hardware	NXP Secure Smart Card Controllers P5CD016/021/041/051V1A and P5Cx081V1A	V1A	T046B_20090210 .gds2	Wafer, modules and packages (dice include reference T046A)
IC Dedicated Test Software	Test-ROM Software	87	16 January 2009	Test-ROM on the chip, tmfos_87.lst

Type	Name	Release	Date	Form of delivery
IC Dedicated Support Software	Boot-ROM Software	87	16 January 2009	Test-ROM on the chip, tmfos_87.lst
	MIFARE Operating System	87	16 January 2009	Test-ROM on the chip, tmfos_87.lst
Document	Data Sheet P5CD016/021/041/051 and P5Cx081 family, Secure dual interface and contact PKI smart card controller			Electronic document
Document	Instruction Set, SmartMX-Family, Secure and PKI Smart Card Controller	1.1	4 July 2006	Electronic document
Document	Guidance, Delivery and Operation Manual NXP Secure Smartcard Controllers P5CD016/021/041/051 and P5Cx081			Electronic document

1.4.1.1 Evaluated hardware configurations

Seven major configuration options are present, which are denoted by product names P5CD016V1A, P5CD021V1A, P5CD041V1A, P5CD051V1A, P5CD081V1A, P5CC081V1A and P5CN081V1A. All of them are equipped with an EEPROM of 80 kBytes and both, the ISO/IEC 7816 contact interface and the ISO/IEC 14443 contactless interface. Their major differences are related to availability of EEPROM space and the contactless interface as detailed below.

The minor configuration options of all major configurations are described in subsection 1.4.1.2.

Major configuration P5CD016V1A

P5CD016V1A supports all minor configuration options presented in subsection 1.4.1.2. Its major differences to other configurations are as follows.

- The EEPROM space available to the Security IC Embedded Software is reduced to 16 kBytes minus 256 Bytes, which are reserved for Security Rows (128 Bytes) and configuration data (128 Bytes).
- The contactless interface is enabled and configured for contactless communication according to ISO/IEC 14443 A in [23] and [24].

Major configuration P5CD021V1A

P5CD021V1A supports all minor configuration options presented in subsection 1.4.1.2. Its major differences to other configurations are as follows.

- The EEPROM space available to the Security IC Embedded Software is reduced to 20 kBytes minus 256 Bytes, which are reserved for Security Rows (128 Bytes) and configuration data (128 Bytes).
- The contactless interface is enabled and configured for contactless communication according to ISO/IEC 14443 A in [23] and [24].

Major configuration P5CD041V1A

P5CD041V1A supports all minor configuration options presented in subsection 1.4.1.2. Its major differences to other configurations are as follows.

- The EEPROM space available to the Security IC Embedded Software is reduced to 40 kBytes minus 256 Bytes, which are reserved for Security Rows (128 Bytes) and configuration data (128 Bytes).
- The contactless interface is enabled and configured for contactless communication according to ISO/IEC 14443 A in [23] and [24].

Major configuration P5CD051V1A

P5CD051V1A supports all minor configuration options presented in subsection 1.4.1.2. Its major differences to other configurations are as follows.

- The EEPROM space available to the Security IC Embedded Software is reduced to 52 kBytes minus 256 Bytes, which are reserved for Security Rows (128 Bytes) and configuration data (128 Bytes).
- The contactless interface is enabled and configured for contactless communication according to ISO/IEC 14443 A in [23] and [24].

Major configuration P5CD081V1A

P5CD081V1A supports all minor configuration options presented in subsection 1.4.1.2. Its major differences to other configurations are as follows.

- The physically implemented EEPROM of 80 kBytes is available to the Security IC Embedded Software except for 256 Bytes, which are reserved for Security Rows (128 Bytes) and configuration data of the manufacturer (128 Bytes).
- The contactless interface is enabled and configured for contactless communication according to ISO14443 A in [23] and [24].

Major configuration P5CC081V1A

P5CC081V1A supports all minor configuration options presented in subsection 1.4.1.2. Its major differences to other configurations are as follows.

- The physically implemented EEPROM of 80 kBytes is available to the Security IC Embedded Software except for 256 Bytes, which are reserved for Security Rows (128 Bytes) and configuration data (128 Bytes).
- The contactless interface is disabled.

Major configuration P5CN081V1A

P5CN081V1A supports all minor configuration options presented in subsection 1.4.1.2. Its major differences to other configurations are as follows.

- The physically implemented EEPROM of 80 kBytes is available to the Security IC Embedded Software except for 256 Bytes, which are reserved for Security Rows (128 Bytes) and configuration data (128 Bytes).
- The contactless interface is enabled and configured in S^2C mode. The TOE now can be used for Near Field Communication (NFC) [25], for which an additional NFC helper IC is connected to IO3/SIGIN and LB/SIGOUT. However, it is possible that the TOE is powered in an appropriate electrical field when an antenna is connected to LA and LB. Pad I/O3 is connected to the NFC helper IC and can not be used for contact communication according to ISO 7816.

1.4.1.2 Common minor configuration options

The following minor configuration options can be selected by the customer via order entry forms.

Table 2. Evaluated minor configuration options

Name	Values	Description
EDATASCALE	10h up to FFh	This value determines the size of the memory area available for the extended stack pointer. Refer to section 10.5 of [8].
Card Disable Function	Yes or No	When the Card Disable Function is enabled, the TOE can be locked completely. Once set by the Security IC Embedded Software, the execution of the Security IC Embedded Software is inhibited after the next reset. Refer to section 29.4 of [8].
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 [8].
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 [8].
128 Byte Page Mode	Yes or No	In the 128 Byte Page Mode up to 128 Bytes of EEPROM can be programmed simultaneously, instead of up to 64 Bytes. Refer to section 10.9.1 of [8].
MIFARE Emulation	A, B1 or B4	Different MIFARE configurations refer to chapter 21 of [8].
UID in MIFARE Emulation A	Single or Double	The size of the UID can be 4 bytes (Single UID) or 7 bytes (Double UID). Refer to section 11.1.1 of [8].
Contactless communication protocol	(i) “proprietary protocol (compliant to ISO 14443 part 3)”; (ii) “T=CL protocol (compliant to ISO 14443 part 3 and part 4)”	Refer to section 21 of [8].
Maximum CIU Baudrate	106, 212, 424 or 848 kBaud	Defines the maximum available baudrate of the contactless interface. Refer to section 21 of [8].
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 34.1.3, 35.2, 29.2.2, 5 and 33 of [8].

Name	Values	Description
Voltage Class C operation activated	Yes or No	If Voltage Class C is activated, supply voltage range is: 1.62 V ≤ V_{DD} ≤ 1.98 V. Refer to sections 34.1.3, 35.2, 29.2.2, 5 and 33 of [8].
Requested LA/LB input capacitance	17 pF or 69 pF	Additional capacitance (2x26 pF) between LA/LB required to meet resonance frequency at ID1/2 operation.
Simultaneous Operation of ISO/IEC 7816 and ISO/IEC 14443 applications	Yes or No	Disables the Low Frequency Sensor to allow parallel operation via contact and contactless interfaces. The Low Frequency Sensor is disabled only when the CPU is free-running or runs at an internal clock.

The values of all options listed in Table 2 can be chosen independently.

The Order Entry Forms [11], [12], [13], [14], [15], [16] and [17] contain a further option, which must be selected with a fixed value:

The option “Allow execution from RAM” must be selected with “No”.

1.4.1.3 Evaluated package types

A number of package types is supported for each major configuration of the TOE. The commercial types are named according to the following format.

P5CD016*pp*/T1A*rrffz* for major configuration P5CD016V1A

P5CD021*pp*/T1A*rrffz* for major configuration P5CD021V1A

P5CD041*pp*/T1A*rrffz* for major configuration P5CD041V1A

P5CD051*pp*/T1A*rrffz* for major configuration P5CD051V1A

P5CD081*pp*/T1A*rrffz* for major configuration P5CD081V1A

P5CC081*pp*/T1A*rrffz* for major configuration P5CC081V1A

P5CN081*pp*/T1A*rrffz* for major configuration P5CN081V1A

The commercial type name of each major configuration varies with the package type as indicated by the variable *pp*, - and with the Security IC Embedded Software as indicated by the variables *rr*, *ff* and *z*. The variables are replaced according to the rules in Table 3.

Table 3. 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 with electronically marked defects. The different types are defined in Table 4.
rr	ROM code number, different for every Security IC Embedded Software
ff	FabKey number, multiple keys are supported for each Security IC Embedded Software
z	MIFARE Configuration (0=A, 1=B1, 4=B4)

Table 4 depicts the package types, which are supported in this Security Target, and assigns these to the major configuration types. The two characters in each entry of the table stand for *pp*, and identify the package type. An empty cell means that the Security

Target does not support the respective package type for the corresponding major configuration.

Table 4. Supported commercial types

P5CD016V1A	P5CD021V1A	P5CD041V1A	P5CD051V1A	P5CD081V1A	P5CC081V1A	P5CN081V1A	
Ux	Ux	Ux	Ux	Ux	Ux	Ux	Wafer not thinner than 75µm (The letter “x” in “Ux” stands for a capital letter or a number, which identifies the wafer type)
Xn	Xn	Xn	Xn	Xn	Xn	Xn	Module (The letter “n” in” Xn” stands for a capital letter or a number, which identifies the module type)
A4		A4	A4	A4			MOB4 module
A6		A6	A6	A6			MOB6 module
		Ai		Ai			Inlay (The letter ‘i” in “Ai” stands for a capital letter, which identifies both, the inlay type and the package type inside the inlay.)

For example, the commercial type name “P5CD081A4/T1Arrffz” denotes a P5CD081V1A in a MOB4 module and “P5CD081UA/T1Arrffz” denotes a P5CC081V1A on a 150µm sawn wafer inkless, which means that the defect ICs are electronically marked.

The package types do not influence the security functionality of the TOE. They only define which pads are connected in the package and for what purpose the chip 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 on his own.

Security during development and production is ensured for all package types listed above, please refer to section 1.4.3.

As already described above the complete resulting commercial type name is dependent on the customer software, i.e. the Security IC Embedded Software. Thus, a full commercial product name, which fits in the variable forms described in Table 4, determines an evaluated hardware, but gives no conclusion on the Security IC Embedded Software and whether it uses the proper hardware configuration as detailed in subsection 1.4.1.2.

1.4.2 Logical Scope of TOE

1.4.2.1 Hardware Description

The CPU of P5CD016/021/041/051V1A and P5Cx081V1A 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. P5CD016/021/041/051V1A and P5Cx081V1A distinguishes five CPU modes, which are summarized in the following table.

Table 5. CPU modes of the TOE

Super System Mode				
Boot Mode	Test Mode	MIFARE Mode	System Mode	User Mode

Boot Mode, Test Mode and MIFARE Mode are sub-modes of the so-called Super System Mode. These three modes are not available to the Security IC Embedded Software, they are reserved for the IC Dedicated Software. The IC Dedicated Software is composed of the Boot-ROM Software, the Test-ROM Software and the MIFARE Operating System as introduced in section 1.4.1. The three software components are mapped one-to-one to the three modes: In Boot Mode the TOE executes the Boot-ROM Software, in Test Mode the TOE executes the Test-ROM Software and in MIFARE Mode the TOE executes the MIFARE Operating System. Please 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, which is not available to the Security IC Embedded Software.

P5CD016/021/041/051V1A and P5Cx081V1A is able to control two different logical phases. After production of the IC hardware every start-up or reset completes with Test Mode and execution of the Test-ROM Software. The Test Mode is disabled at the end of the production test. Afterwards, every start-up or reset ends up in System Mode and execution of the Security IC Embedded Software.

System Mode and User Mode are available to the developer of the Security IC Embedded Software. The System Mode provides unlimited access to the hardware components. In User Mode the access is restricted to the CPU and specific Special Function Registers. Access rights to hardware components for software running in User Mode can be granted by software running in System Mode. The hardware components are controlled by the Security IC Embedded Software via Special Function Registers. Special Function Registers are interrelated to the activities of the CPU, the Memory Management Unit, interrupt control, I/O configuration, EEPROM, timers, UART, the contactless interface and the coprocessors.

Communication with P5CD016/021/041/051V1A and P5Cx081V1A can be established via the contact interface through UART or direct usage of the I/O ports. Contactless communication is done via the contactless interface unit (CIU) compatible to MIFARE and ISO/IEC 14443. P5CD016/021/041/051V1A and P5Cx081V1A provides two types of interrupts: (i) exception interrupts, called “exception” in the following and (ii) event interrupts, called “interrupts” in the following. Exceptions and interrupts each force a jump to a specific fixed vector address in the ROM. Any exception and interrupt can therefore be controlled and guided by a specific part of the Security IC Embedded Software. In conjunction with the jump to a specific fixed vector address the IC hardware always enables a pre-defined CPU mode, which is either System Mode or User Mode. In addition, the P5CD016/021/041/051V1A and P5Cx081V1A provides eight configuration vectors (CVEC) and 32 system call vectors (SVEC). These vectors have to be explicitly called by the Security IC Embedded Software. A jump to a configuration vector forces MIFARE Mode, a jump to a system call vector forces System Mode.

Special hardware protects and separates the CPU modes from each other in particular consideration of Boot Mode and Test Mode.

P5CD016/021/041/051V1A and P5Cx081V1A incorporates 288 kBytes of ROM, 7680 Bytes of RAM and 80 kBytes of EEPROM. Access control to all three memory types is enforced by a Memory Management Unit. The Memory Management Unit

partitions each memory into two parts: The ROM is partitioned in 264 kBytes Application-ROM and 24 kBytes Test-ROM. The EEPROM is partitioned depending on the configuration, 128 Bytes are always reserved for the manufacturer and either zero, one or four kBytes are additionally reserved for the MIFARE Operating System according to the MIFARE configurations A, B1 or B4. The RAM is also partitioned depending on the configuration, either 0 Byte for MIFARE configuration A and the whole 7680 Bytes for the application, or 128 Bytes for MIFARE configurations B1 or B4 and the remaining 7552 Bytes for the application. Note that the ROM size is displayed as 264 kBytes in the block diagram in Fig 1 because only 264 kBytes are available to the Security IC Embedded Software.

In Test Mode the CPU has unrestricted access to the all memories. In Boot Mode and MIFARE Mode access is limited to the Test-ROM, the 128 kBytes EEPROM plus its configured part of zero, one or four kBytes and the configured part of 0 or 128 kBytes RAM. All other parts of the memories are accessible in System Mode and User Mode, namely the Application-ROM and the larger parts of EEPROM and RAM. User Mode is further restricted by the Memory Management Unit, which can be configured in System Mode.

The RAM, which is available to the Security IC Embedded Software, is further split in two parts. These are 5120 or 4992 Bytes general purpose RAM depending on the MIFARE configuration and 2560 Bytes FameXE RAM. Both parts are accessible to the CPU, but the FameXE coprocessor can only access the FameXE RAM. The FameXE can access the FameXE RAM without control of access rights by the Memory Management Unit. Since the Memory Management Unit does not control accesses of the FameXE, software which has access to the FameXE implicitly has access to the FameXE RAM. This also holds for the EEPROM, which is available to the Security IC Embedded Software. FameXE accesses to this part of the EEPROM are not controlled by the Memory Management Unit, i.e. software, which has access to the FameXE, implicitly has access to this part of the EEPROM.

The Triple-DES coprocessor supports single DES and Triple-DES operations. Only Triple-DES is used in this evaluation, in 2-key or 3-key operation. The AES coprocessor supports AES operation with three different key lengths. The FameXE coprocessor supplies basic arithmetic functions to support implementation of asymmetric cryptographic algorithms by the Security IC Embedded Software. The random generator provides true random numbers without pseudo random calculation.

P5CD016/021/041/051V1A and P5Cx081V1A operates with a single power supply of 1.8 V, 3 V or 5 V nominal. The maximum external clock frequency is 10 MHz nominal. P5CD016/021/041/051V1A and P5Cx081V1A can be operated as well with an internal clock. This decreases the calculation time of security algorithms. P5CD016/021/041/051V1A and P5Cx081V1A provides power saving modes with reduced activity. These are named IDLE Mode and SLEEP Mode, of which the latter one includes CLOCK STOP Mode.

The TOE protects secret data, which are stored to and operated by the TOE, against physical tampering. The security functionality of a Security IC is partially provided by the TOE, and completed by the Security IC Embedded Software. This causes dependencies between the security functionality of the TOE and the security functionality provided by the Security IC Embedded Software.

1.4.2.2 Software Description

Operating system and applications of a Security IC are developed by the customers and included under the heading Security IC Embedded Software. The Security IC Embedded Software is stored in the Application-ROM and/or in the EEPROM and is not part of the TOE. The Security IC Embedded Software depends on the usage of the Security IC.

The IC Dedicated Test Software, which is named Test-ROM Software, is stored to the Test-ROM and used by the manufacturer of the Security IC during production test. The test functionality is disabled before operational use of the Security IC by disabling the Test Mode of the CPU in hardware. The IC Dedicated Test Software is developed by NXP and embedded in the Test-ROM. The Test-ROM 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 Rows and shutdown functions to ensure that security relevant test operations can not be executed illegally after phase 3.

The Dedicated Support Software is also stored to the Test-ROM and consists of two parts.

- The Boot-ROM Software, which is executed during start-up or reset of the TOE, i.e. each time when the TOE powers up or resets. It sets up the TOE and its basic configuration.
- The MIFARE Operating System, which provides MIFARE functionality to the Security IC Embedded Software. The MIFARE Operating System provides support for contactless communication. Please refer to [26] for more information.

1.4.2.3 Documentation

The data sheet "Data Sheet P5CD016/021/041/051 and P5Cx081 family, Secure dual interface and contact PKI smart card controller" [8] contains a functional description and guidelines for the use of the security functionality, as needed to develop Security IC Embedded Software. The instruction set of the CPU is described in "Instruction Set, SmartMX-Family, Secure and PKI Smart Card Controller" [9]. The manual "Guidance, Delivery and Operation Manual NXP Secure Smartcard Controllers P5CD016/021/041/051 and P5Cx081" [10] describes aspects of the program interface and the use of programming techniques to improve the security. The whole documentation shall be used by the developer to develop the Security IC Embedded Software.

1.4.3 Security during Development and Production

The Security IC product life-cycle is scheduled in phases as introduced in the PP [6]. IC Development as well as IC Manufacturing and Testing, which are phases 2 and 3 of the life-cycle, are part of the evaluation. The Security IC is delivered at the end of phase 3 or after IC Packaging, which is phase 4 in the life-cycle, and then part of the evaluation as well. The development and production environment of the TOE ranges from phase 2 to TOE Delivery.

With respect to Application Note 3 in [6] the TOE supports the authentic delivery using the FabKey feature. For details on this feature please refer to the data sheet in [8] and the manual in [10].

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 Line 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. Accountability and traceability are 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 control of the complete process including 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 modules, inlays or 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 four different forms, which are

- dice on wafers
- smartcard modules on a module reel
- inlays
- packaged devices in tubes or reels

The availability of major configuration options of the TOE in package types is detailed in section 1.4.1.3.

1.4.4 TOE Intended Usage

The end-consumer environment of the TOE is phase 7 of the Security IC product life-cycle in the PP [6]. In this phase the Security IC product is in usage by the end-consumer. Its method of use now depends on the application. Security ICs are used to assure authorised conditional access in a wide range of applications. Examples are identity cards, Banking Cards, Pay-TV, Portable communication SIM cards, Health cards, Transportation cards. The end-user environment covers a wide spectrum of very different functions, thus making it difficult to monitor and avoid abuse of the TOE. The TOE is intended to be used in an insecure environment, which does not protect against threats.

The device is developed for most high-end safeguarded applications, and is designed for embedding into chip cards according to ISO/IEC 7816 [21] and for contactless applications. Usually a Security IC (e.g. a smartcard) is assigned to a single individual only, but it may also be used by multiple applications in a multi-provider environment. Therefore the TOE might store and process secrets of several systems, which must be protected from each other. The TOE then must meet security requirements for each single security module. Secret data shall be used as input for calculation of authentication data, calculation of signatures and encryption of data and keys.

In development and production environment of the TOE the Security IC Embedded Software developer and system integrators such as the terminal software developer may

use samples of the TOE for their testing purposes. It is not intended that they are able to change the behaviour of the Security IC in another way than an end-consumer.

The user environment of the TOE ranges from TOE delivery to phase 7 of the Security IC product life-cycle, and must be a controlled environment up to phase 6.

Note: The phases from TOE Delivery to phase 7 of the Security IC Product life-cycle are not part of the TOE construction process in the sense of this Security Target. Information about these phases is just included to describe how the TOE is used after its construction. Nevertheless such security functionality of the TOE, that is independent of the Security IC Embedded Software, is active at TOE Delivery and can not be disabled by the Security IC Embedded Software in the following phases.

1.4.5 Interface of the TOE

The electrical interface of the TOE are the pads to connect the lines power supply, reset input, clock input, ground, serial communication pads I/O1, I/O2 and I/O3 as well as two pads (called LA and LB) for the antenna of the contactless interface unit.

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 comprises the test operating system and the package of test function calls stored in the Test-ROM.

In the MIFARE Mode the MIFARE Operating System is executed by the CPU – only on request by the Security IC Embedded Software.

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 Security IC Embedded Software.

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

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 behaviour the data interface is defined by the Security IC Embedded Software.

2. Conformance Claims

This chapter is divided into the following sections: “CC Conformance Claim", "Package claim", "PP claim", and "Conformance Claim Rationale”.

2.1 CC Conformance Claim

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

- “Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, Version 3.1, Revision 1, September 2006, CCMB-2006-09-001”, [1]
- “Common Criteria for Information Technology Security Evaluation – Part 2: Security functional requirements, Version 3.1, Revision 2, September 2007, CCMB-2007-09-002”, [2]
- “Common Criteria for Information Technology Security Evaluation – Part 3: Security Assurance Requirements, Version 3.1, Revision 2, September 2007, CCMB-2007-09-003”, [3]

The following methodology will be used for the evaluation.

- “Common Methodology for Information Technology Security Evaluation CEM-99/045 Part 2: Evaluation Methodology, Version 3.1, Revision 2, September 2007, CCMB-2007-09-004”, [4]

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

2.2 Package claim

This Security Target claims conformance to the assurance package EAL 5 augmented. The augmentations to EAL5 are ALC_DVS.2 and AVA_VAN.5. In addition, the Security Target is augmented using the component ASE_TSS.2, which is chosen to include architectural information on the security functionality of the TOE.

Note: The PP “Security IC Platform Protection Profile” [6] to which this Security Target claims conformance (refer to section 2.3) requires assurance level EAL4 augmented. The changes, which are needed for EAL5, are described in the relevant sections of this Security Target.

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.

2.3 PP claim

This Security Target claims conformance to the Protection Profile (PP)

“Security IC Platform Protection Profile, Version 1.0, registered and certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-PP-0035” ,[6].

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

The TOE provides additional functionality, which is not covered in the PP [6]. In accordance with Application Note 4 of [6] this additional functionality is added using the policy “P.Add-Components” (see section 3.3 of this Security Target).

2.4 Conformance Claim Rationale

According to section 2.3 this Security Target claims conformance to the PP “Security IC Platform Protection Profile” [6].

The TOE type defined in section 1.3.2 of this Security Target is a smartcard controller. This is consistent with the TOE definition for a Security IC in section 1.2.2 of [6].

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

The evaluation assurance level claimed for this target (EAL5+) is shown in section 6.2 to include respectively exceed the requirements claimed by the PP (EAL4+).

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

3. Security Problem Definition

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

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

3.1 Description of Assets

Since this Security Target claims conformance to the PP “Security IC Platform Protection Profile” [6] the assets defined in section 3.1 of [6] are applied here. These assets are cited here.

The assets related to standard functionality are:

- the User Data,
- the Security IC Embedded Software, stored and in operation,
- the security services provided by the TOE for the Security IC Embedded Software.

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

logical design data, physical design data, IC Dedicated Software, configuration data, Initialization Data and Pre-personalization Data, specific development aids, test and characterization related data, material for software development support, photomasks.

Note that the keys for the cryptographic coprocessors are seen as User Data.

3.2 Threats

Since this Security Target claims conformance to the PP “Security IC Platform Protection Profile” [6] the threats defined in section 3.2 of [6] are valid for this Security Target. The following table lists the threats defined in the PP [6].

Table 6. Threats defined by the PP [6]

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

Considering Application Note 5 in [6] there are no additional threats defined in this Security Target.

3.3 Organizational Security Policies

Since this Security Target claims conformance to the PP “Security IC Platform Protection Profile” [6] the policy P.Process-TOE “Protection during TOE Development and Production” in [6] is applied here as well.

In accordance with Application Note 6 in [6] there is one additional policy defined in this Security Target as detailed below.

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

The IC Developer/Manufacturer therefore applies 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 Security IC Embedded Software: Triple-DES encryption and decryption AES encryption and decryption Area based Memory Access Control Memory separation for different software parts (including IC Dedicated Software and Security IC Embedded Software) Special Function Register Access Control.

3.4 Assumptions

Since this Security Target claims conformance to the PP “Security IC Platform Protection Profile” [6] the assumptions defined in section 3.4 of [6] are valid for this Security Target. The following table lists these assumptions.

Table 7. Assumptions defined in the PP [6]

Name	Title
A.Process-Sec-IC	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 added in this Security Target according to Application Notes 7 and 8 in [6].

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

The Security IC Embedded Software must provide a function to check initialization data. 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 additional assumption considers specialized encryption hardware of the TOE.

The developer of the Security IC 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 Security IC Embedded Software in a way that they are not susceptible to leakage attacks (as described under T.Leak-Inherent and T.Leak-Forced).

Note that here the routines which may compromise keys when being executed are part of the Security IC Embedded Software. In contrast to this 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.

4. Security Objectives

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

4.1 Security Objectives for the TOE

The TOE shall provide the following security objectives, which are taken from the PP “Security IC Platform Protection Profile” [6].

Table 8. Security objectives defined in the PP [6]

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

Regarding the Application Notes 9 and 10 in [6] 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 Security IC 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.HW_AES — AES Functionality

The TOE shall provide the cryptographic functionality to calculate an AES encryption and decryption to the Security IC Embedded Software. The TOE supports directly the calculation of AES with three different key lengths.

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

O.MF_FW	MIFARE Firewall The TOE shall provide separation between the “MIFARE Operating System” as part of the IC Dedicated Support Software and the Security IC 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 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.

4.2 Security Objectives for the Security IC Embedded Software development Environment

In addition to the security objectives for the operational environment as required by CC Part 1 [1] the PP “Security IC Platform Protection Profile” [6] defines security objectives for the Security IC Embedded Software development environment which are listed below.

Table 9. Security objectives for the Security IC Embedded Software development environment, taken from the PP [6]

Security objective	Description	Applies to phase...
OE.Plat-Appl	Usage of Hardware Platform	Phase 1
OE.Resp-Appl	Treatment of User Data	Phase 1

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

The TOE supports cipher schemes as additional specific security functionality. If required the Security IC Embedded Software shall use these cryptographic services of the TOE and their interface as specified. When key-dependent functions implemented in the Security IC Embedded Software are just being executed, the Security IC 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 Security IC Embedded Software (Operating System) can implement a memory management scheme based upon security functionality 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 Security IC 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 Security IC 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.

4.3 Security Objectives for the Operational Environment

The following security objectives for the operational environment are specified according to the PP "Security IC Platform Protection Profile" [6].

Table 10. Security objectives for the operational environment, taken from the PP [6]

Security objective	Description	Applies to phase...
OE.Process-Sec-IC	Protection during composite product manufacturing	TOE delivery up to the end of phase 6

Check of initialization data

The TOE provides specific functionality 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 Security IC Embedded Software

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

4.4 Security Objectives Rationale

Section 4.4 in the PP “Security IC Platform Protection Profile” [6] provides a rationale how the assumptions, threats, and organizational security policies are addressed by the objectives that are specified in the PP [6]. The following Table 11 reproduces the table in section 4.4 of [6].

Table 11. Security Objectives versus Assumptions, Threats or Policies

Assumption, Threat or OSP	Security Objective	Notes
A.Plat-Appl	OE.Plat-Appl	Phase 1
A.Resp-Appl	OE.Resp-Appl	Phase 1
P.Process-TOE	O.Identification	Phase 2 – 3
A.Process-Sec—IC	OE.Process-Sec-IC	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 12 provides the justification for the additional security objectives. They are in line with the security objectives of the PP [6] and supplement these according to the additional assumptions and organizational security policy.

Table 12. Additional Security Objectives versus Assumptions or Policies

Assumption/Policy	Security Objective	Note
P.Add-Components	O.HW_DES3 O.HW_AES O.MF_FW O.MEM_ACCESS O.SFR_ACCESS	
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 detailed below.

The justification related to the security objectives O.HW_DES3, O.HW_AES, O.MF_FW, O.MEM_ACCESS and O.SFR_ACCESS is as follows. Since these objectives require 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 functionality can not be influenced or used in User Mode.

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

- Compared to [6] a clarification has been made for the security objective “Usage of Hardware Platform (OE.Plat-Appl)”: If required the Security IC Embedded Software shall use the cryptographic service of the TOE and its interface as specified. In addition, the Security IC 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 Security IC Embedded Software uses random numbers provided by the security service SS.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 [6] 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 Security IC Embedded Software will protect such data if required and use keys and functions appropriately in order to ensure the strength of cryptographic operation. Quality and confidentiality must be maintained for keys that are imported and/or derived from other keys. This implies that appropriate key management has to be implemented in the environment. In addition, the treatment of User Data 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 Security IC Embedded Software (A.Check-Init)" is as follows:

Since OE.Check-Init requires the Security IC 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 PP [6] for the assumptions, policy and threats defined there.

5. Extended Components Definition

This Security Target does not define extended components.

Note that the PP “Security IC Platform Protection Profile” [6] defines extended security functional requirements in chapter 5, which are included in this Security Target.

6. Security Requirements

This chapter consists of the sections “Security Functional Requirements”, “Security Assurance Requirements” and “Security Requirements Rationale”.

6.1 Security Functional Requirements

The Security Functional Requirements (SFRs) of the TOE are presented in the following sections to support a better understanding of the combination of PP “Security IC Platform Protection Profile” [6] and Security Target.

6.1.1 SFRs of the Protection Profile

Table 13 below shows all SFRs, which are specified in the PP [6] (in the order of definition in the PP). Some of the SFRs are CC Part 2 extended and defined in the PP [6]. This is shown in the third column of the table.

Table 13. SFRs taken from the PP [6]

SFR	Title	Defined in
FRU_FLT.2	Limited fault tolerance	CC, Part 2
FPT_FLS.1	Failure with preservation of secure state	CC, Part 2
FMT_LIM.1	Limited capabilities	PP, Section 5.2
FMT_LIM.2	Limited availability	PP, Section 5.2
FAU_SAS.1	Audit storage	PP, Section 5.3
FPT_PHP.3	Resistance to physical attack	CC, Part 2
FDP_ITT.1	Basic internal transfer protection	CC, Part 2
FPT_ITT.1	Basic internal TSF data transfer protection	CC, Part 2
FDP_IFC.1	Subset information flow control	CC, Part 2
FCS_RNG.1	Random number generation	PP, Section 5.1

All operations except for the following assignments and selections are already performed in the PP [6].

For the SFR FAU_SAS.1 the PP [6] leaves the assignment operation open for the non-volatile memory type in which initialization data, pre-personalization data and/or other supplements for the Security IC Embedded Software are stored. This assignment operation is filled in by the following statement. Note that the assignment operations for the list of subjects and the list of audit information have already been filled in by the PP [6].

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

[1] [assignment: *list of subjects*]

personalization Data and/or supplements of the Security IC Embedded Software [2] in the *EEPROM* [3].

For FCS_RNG.1.1 the PP [6] partially fills in the assignment for the security capabilities of the RNG by requiring a total failure test of the random source and adds an assignment operation for additional security capabilities of the RNG.

In addition, for FCS_RNG.1.2 the PP [6] partially fills in the assignment operation for the defined quality metric for the random numbers by replacing it by a selection and assignment operation.

For the above operations the original operations defined in chapter 5 of the PP [6] have been replaced by the open operations of the partially filled in operations in the statement of the security requirements in chapter 6 of [6] for better readability. Note that the selection operation for the RNG type has already been filled in by the PP [6].

FCS_RNG.1	Random number generation
Hierarchical to:	No other components.
FCS_RNG.1.1	The TSF shall provide a *physical* [4] Random Number Generator that implements *total failure test of the random source and none* [5].
FCS_RNG.1.2	The TSF shall provide random numbers that meet *independent bits with Shannon entropy of 7.976 bits per octet*[6].
Dependencies:	No dependencies.
Note:	Application Note 20 in [6] requires that the Security Target specifies for the security capabilities in FCS_RNG.1.1 how the results of the total failure test of the random source are provided to the Security IC Embedded Software. The TOE features a hardware test which is called by the Security IC Embedded Software. The results of the internal test sequence are provided to the Security IC Embedded Software as a pass or fail criterion by means of a special function register. The entropy of the random number is measured by the Shannon-Entropy as follows: $E = -\sum_{i=0}^{255} p_i \cdot \log_2 p_i$, where p_i is the probability that the byte $(b_7, b_6, \ldots, b_0)$ 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 PP [6].

---

2 [assignment: *list of audit information*]

3 [assignment: *type of persistent memory*]

4 [selection: *physical, non-physical true, deterministic, hybrid*]

5 [assignment: *list of additional security capabilities*]

6 [selection: *independent bits with Shannon entropy of 7.976 bits per octet, Min-entropy of 7.95 bit per octet, [assignment: other comparable quality metric]*]

Considering the Application Note 12 of [6] in the following paragraphs the additional functions for cryptographic support and access control are defined. These SFRs are not required in the PP [6].

As required by the Application Note 14 of [6] the secure state is described in section 7.2.1 in the rationale for SF.OPC.

Regarding the Application Note 15 of [6] 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).

As required by the Application Note 18 of [6] the automatic response of the TOE is described in section 7.2.1 in the rationale for SF.PHY.

6.1.2 Additional SFRs regarding cryptographic functionality

The (DES coprocessor 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* [7] in accordance with a specified cryptographic algorithm *Triple Data Encryption Algorithm (TDEA)* [8] and cryptographic key sizes *of 112 or 168 bit* [9] that meet the following *list of standards* [10]: *FIPS PUB 46-3 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION DATA ENCRYPTION 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.

The (AES coprocessor of the) TOE shall meet the requirement “Cryptographic operation (FCS_COP.1[AES])” as specified below.

FCS_COP.1[AES]	**Cryptographic operation**
Hierarchical to:	No other components.
FCS_COP.1.1	The TSF shall perform *encryption and decryption* [11] in accordance with a specified cryptographic algorithm *Advanced Encryption Standard (AES) algorithm* [12] and cryptographic key sizes *of 128, 192 or 256 bit* [13] that meet the following *list of standards* [14]:

[7] [assignment: list of cryptographic operations]

[8] [assignment: cryptographic algorithm]

[9] [assignment: cryptographic key sizes]

[10] [assignment: list of standards]

[11] [assignment: list of cryptographic operations]

[12] [assignment: cryptographic algorithm]

[13] [assignment: cryptographic key sizes]

[14] [assignment: list of standards]

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

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.

6.1.3 Additional SFRs regarding access control

Access Control Policy

The hardware shall provide different CPU modes to the IC Dedicated Software and Security IC Embedded Software. The TOE shall separate IC Dedicated Software and Security IC 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 each in a dedicated CPU mode. The hardware shall enforce a separation between different applications (i.e. parts of the Security IC Embedded Software) running on the TOE. An application shall not be able to access hardware components without explicitly granted permission.

The following table provides an overview on the differences in the “xD”-configuration and the “xC”-configuration of the TOE. This must be considered for the following access control policy and the related Security Functional Requirements.

Table 14. Differences between TOE configurations with regard to the Access Control Policy

	TOE major configurations P5CD016V1A, P5CD021V1A, P5CD041V1A, P5CD051V1A, P5CD081V1A, P5CN081V1A	**TOE major configuration P5CC081V1A**	**Remark**
Access to special area in the EEPROM	depends on the MIFARE configuration	fixed	described in detail below, function in general not influenced by the configuration
Access to RAM	depends on the MIFARE configuration	completely accessible	described in detail below, function in general not influenced by the configuration
MIFARE firewall for the RAM access	configuration of the MIFARE firewall supports the separation between IC Dedicated Support Software and Security IC Embedded Software	MIFARE firewall can be configured because the Special Function Registers are accessible as for P5CD016V1A, P5CD021V1A P5CD041V1A, P5CD051V1A, P5CD081V1A and P5CN081V1A. System Mode and User Mode always have access to the	refer to the description of the MIFARE firewall below, function in general not influenced by the configuration

	TOE major configurations P5CD016V1A, P5CD021V1A, P5CD041V1A, P5CD051V1A, P5CD081V1A, P5CN081V1A	TOE major configuration P5CC081V1A	Remark
		RAM, but the configuration can allow access of the MIFARE Operating System to a RAM area. This configuration has no impact since the MIFARE Mode is disabled.	
MIFARE firewall for the Special Function Register access	the configuration of the MIFARE firewall restricts the access of the MIFARE Operating System to the hardware related Special Function Registers	MIFARE firewall can be configured because the Special Function Registers are accessible as for P5CD016V1A, P5CD021V1A P5CD041V1A, P5CD051V1A, P5CD081V1A and P5CN081V1A, but has no impact since the MIFARE Mode is disabled.	refer to the description of the MIFARE firewall below, function in general not influenced by the configuration
supported CPU mode	System Mode, User Mode and MIFARE Mode	System Mode and User Mode. The change of these two modes to the MIFARE Mode is suppressed in this configuration	If the MIFARE Mode is suppressed the related “lcall” commands (CVEC calls) do not force a change of the CPU mode. Thereby they are executed as a normal “lcall” command.

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

The subjects are

- The Security IC Embedded Software i.e. data in the memories of the TOE executed as instructions by the CPU
- The Test-ROM Software as IC Dedicated Test Software
- The Boot-ROM Software as part of the IC Dedicated Support Software
- The MIFARE Operating System as part of the 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 MIFARE Operating System is called MIFARE-EEPROM, the other part Application-EEPROM.

- RAM, which is partitioned into two parts. For the ease of referencing the part reserved for the MIFARE Operating System is called MIFARE-RAM, the other part Application-RAM.
- the code and data in the Memory Segments defined by the Memory Management Unit in Application-ROM, Application-EEPROM and Application-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 Memory Management Unit 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 to configure the MIFARE firewall. These Special Function Registers allow to modify the MIFARE firewall regarding data exchange and Special Function Register access control.
- Special Function Registers used by the MIFARE Operating System. The MIFARE Operating System uses a number of internal Special Function Registers.
- 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 coprocessors 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 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 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.

The values of the Special Function Registers FWCTRLL, FWCTRLH, MXBASL, MXBASH, MXSZL and MXSZH belonging to the group Special Function Registers to configure the MIFARE firewall that define the access rights to the Special Function Registers related to hardware components for code executed in MIFARE Mode and the RAM area used for data exchange between IC Dedicated Support Software (MIFARE OS) and Security IC Embedded Software.

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

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

The amount of the partitioned memory for EEPROM and RAM depends on the configuration of the TOE. For P5CD016V1A, P5CD021V1A, P5CD041V1A, P5CD081V1A, and P5CN081V1A it depends on the MIFARE configuration A, B1 or B4, refer to section 1.4. 128 bytes of the EEPROM are always reserved for the manufacturer.

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* [15] on *all code running on the TOE, all memories and all memory operations* [16].
Dependencies:	FDP_ACF.1 Security attribute based access control
Application Note:	The Access Control Policy shall be enforced by implementing a Memory Management Unit, which maps virtual addresses to physical addresses. The CPU always uses virtual addresses, which are mapped to physical addresses by the Memory Management Unit. Prior to accessing the respective memory address, the Memory Management Unit checks if the access is allowed.
FDP_ACC.1[SFR]	**Subset access control**
Hierarchical to:	No other components.
FDP_ACC.1.1	The TSF shall enforce the *Access Control Policy* [17] on *all code running on the TOE, all Special Function Registers, and all Special Function Register operations* [18].

[15] [assignment: access control SFP]

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

[17] [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 and MIFARE Mode the access rights to the Special Function Registers related to hardware components are provided by the MMU Segment Table and the Special Function Registers to configure the MIFARE firewall. 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* [19] 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* [20].
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 MIFARE-EEPROM* *has read and write access to all data in the MIFARE-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 whole RAM* *Code executed in the MIFARE Mode*

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

[19] [assignment: access control SFP]

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

has read and execute access to all code/data in the Test-ROM,

has read, write and execute access to all code/data in the MIFARE-EEPROM

has read and write access to all data in the MIFARE-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 Application-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 Memory Management Unit,

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 Memory Management Unit,

has read and/or write access to data in the Application-RAM controlled by the MMU Segment Table used by the Memory Management Unit. [21]

FDP_ACF.1.3	The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *Code running in MIFARE Mode has read access to 64 bytes in the Application-ROM storing the "Access Condition Matrix". Code running in MIFARE Mode has access to the Application-RAM defined by the Special Function Register MXBASL, MXBASH, MXSZL and MXSZH. Code running in Boot Mode or MIFARE Mode has read access to the Security Rows stored in the Application-EEPROM. The FameXE coprocessor has read access to the EEPROM and read/write access to the FameXE RAM.* [22]
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the *rules: no explicit rules* [23].
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* [24] to objects based on the following: *all subjects and objects and the*

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

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

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

[24] [assignment: access control SFP]

	attributes CPU mode, the MMU Segment Table and the Special Function Registers FWCTRLL and FWCTRLH [25].
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 read Special Function Registers to configure the MIFARE firewall and to read/write Special Function Registers used by the MIFARE Operating System. Access to Special Function Registers related to hardware components is based on the access rights determined by the Special Function Registers FWCTRLL and FWCTRLH.* *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, Special Function Registers to configure the MIFARE firewall 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* [26].
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. The Special Function Register CLKSEL of the group Special Function Registers related to hardware components can be read in the MIFARE Mode regardless of the MIFARE firewall settings given by FWCTRLL and FWCTRLH.* [27]
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*

[25] [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]

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

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

	Function Register RNR of the group Special Function Registers related to hardware components is read-only. The Special Function Registers AKEY and DKEY of the group Special Function Registers related to hardware components are not readable. [28]
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization

Implications of the Access Control Policy

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

Code executed in 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 Memory Management Unit. Only small memory areas are used for data exchange between the MIFARE Operating System and the Security IC Embedded Software. Furthermore, the exchange area in RAM is fully controlled by code running in System Mode.

Code executed in the System Mode can administrate the configuration of Memory Management Unit, 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 Memory Management Unit, 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* [29] to provide *restrictive* [30] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow *no subject* [31] 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

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

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

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

[31] [assignment: the authorised identified 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 can not be configured at all. The TOE does not provide objects or information that can be created, since it provides access to memory areas. The definition of objects that are stored in the TOE's memory is subject to the Security IC Embedded Software.
FMT_MSA.3[SFR]	**Static attribute initialization**
Hierarchical to:	No other components.
FMT_MSA.3.1	The TSF shall enforce the *Access Control Policy* [32] to provide *restrictive* [33] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow *no subject* [34] to specify alternative initial values to override the default values when an object or information is created.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
Application Note:	The TOE does not provide objects or information that can be created since no further security attributes can be derived (i.e. the set of Special Function Registers that contain security attributes is fixed). The definition of objects that are stored in the TOE's memory is subject to the Security IC Embedded Software.

The TOE shall meet 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* [35] to restrict the ability to *modify* [36] the security attributes *Special Function Registers to configure the MMU segmentation* [37] to *code executed in the System Mode* [38].
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

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

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

[34] [assignment: the authorised identified roles]

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

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

[37] [assignment: list of security attributes]

[38] [assignment: the authorised identified roles]

Application Note:	The MMU Segment Table is not included in this requirement because it is located in the memory of the TOE and access to it is possible for every role that has access to the respective memory locations. This component does not include any management functionality for 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* [39] to restrict the ability to *modify* [40] the security attributes *defined in Special Function Registers* [41] to *code executed in a CPU mode which has write access to the respective Special Function Registers* [42].
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 call vector (SVEC) or configuration vector (CVEC) 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.* [43]
Dependencies:	No dependencies

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

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

[41] [assignment: list of security attributes]

[42] [assignment: the authorised identified roles]

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

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.

6.2 Security Assurance Requirements

Table 15 below lists all security assurance components that are valid for this Security Target. With one exception these security assurance components are required by EAL5 (see section 2.2) or by the PP “Security IC Platform Protection Profile” [6].

The exception is the component ASE_TSS.2 which is chosen as an augmentation in this Security Target to give architectural information on the security functionality of the TOE.

Considering Application Note 21 of [6] the column “Required by” shows the differences in the requirements of security assurance components between the PP [6] and the Security Target. The entry “EAL5 / PP” denotes, that an SAR is required by both EAL5 and the requirement of the PP [6], “EAL5” means that this requirement is due to EAL5 and beyond the requirement of the PP [6], and “PP” identifies this component as a requirement of the PP which is beyond EAL5. The augmentation ASE_TSS.2 chosen in this Security Target is denoted by "ST". The refinements of the PP [6], that must be adapted for EAL5, are described in section 6.2.1.

Table 15. Security Assurance Requirements

SAR	Title	Required by
ADV_ARC.1	Security architecture description	EAL5 / PP
ADV_FSP.5	Complete semi-formal functional specification with additional error information	EAL5
ADV_IMP.1	Implementation representation of the TSF	EAL5 / PP
ADV_INT.2	Well-structured internals	EAL5
ADV_TDS.4	Semiformal modular design	EAL5
AGD_OPE.1	Operational user guidance	EAL5 / PP
AGD_PRE.1	Preparative procedures	EAL5 / PP
ALC_CMC.4	Production support, acceptance procedures and automation	EAL5 / PP
ALC_CMS.5	Development tools CM coverage	EAL5
ALC_DEL.1	Delivery procedures	EAL5 / PP
ALC_DVS.2	Sufficiency of security measures	PP
ALC_LCD.1	Developer defined life-cycle model	EAL5 / PP
ALC_TAT.2	Compliance with implementation standards	EAL5
ASE_CCL.1	Conformance claims	EAL5 / PP
ASE_ECD.1	Extended components definition	EAL5 / PP

SAR	Title	Required by
ASE_INT.1	ST introduction	EAL5 / PP
ASE_OBJ.2	Security objectives	EAL5 / PP
ASE_REQ.2	Derived security requirements	EAL5 / PP
ASE_SPD.1	Security problem definition	EAL5 / PP
ASE_TSS.2	TOE summary specification with architectural design summary	ST
ATE_COV.2	Analysis of coverage	EAL5 / PP
ATE_DPT.3	Testing: modular design	EAL5
ATE_FUN.1	Functional testing	EAL5 / PP
ATE_IND.2	Independent testing - sample	EAL5 / PP
AVA_VAN.5	Advanced methodical vulnerability analysis	PP

6.2.1 Refinements of the Security Assurance Requirements

The Security Target claims conformance to the PP [6] and therefore it has to be conform to the refinements of the TOE security assurance requirements (see Application Note 22 in [6]). Because the refinements in the PP [6] 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 16 lists the influences of the refinements of the PP [6] on the Security Target. Most of the refined security assurance components have the same level in both documents (PP [6] and Security Target). The following two subsections apply the refinements to ALC_CMS.5 and ADV_FSP.5 which are different between the PP [6] and the Security Target.

Table 16. Security Assurance Requirements, overview of differences of refinements

Refined in PP [6]	Influence on Security Target
ALC_DEL	Same as in PP, refinement valid without change
ALC_DVS	Same as in PP, refinement valid without change
ALC_CMS	ALC_CMS.5, refinements valid without change
ALC_CMC	Same as in PP, refinement valid without change
ADV_ARC	Same as in PP, refinement valid without change
ADV_FSP	ADV_FSP.5, refinements have to be adapted
ADV_IMP	Same as in PP, refinement valid without change
ATE_COV	Same as in PP, refinement valid without change
AGD_OPE	Same as in PP, refinement valid without change
AGD_PRE	Same as in PP, refinement valid without change

Refined in PP [6]	Influence on Security Target
AVA_VAN	Same as in PP, refinement valid without change [44]

6.2.1.1 Refinements regarding CM scope (ALC_CMS)

This Security Target requires a higher evaluation level for the CC family ALC_CMS, namely ALC_CMS.5 instead of ALC_CMS.4. The refinement of the PP [6] regarding ALC_CMS.4 is a clarification of the configuration item “TOE implementation representation”. Since in ALC_CMS.5, the content and presentation of evidence element ALC_CMS.5.1C 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 ALC_CMS.4 can be found in section 6.2.1.3 of [6] and is not cited here.

6.2.1.2 Refinements regarding functional specification (ADV_FSP)

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

The higher level ADV_FSP.5 requires a Functional Specification in a “semi-formal style" (ADV_FSP.5.2C).

The component ADV_FSP.5 enlarges the scope of the error messages to be described from those resulting from an invocation of a TSFI (ADV_FSP.5.6C) to also those not resulting from an invocation of a TSFI (ADV_FSP.5.7C). For the latter a rationale shall be provided (ADV_FSP.5.8C).

Since the higher level ADV_FSP.5 only affects the style of description and the scope of and rationale for error messages, the refinements can be applied without changes and are valid for ADV_FSP.5.

The refinement of the original component ADV_FSP.4 can be found in section 6.2.1.6 of the Protection Profile [6] and is not cited here.

6.3 Security Requirements Rationale

6.3.1 Rationale for the security functional requirements

Section 6.3.1 in [6] provides a rationale for the mapping between security functional requirements and security objectives defined in the PP [6]. The mapping is reproduced in the following table.

Table 17. Security Requirements versus Security Objectives

Objective	TOE Security Functional Requirements
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”

44 According to the Application Note 30 in [6] the Security Target should indicate the version of the document Supporting Document Mandatory Technical Document Application of Attack Potential to Smartcards [7] used for the vulnerability analysis. The current version is given in the bibliography.

Objective	TOE Security Functional Requirements
O.Phys-Probing	FPT_PHP.3 “Resistance to physical attack”
O.Malfunction	FRU_FLT.2 “Limited fault tolerance FPT_FLS.1 “Failure with preservation of secure state”
O.Phys-Manipulation	FPT_PHP.3 “Resistance to physical attack”
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_PHP.3
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
O.Identification	FAU_SAS.1 “Audit storage”
O.RND	FCS_RNG.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
OE.Plat-Appl	not applicable
OE.Resp-Appl	not applicable
OE.Process-Sec-IC	not applicable

The Security Target additionally defines the SFRs for the TOE that are listed in Table 18. 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 18. Mapping of security objectives and requirements

Objective	TOE Security Functional Requirement
O.HW_DES3	FCS_COP.1[DES]
O.HW_AES	FCS_COP.1[AES]
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]

Objective	TOE Security Functional Requirement
	FMT_MSA.1[SFR] FMT_SMF.1
O.SFR_ACCESS	FDP_ACC.1[SFR] FDP_ACF.1[SFR] FMT_MSA.3[SFR] FMT_MSA.1[SFR] FMT_SMF.1
OE.Check-Init	not applicable

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 “AES Functionality” (O.HW_AES) is as follows:

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

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 require 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 require to implement an area based memory access control as demanded by O.MEM_ACCESS. Therefore, FDP_ACC.1[MEM] with its SFP is suitable to meet 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 Security IC Embedded Software. Since the pointer to the MMU Segment Table can only be changed in System Mode and this protection is implemented by access control to the respective Special Function Registers, both iterations are needed for O.MEM_ACCESS.

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” require 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 Memory Management Unit can only be accessed in the System Mode. The Special Function Register required to use hardware components like e.g. the coprocessors 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 Memory Management Unit is configured to allow this.

The security functional requirement “Security attribute based access control (FDP_ACF.1[SFR])” with the related Security Function Policy “Access Control Policy”

exactly require 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.

6.3.2 Dependencies of security functional requirements

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

The following discussion demonstrates how the dependencies defined by Part 2 of the Common Criteria for the requirements specified in sections 6.1.2 and 6.1.3 are satisfied.

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

Table 19. 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	See discussion below
FCS_COP.1[AES]	FDP_ITC.1 or FDP_ITC.2, or FCS_CKM.1 FCS_CKM.4	See discussion below
FDP_ACC.1[MEM]	FDP_ACF.1	Yes, by FDP_ACF.1[MEM]
FDP_ACC.1[SFR]	FDP_ACF.1	Yes, by FDP_ACF.1[SFR]
FDP_ACF.1[MEM]	FDP_ACC.1 FMT_MSA.3	Yes, by FDP_ACC.1[MEM] Yes
FDP_ACF.1[SFR]	FDP_ACC.1 FMT_MSA.3	Yes, by FDP_ACC.1[SFR] Yes

Security Functional Requirement	Dependencies	Fulfilled by security requirements in this ST
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 Security IC Embedded Software must ensure that the additional security functional requirements FCS_COP.1[DES] and FCS_COP.1[AES] are used as specified and that the User Data processed by the related security functionality is protected as defined for the application context.

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

The functional requirements [FDP_ITC.1, or FDP_ITC.2 or FCS_CKM.1] and FCS_CKM.4 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. Therefore the Security IC Embedded Software 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 must be fulfilled by the Security IC Embedded Software. The definition and maintenance of the roles that act on behalf of the functions provided by the hardware must be subject of the Security IC Embedded Software.

6.3.3 Rationale for the Assurance Requirements

The selection of assurance components is based on the underlying PP [6]. The Security Target uses the same augmentations as the PP, but chooses a higher assurance level. The level EAL5 is chosen in order to meet assurance expectations of digital signature applications and electronic payment systems. Additionally, the requirement of the PP [6] to choose at least EAL4 is fulfilled.

The rationale for the augmentations is the same as in the PP. The assurance level 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 section 6.3.3 of [6], 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_VAN.5 was chosen by the PP [6] in order to assure that even these attackers cannot successfully attack the TOE.

6.3.4 Security Requirements are Internally Consistent

The discussion of security functional requirements and 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 requirement FCS_COP.1[DES], FCS_COP.1[AES] 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], FCS_COP.1[AES] and of FDP_ACC.1 with FDP_ACF.1 as well as the dependent security functional requirements.

A Security IC hardware platform requires Security IC Embedded Software to build a secure product. Thereby the Security IC Embedded Software must support the security functionality of the hardware and implement a sufficient management of the security services 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 behaviour of the TOE.

7. TOE Summary Specification

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

7.1 Portions of the TOE Security Functionality

The TOE Security Functionality (TSF) directly corresponds to the TOE security functional requirements defined in chapter 6. It is split into Security Services (SS) and Security Features (SF), which are applicable to phases 4 to 7 of the Security IC product life-cycle.

Note: Parts of the security functionality are configured at the end of phase 3 and all security functionality is active after phase 3.

The TOE also comprises security mechanisms, which are not listed as security functionality in the following. Such mechanisms do not provide a complete portion of TOE security functionality, but they can be used to support a portion of security functionality implemented by the Security IC Embedded Software, e.g. the FameXE coprocessor for asymmetric cryptographic algorithms or the CRC calculation for the control of data integrity.

7.1.1 Security Services

SS.RNG: Random Number Generator

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

The TSF provides a hardware test functionality, which can be used by the Security IC Embedded Software to detect faults in the hardware of the Random Number Generator.

According to “AIS31” [5] 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 the generation of seeds for DRNGs.

SS.HW_DES: Triple-DES coprocessor

The TOE provides the Triple Data Encryption Algorithm (TDEA) according to the Data Encryption Standard (DES). SS.HW_DES is a modular basic cryptographic function, which provides the TDEA algorithm as defined by FIPS PUB 46 by means of a hardware coprocessor and supports (a) the 3-key Triple-DEA algorithm according to keying option 1 and (b) the 2-key Triple DEA algorithm according to keying option 2 in FIPS PUB 46-3 [18]. The two/three 56-bit keys (112-/168-bit) for the 2-key/3-key Triple DES algorithm shall be provided by the Security IC Embedded Software. For encryption the Security IC Embedded Software provides 8 bytes of the plain text and SS.HW_DES calculates 8 bytes cipher text. The calculation output is read by the Security IC Embedded Software. For decryption the Security IC Embedded Software provides 8 bytes of cipher text and SS.HW_DES calculates 8 bytes plain text. The calculation output is read by the Security IC Embedded Software.

SS.HW_AES: AES coprocessor

The TOE provides the Advanced Encryption Standard (AES) algorithm according to the Advanced Encryption Standard as defined by FIPS PUB 197 [19]. SS.HW_AES is a modular basic cryptographic function, which provides the AES algorithm by means of a

hardware coprocessor and supports the AES algorithm with three different key lengths of 128, 192 or 256 bit. The keys for the AES algorithm shall be provided by the Security IC Embedded Software. For encryption the Security IC Embedded Software provides 16 bytes of the plain text and SS.HW_AES calculates 16 bytes cipher text. The calculation output is read by the Security IC Embedded Software. For decryption the Security IC Embedded Software provides 16 bytes of cipher text and SS.HW_AES calculates 16 bytes plain text. The calculation output is read by the Security IC Embedded Software.

7.1.2 Security Features

SF.OPC: Control of Operating Conditions

SF.OPC ensures correct operation of the TOE (functions offered by the microcontroller including the standard CPU as well as the Triple-DES coprocessor, AES coprocessor, the arithmetic coprocessor, the memories, registers, I/O interfaces and the other system peripherals) during execution of the IC Dedicated Support Software and Security IC Embedded Software. This includes all specific security mechanisms of the TOE, which are able to provide an active response.

The TOE ensures its correct operation and prevents from any malfunction using the following mechanisms: filtering of power supply and clock frequency input as well as monitoring of voltage supply, clock frequency input and the temperature of the chip by means of sensors. There are multiple sensors for the different ISO/IEC 7816 voltage classes and the contactless operation mode. Light sensors are distributed over the chip surface and used to detect light attacks. Thresholds of the parameters, which are monitored by the mechanisms, are set appropriate to the range where the TOE ensures its correct operation. In addition to the light sensors the EEPROM provides two functions to detect light attacks. The Security IC 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 the Secure Fetch Technology™ or other special circuitry to detect fault injection attacks. These are the Program Counter, the stack pointer, the PSWH register, the MMU address cache registers, the DES coprocessor, AES coprocessor, and the FameXE coprocessor.

If one of the monitored parameters is out of the specified range, either (i) a reset is forced and the actually running program is aborted or (ii) an exception is raised which interrupts the program flow and allows a reaction of the Security IC Embedded Software. A reset is forced by the sensors for voltage, frequency, temperature and light, the Secure Fetch Technology™ and the single fault injection detection in the MMU address cache registers. An exception is forced by the EEPROM light detector and the other single fault injection detection circuitry. In case the inverse error correction of the EEPROM is enabled (refer to section 1.4.1.2) the probability to detect fault injection errors increases and the error correction logic raises an exception when detecting an error. In case the TOE resets all components of the TOE are initialized with their reset values and the TOE provides a reset cause indicator to the Security IC Embedded Software. In the case an exception is raised an indicator for the reason of the exception is provided.

Test Mode is disabled before TOE delivery. In all other modes except Test Mode the TOE automatically enables the sensors when operated. Furthermore the TOE defends the sensors from being disabled by the Security IC Embedded Software. The assignment which sensor raises an exception or forces a reset is hard-wired and can not be changed by the Security IC Embedded Software.

The TOE also controls the specified range of the stack pointer. The stack pointer and the control logic are implemented threefold for the User Mode, System Mode and Super System Mode (comprising Boot Mode, Test Mode and MIFARE Mode). An exception is generated in case the specified limits are exceeded.

In addition, SF.OPC comprises a sensor, which checks the high voltage of the write process to the EEPROM during each 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).

SF.PHY: Protection against Physical Manipulation

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

The protection of the TOE comprises several security mechanisms in design and construction, which make reverse-engineering and tamper attacks more difficult. These mechanisms comprise dedicated shielding techniques for different components and specific encryption mechanisms for the memories. SF.PHY supports the efficiency of other portions of the security functionality.

SF.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.

SF.LOG: Logical Protection

SF.LOG implements security mechanisms to limit or eliminate the information in the shape and amplitude of signals or in the time between events, which might be found by measuring such signals. This comprises the power supply and signals on other pads, which are not intentionally used for communication by the terminal or the Security IC Embedded Software. Thereby SF.LOG prevents from disclosure of User Data and TSF data stored and/or processed in the Security IC through measurement of power consumption and subsequent complex signal analysis. This protection of the TOE is enforced by several security mechanisms in the design, which support other portions of security functionality.

The Triple-DES coprocessor includes security mechanisms 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 AES coprocessor includes security mechanisms to prevent SPA/DPA analysis of shape and amplitude of the power consumption and ensures that the calculation time is with respect to the key length independent from any plain/cipher text.

The FameXE coprocessor provides measures to prevent timing attacks on basic modular function. The calculation time of an operation depends on the lengths of the operands, but not on the value of the operands, with the following exceptions: multiplication with reduction, modular inversion and modular division. These three operations have no constant timing due to correction cycles that are needed based on the calculation method. In addition, mechanisms are included, which 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 by the Security IC Embedded Software when using the FameXE.

Additional security mechanisms being configured by the Security IC Embedded Software comprise (i) 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)

Some mechanisms described for SF.PHY (e.g. the encryption mechanisms) and for SF.OPC (e.g. the filter mechanisms) also support SF.LOG.

SF.COMP: Protection of Mode Control

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

Control of the CPU mode for Boot Mode, Test Mode and MIFARE Mode prevents from abuse of test functions after TOE delivery. It also inhibits abuse of features, which are used during start-up or reset to configure the TOE. The initial – but not user visible – CPU mode during start-up or reset is the Boot Mode. Hardware circuitry determines whether Test Mode is available or not. If available, the TOE jumps to the IC Dedicated Test Software in Test Mode. Otherwise, the TOE switches to System Mode – the initial user visible CPU mode – and starts execution of the Security IC Embedded Software.

The protection of electronic fuses ensures secure storage of configuration and calibration data, which are set up in Test Mode. SF.COMP protects CPU mode switches regarding Boot Mode, Test Mode and MIFARE Mode in the following way: Switching from Boot Mode to Test Mode or MIFARE Mode is allowed, switching from these modes back to Boot Mode is prevented. Switching to Test Mode is prevented as well after TOE delivery, because Test Mode then is permanently disabled. SF.COMP also ensures that Boot Mode is active only in the boot phase during start-up or reset of the TOE, and can not be invoked afterwards. Therefore, once the TOE has left the test phase and each time the TOE completed start-up or reset, the MIFARE Mode is the only one available out of Super System Mode. Super System Mode is indicated by PSWH.SSM being set and means, that one CPU mode out of Boot Mode, Test Mode and MIFARE Mode is active. SF.COMP controls the mode, which is used, when bit PSWH.SSM is set.

The protection of electronic fuses especially ensures that configuration options with regard to the security functionality can not be changed, abused or influenced in any way. SF.COMP ensures that activation or deactivation of security mechanisms can not be influenced by the Security IC Embedded Software so that the TSF provides self-protection against interference and tampering by untrusted Security IC Embedded Software.

The TSF controls access to the Security Rows, the top-most 128 Bytes of the EEPROM memory, accessible at reserved addresses in the memory map. The available EEPROM memory space for the Security IC Embedded Software is reduced by this area.

SF.COMP provides three memory areas in the Security Rows that can be used by the Security IC Embedded Software. These are

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

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

If the Card Disable Function is used (refer to section 1.4.1.2) SF.COMP inhibits any start-up of the Security IC Embedded Software once the Security IC Embedded Software disables the card.

SF.COMP also provides the FabKey Area in which initialization and identification data can be stored. The FabKey area does not belong to the Security Rows and is not protected by hardware mechanisms. The FabKey Area as well as the Security Rows can be used by SF.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 Security IC 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) Security IC (e.g. 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.

SF.COMP limits the capabilities of the test functions and provides test personnel during phase 3 with the capability to store identification and/or pre-personalization data and/or supplements of the Security IC Embedded Software in the EEPROM. SF.COMP provides self-protection against interference and tampering by untrusted subjects both in the Test Mode and in the other modes. It also enforces the separation of domains regarding the IC Dedicated Software and the Security IC Embedded Software.

SF.MEM_ACC: Memory Access Control

SF.MEM_ACC controls access of any subject (program code comprising processor instructions) to the memories of the TOE through the Memory Management Unit. 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 passed from the Memory Management Unit to the memory interfaces to access the memories. The access control is performed in two ways:

- Memory partitioning: Each memory type ROM, RAM and EEPROM is partitioned into two parts. In Boot Mode, MIFARE Mode, System Mode and User Mode the CPU has access to only one part of each memory type. Access to both parts of each type is allowed in Test Mode for testing.
- Memory segmentation in User Mode: The three accessible parts of the memory in ROM, RAM and EEPROM can be segmented into smaller areas. Access rights (readable, writeable or executable) can be defined for these segments. In addition, access rights to Special Function Registers related to hardware components can be defined for code that is executed in a segment.

Memory partitioning is fixed and can not be changed. It is determined during production of the TOE and is solely dependent on the MIFARE configuration (refer to section 0).

Memory segmentation can be defined in System Mode. The segmentation is active when the CPU switches to User Mode. The segments, their access rights 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 to Special Function Registers for code that is executed in the segment. The address offset is used to relocate the segment anywhere in the memory map. The resulting address computed by the Memory Management Unit can not overrule memory partitioning. Up to 64 segments can be defined in the MMU Segment Table. Special configurations of the memory access rights allow to specify less than 64 segments and to split the MMU Segment Table into several parts being stored at different locations in memory.

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

In addition, SF.MEM_ACC permanently checks whether accessed addresses point to physically implemented memory. Access to forbidden memory addresses in User Mode and accesses outside the boundaries of the physical memory are notified by raising an exception.

As stated above the Memory Management Unit handles access rights to Special Function Registers related to hardware components for code running in User Mode. This information is used by SF.SFR_ACC to grant or block a certain access. The access rights can be defined for up to 16 groups of Special Function Registers, which are related to 16 hardware components. SF.SFR_ACC receives the access rights to these 16 groups from the Memory Management Unit as well for the other CPU modes. In Boot Mode, Test Mode and System Mode the Memory Management Unit indicates full access to the 16 groups. In MIFARE Mode the Memory Management Unit indicates access rights as set in two Special Function Registers, which can not be modified in MIFARE Mode. Thus, MIFARE Mode can be restricted in its access to the 16 hardware components on demand of the Security IC Embedded Software. Note that SF.MEM_ACC only provides the access rights to SF.SFR_ACC, the access control is enforced by SF.SFR_ACC.

SF.SFR_ACC: Special Function Register Access Control

SF.SFR_ACC controls access to the Special Function Registers and CPU mode switches based on Special Function Register PSWH.

SF.SFR_ACC implements 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].

The function of the Special Function Register and the CPU mode determine, whether read and/or write access to a Special Function Register is allowed or not. For example, read access to the DES key register is write-only according to its function and its write access is granted depending on the CPU mode. Similar for the output register of the Random Number Generator, which is read-only based on its function, and read access is granted based on the CPU mode.

SF.SFR_ACC ignores accesses to Special Function Registers, which are not allowed. Ignoring means that a write access has no influence and/or a read access always returns a fixed value independent of the true value of the Special Function Register.

Some Special Function Registers are implemented threefold, one for User Mode, a second one for System Mode and a third one for Super System Mode meaning Boot Mode, Test Mode and MIFARE Mode. Hence, such Special Function Registers are already separated from the outset.

SF.SFR_ACC relies on access rights to Special Function Registers related to hardware components, which are provided by SF.MEM_ACC. Access rights to all other Special Function Registers are pre-defined and can not be changed.

This implies that code running in User Mode or MIFARE Mode is not able to use SS.RNG, SS.HW_DES, and SS.HW_AES until access to the respective group of Special Function Registers is explicitly granted by code running in System Mode. This holds for all 16 hardware components, which are controlled by the 16 groups of Special Function Registers related to hardware components.

SF.SFR_ACC also implements transitions among CPU modes based on Special Function Register PSWH. This Special Function Register contains two bits, which are PSWH.SSM and PSWH.SM. Bit PSWH.SSM indicates Super System Mode when set, which means, that one out of Boot Mode, Test Mode and MIFARE Mode is active. Bit PSWH.SM indicates System Mode when set. If both bits are zero, the CPU operates in User Mode.

The CPU mode changes by the following operations.

- Call of a system call vector (SVEC) address or a configuration vector (CVEC) address. A call of a SVEC sets bit PSWH.SM and enables System Mode, a call of a CVEC sets bit PSWH.SSM and enables MIFARE Mode. Calls of SVEC addresses are only allowed in User Mode, otherwise an exception is raised. For P5CC081V1A calls to the CVEC addresses do not set bit PSWH.SSM. Instead, the call is executed like any other call.
- Execution of an exception or interrupt. Any event that leads to the execution of an exception sets bit PSWH.SM. Interrupts can be executed in User Mode or in System Mode. The Security IC Embedded Software running in System Mode can configure this option at run time and based on this configuration bit PSWH.SM is modified or not.
- Return from an exception/interrupt or vector call with a RETI instruction. This restores the value of PSWH to the value before the event occurred. A RETI in User Mode is allowed only in case interrupts are allowed to be executed in User Mode and an interrupt is actually active, otherwise an exception is raised.
- Execution of an LCALL/ACALL/ECALL instruction to a specific address. A call of address 0x800000 in System Mode enables User Mode and starts execution at this (virtual) address. This is similar to a CVEC or SVEC call, but no return address is pushed onto the stack.
- Direct modification of the two bits in PSWH. Hardware provided by SF.SFR_ACC ensures that the bits can only be cleared. Therefore it is not possible for code running in User Mode to enter System Mode, but System Mode can switch to User Mode.

Two CPU modes are available to the Security IC Embedded Software, which are System Mode and User Mode. System Mode is the more privileged CPU mode since it allows access to all Special Function Registers of the hardware components and for system management (i.e. configuration of Memory Management Unit, clock settings and some mechanisms provided by SF.LOG). User Mode is the less privileged, but with regard to the hardware components it can be made as powerful as System Mode.

SF.SFR_ACC and SF.COMP together ensure that other CPU modes are not available to the Security IC Embedded Software, but reserved for specific purposes fulfilled by the IC Dedicated Software. In addition, SF.MEM_ACC provides separation of the memories and access control information.

7.2 TOE Summary Specification Rationale

7.2.1 Rationale for the portions of the TOE security functionality

The following table provides a mapping of portions of the TOE security functionality to the Security Functional Requirements. The mapping is described in detail in the text following the table (only in the full version of the security target).

Table 20. Mapping of Security Functional Requirements and the portions of the TOE Security Functionality

	SS.RNG	SS.HW_DES	SS.HW_AES	SF.OPC	SF.PHY	SF.LOG	SF.COMP	SF.MEM_ACC	SF.SFR_ACC
FAU_SAS.1					X		X		
FCS_RNG.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				
FRU_FLT.2				X	X				
FCS_COP.1[DES]		X			X				
FCS_COP.1[AES]			X		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

An "X" in the above table means that the specific portion of the TOE security functionality realizes or supports the functionality required by the respective Security Functional Requirement.

As already stated in the definition of the portions of the TOE security functionality there are additional security mechanisms, which can contribute to security when they are sufficiently controlled by the Security IC Embedded Software. The CRC calculation can

be used to verify the integrity of memory areas defined by the Security IC Embedded Software, the FameXE coprocessor can be used to build leakage-resistant asymmetric cryptographic algorithms.

7.2.2 Security architectural information

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

As described in section 7.2.1, the aspects self-protection and non-bypassability are implemented by SF.PHY, SF.OPC, and SF.COMP.

SF.PHY covers the physical protection of the TOE. SF.OPC contributes by covering the aspects failure with preservation of secure state and limited fault tolerance. SF.COMP limits capability and availability of the Test Features.

8. Annexes

8.1 Further Information contained in the PP

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

8.2 Glossary and Vocabulary

Administrator	(in the sense of the Common Criteria) The TOE may provide security functionality which can or need to be administrated (i) by the Security IC 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.
Application Data	All data managed by the Security IC Embedded Software in the application context. Application data comprise all data in the final Security IC.
Boot Mode	CPU mode of the TOE dedicated to start-up and reset of the TOE. This mode is not accessible for the Security IC Embedded Software.
Composite Product	Integrator Role installing or finalizing the IC Embedded Software and the applications on platform transforming the TOE into the unpersonalized Composite Product after TOE delivery The TOE Manufacturer may implement IC Embedded Software delivered by the Security IC Embedded Software Developer before TOE delivery (e.g. if the IC Embedded Software is implemented in ROM or is stored in the non-volatile memory as service provided by the IC Manufacturer or IC Packaging Manufacturer).
Composite Product Manufacturer	The Composite Product Manufacturer has the following roles (i) the Security IC Embedded Software Developer (Phase 1), (ii) the Composite Product Integrator (Phase 5) and (iii) 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. The customer of the TOE Manufacturer, who receives the TOE during TOE Delivery. The Composite Product Manufacturer includes the Security IC Embedded Software developer and all roles after

	TOE Delivery up to Phase 6 (refer to [6], Figure 2 on page 240H10 and Section 7.1.1)
CPU mode	Mode in which the CPU operates. The TOE supports five CPU modes, which are Boot Mode, Test Mode, MIFARE Mode, System Mode and User Mode.
exception interrupt	Non-maskable interrupt of program execution jumping to fixed addresses (depending on the exception source) and enabling System Mode. Sources of exceptions are hardware breakpoints, single fault injection detections, illegal instructions, stack overflows, unauthorized system call vector calls, execution of RETI instruction in User Mode, and the MMU exceptions access violation and access collision.
FabKey Area	A memory area in the EEPROM containing data, which are programmed during testing by the IC Manufacturer. The amount of data and the type of information can be selected by the customer.
IC Dedicated Software	IC proprietary software embedded in a Security 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	That 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	That 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.
End-consumer	User of the Composite Product in Phase 7
Initialization Data	Initialization Data defined by the TOE Manufacturer to identify the TOE and to keep track of the Security IC's production and further life-cycle phases are considered as belonging to the TSF data. These data are for instance used for traceability and for TOE identification (identification data).
Integrated Circuit (IC)	Electronic component(s) designed to perform processing and/or memory functions.
Memory	IC hardware component, that stores code and/or data, i.e. ROM, RAM or EEPROM of the TOE.
Memory Management Unit	The MMU maps the virtual addresses used by the CPU into the physical addresses of RAM, ROM and EEPROM. This mapping is done based on (a)

	memory partitioning and (b) memory segments for code running in User Mode. Memory partitioning is fixed, whereas up to 64 memory segments can be configured individually. Each segment can be (i) positioned and sized (ii) enabled and disabled, (iii) configured for access rights in terms of read, write and execute in User Mode and (iv) configured for User Mode access rights to Special Function Registers related to hardware components of code executed in this segment.
Memory Segment	Address space provided by the Memory Management Unit according to the configuration in the MMU Segment Table. A memory segment defines a memory area, are accessible for code running in User Mode. Memory segments may address RAM, ROM and EEPROM.
MIFARE	Contactless smartcard interface standard complying with ISO/IEC 14443 A.
MIFARE Mode	CPU mode of the TOE dedicated to execution of the MIFARE Operating System, which is part of the IC Dedicated Support Software. This mode is not accessible for the Security IC Embedded Software.
MMU Segment Table	This structure defines the memory segments for code running in User Mode, which are controlled by the MMU. The structure can be located anywhere in the memory that is available in System Mode. It also contains User Mode access rights to Special Function Registers related to hardware components of code executed in each segment.
Pre-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.
S^2C	Smart card interface standard complying with ISO/IEC 18092.
Security IC	(as used in the PP [6]) Composition of TOE, Security IC Embedded Software, User Data and package (Security IC carrier).
Security IC Embedded Software	Software embedded in a Security IC and normally not being developed by the IC Designer. The Security IC Embedded Software is designed in Phase 1 and embedded into the Security IC in Phase 3 or in later phases of the Security IC product life-cycle. Some part of that software may actually implement a Security IC application others may provide standard services. Nevertheless, this distinction does not matter here so that the Security IC Embedded

	Software can be considered as being application dependent whereas the IC Dedicated Software is definitely not.
Security IC Product	Composite product which includes the Security Integrated Circuit (i.e. the TOE) and the Embedded Software and is evaluated as composite target of evaluation in the sense of the Supporting Document
Security Rows	Top-most 128 bytes of the EEPROM memory reserved for configuration purposes as well as dedicated memory area for the Security IC Embedded Software to store life-cycle information about the TOE.
Special Function Registers	Registers used to access and configure the functions for communication with an external interface device, the cryptographic coprocessors for Triple-DES or AES, the FameXE coprocessor for basic arithmetic functions to perform asymmetric cryptographic algorithms, the random numbers generator and chip configuration.
Super System Mode	This term represents either Boot Mode, Test Mode or MIFARE Mode.
System Mode	CPU mode of the TOE with unrestricted access to the hardware resources. The Memory Management Unit can be configured in System 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 of the TOE for its configuration and execution of the IC Dedicated Test Software. The Test Mode is permanently and irreversible disabled after production testing. Specific Special Function Registers are accessible in Test Mode for test purposes.
TOE Delivery	The period when the TOE is delivered which is (refer to [6], Figure 2 on page 242H10) 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 packaged products.
TOE Manufacturer	The TOE Manufacturer must ensure that all requirements for the TOE (as defined in [6], Section 243H1.2.2) and its development and production environment are fulfilled (refer to [6], Figure 2 on page 24H10). 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 packaged products, he has the role of the (iii) IC Packaging Manufacturer (Phase 4) in addition.
TSF data	Data created by and for the TOE, which might affect the operation of the TOE. This includes information about the TOE's configuration, if any is coded in non-volatile non-programmable memories (ROM), in specific circuitry, in non-volatile programmable memories (for instance EEPROM) or a combination thereof.
User	(in the sense of the Common Criteria) The TOE serves as a platform for the Security IC Embedded Software. Therefore, the "user" of the TOE (as used in the Common Criteria assurance class AGD: guidance) is the Security IC Embedded Software. Guidance is given for the Security IC Embedded Software Developer. On the other hand the Security IC (with the TOE as a major element) is used in a terminal where communication is performed through the ISO/IEC interface provided by the TOE. Therefore, another "user" of the TOE is the terminal (with its software).
User Data	All data managed by the Security IC Embedded Software in the application context. User data comprise all data in the final Security IC except the TSF data.
User Mode	CPU mode of the TOE. Access to memories is controlled by the Memory Management Unit. Access to Special Function Registers is restricted.

8.3 List of Abbreviations

CC	Common Criteria Version 3.1
CIU	Contactless Interface Unit
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
NFC	Near Field Communication
PKC	Public Key Cryptography
PP	Protection Profile
PSWH	Program Status Word (High byte)
SAR	Security Assurance Requirement
SFR	as abbreviation of the CC term: Security Functional Requirement, as abbreviation of the technical term of the SmartMX-family: Special Function Register[45]
SIM	Subscriber Identity Module
SOF	Strength of Function
SF	Security Feature
SS	Security Service
ST	Security Target
TOE	Target of Evaluation
TRNG	True Random Number Generator
TSC	TSF Scope of Control
TSF	TOE Security Functionality
TSFI	TSF Interface
TSP	TOE Security Policy
UART	Universal Asynchronous Receiver and Transmitter

[45] To avoid confusion this Security Target does not use SFR as abbreviation for Special Function Register in the explanatory text. However, the abbreviation is used in objective or security functionality identifiers and to distinguish iterations.

9. Bibliography

9.1 Evaluation Documents

[1] Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, Version 3.1, Revision 1, September 2006, CCMB-2006-09-001

[2] Common Criteria for Information Technology Security Evaluation – Part 2: Security functional requirements, Version 3.1, Revision 2, September 2007, CCMB-2007-09-002

[3] Common Criteria for Information Technology Security Evaluation – Part 3: Security Assurance Requirements, Version 3.1, Revision 2, September 2007, CCMB-2007-09-003

[4] Common Methodology for Information Technology Security Evaluation CEM-99/045 Part 2: Evaluation Methodology, Version 3.1, Revision 2, September 2007, CCMB-2007-09-004

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

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

9.2 Developer Documents

[8] Data Sheet P5CD016/021/041/051 and P5Cx081 family, Secure dual interface and contact PKI smart card controller, NXP Semiconductors

[9] Instruction Set, SmartMX-Family, Secure and PKI Smart Card Controller, Philips Semiconductors, Revision 1.1, Document Number 084111, 4 July 2006

[10] Guidance, Delivery and Operation Manual NXP Secure Smartcard Controllers P5CD016/021/041/051 and P5Cx081, NXP Semiconductors, Business Unit Identification

[11] Order Entry Form P5CD016, NXP Semiconductors, Business Unit Identification, Release 4.4

[12] Order Entry Form P5CD021, NXP Semiconductors, Business Unit Identification, Release 4.4

[13] Order Entry Form P5CD041, NXP Semiconductors, Business Unit Identification, Release 4.4

[14] Order Entry Form P5CD051, NXP Semiconductors, Business Unit Identification, Release 4.4

[15] Order Entry Form P5CD081, NXP Semiconductors, Business Unit Identification, Release 4.4

[16] Order Entry Form P5CC081, NXP Semiconductors, Business Unit Identification, Release 4.4

[17] Order Entry Form P5CN081, NXP Semiconductors, Business Unit Identification, Release 4.3

9.3 Other Documents

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

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

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

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

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

[23] ISO/IEC 14443-3:2001 Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 3: Initialization and anticollision

[24] ISO/IEC 14443-4:2001 Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 4: Transmission protocol

[25] ISO/IEC 18092:2004: Information technology – Telecommunications and information exchange between systems – Near Field Communication – Interface and Protocol (NFCIP-1)

[26] Mifare Interface Platform, V2.11, Philips Semiconductors, BL Identification

10. Legal information

10.1 Definitions

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

10.2 Disclaimers

General — 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.

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.

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

10.3 Licenses

ICs with DPA Countermeasures functionality

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

10.4 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.

11. Contents

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

Date of release: 25 October 2010